

Alvaro F. Marques Azevedo

Optimização de Estruturas

DEC
FEUP
1994



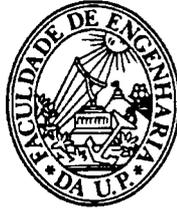
UNIVERSIDADE DO PORTO
FACULDADE DE ENGENHARIA

OPTIMIZAÇÃO DE ESTRUTURAS
COM COMPORTAMENTO LINEAR E NÃO LINEAR

Álvaro Ferreira Marques Azevedo

Dissertação para Doutoramento em Engenharia Civil
Porto, 1994

Reg. 827



UNIVERSIDADE DO PORTO
FACULDADE DE ENGENHARIA

OPTIMIZAÇÃO DE ESTRUTURAS
COM COMPORTAMENTO LINEAR E NÃO LINEAR

Álvaro Ferreira Marques Azevedo

6241042) A260/005 cat

UNIVERSIDADE DO PORTO
Faculdade de Engenharia
BIBLIOTECA 7
N.º 20490-af
CDU
Data 8 1 1994

n.n. 5804

DEC
FEUP

Dissertação para Doutoramento em Engenharia Civil
Porto, 1994

043 D
A 986 σ
RX . 2

Publicação subsidiada pelo
Instituto Nacional de Investigação Científica

Aos meus Pais

AGRADECIMENTOS

O autor expressa por este meio o seu agradecimento ao Prof. Adão da Fonseca, que como supervisor do presente trabalho incentivou e orientou a aplicação da programação matemática à resolução de problemas de engenharia de estruturas.

Deseja também agradecer a todos os colegas do Departamento de Engenharia Civil, e em particular da Secção de Estruturas, as excelentes condições de trabalho proporcionadas, bem como a constante disponibilidade para esclarecedoras trocas de impressões.

O autor agradece ao Instituto Nacional de Investigação Científica a Bolsa de Estudos para Doutoramento no País que lhe foi concedida.

Um agradecimento especial a todas as pessoas que de alguma forma facilitaram o desenvolvimento do presente trabalho, nomeadamente:

- ao Prof. Manuel Azeredo, pelo constante empenho no desenvolvimento dos meios informáticos do Departamento de Engenharia Civil;
- ao Prof. Joaquim Figueiras, ao Prof. Raimundo Delgado e ao Eng. Joaquim Barros pelo incentivo e empenho no desenvolvimento dos métodos numéricos de análise de estruturas;
- ao Prof. Couto Marques, pela constante disponibilidade para trocas de impressões e pelo entusiasmo demonstrado na internacionalização de alguns trabalhos de equipa;
- ao Prof. Serra Neves, pelos esclarecimentos oportunos no domínio das aplicações da programação matemática às estruturas com comportamento não linear;
- à Doutora Paula Milheiro, os esclarecimentos precisos relativos a algumas questões Matemáticas;

- aos entusiastas da informática, que no dia a dia foram contribuindo para o domínio das mais recentes tecnologias, e em particular aos colegas Ricardo Santos, Paulo Avilez, João Macedo, Alípio Ferreira e Luis Guerra;
- e por último, aos seguintes alunos de Mestrado, que no âmbito da disciplina de Optimização de Estruturas realizaram aplicações que de algum modo validaram o *software* de optimização desenvolvido no âmbito do presente trabalho: Elsa Caetano; Miguel Gonçalves, Avelino Teixeira, Francisco Bernardo e Marco Baptista.

O autor deseja também agradecer ao Sr. Manuel Carvalho o excelente trabalho de processamento de texto, preparação das figuras e composição final de toda a documentação.

RESUMO

No presente trabalho é efectuada uma abordagem genérica da optimização, recorrendo ao método de Lagrange-Newton. Uma vez que este método é de segunda ordem, é necessário proceder ao cálculo das primeiras e segundas derivadas das funções que constituem o programa matemático. De modo a permitir uma utilização versátil do programa de computador desenvolvido, foi necessário acrescentar-lhe a capacidade de interpretar programas matemáticos e de proceder à derivação exacta das funções que o constituem. A utilização de variáveis de desvio na conversão das restrições desigualdade em restrições igualdade e a transformação do problema original num problema sem restrições recorrendo a multiplicadores de Lagrange contribuem para que o número de variáveis seja muito elevado. Este problema foi ultrapassado recorrendo a técnicas eficientes de resolução do sistema de equações, cuja matriz dos coeficientes é a Hessiana do Lagrangeano. Na resolução deste subproblema pode ser utilizada uma versão do método de eliminação de Gauss em que é considerada a esparsidade específica da matriz Hessiana ou o método dos gradientes conjugados. Com o objectivo de melhorar o condicionamento do problema são utilizadas diversas técnicas de *scaling*.

O método de optimização desenvolvido no âmbito do presente trabalho é aplicado à optimização de estruturas com comportamento linear ou não linear. Em ambos os casos é utilizada uma formulação integrada, i.e., as variáveis que traduzem o comportamento da estrutura são consideradas como variáveis do problema de optimização. São apresentados resultados correspondentes à minimização do custo de treliças tridimensionais com comportamento linear, tendo sido possível abordar com sucesso a resolução de problemas com um número de variáveis independentes da ordem dos milhares e com um número de restrições da ordem da dezena de milhar. A optimização de estruturas com comportamento não linear é formulada e exemplificada recorrendo a vigas contínuas. É admitida a eventual formação de rótulas plásticas, sendo possível limitar as correspondentes deformações.

ABSTRACT

The thesis presents a generic treatment of the optimization problem via the Lagrange-Newton method. Since this is a second order method, it is necessary to compute the first and second derivatives of the functions that define the mathematical program. In order to enhance its versatility, the code developed has a built-in facility for both interpreting mathematical programs and performing the exact derivation of their constituent functions. The use of slack variables for converting inequality constraints into equality constraints and the transformation of the original problem into an unconstrained one by recourse to Lagrange multipliers, both contribute to increase the number of variables. This problem has been overcome by employing efficient solution techniques for the equation system, whose coefficient matrix is the Hessian of the Lagrangean, namely either a variant of the Gauss elimination method in which the sparsity of the Hessian is accounted for, or the conjugate gradient method. Several scaling techniques are also employed to improve the conditioning of the problem.

The optimization technique developed is applied to the optimization of linear and nonlinear structures. In both cases an integrated formulation is applied, i.e., the variables describing the structural behaviour are considered as variables of the optimization problem. Results are presented for the cost minimization of linear 3D trusses, where successful solutions have been achieved for problems with over 10^3 independent design variables and over 10^4 constraints. The optimization of nonlinear structures is illustrated with continuous beam applications, where the formulation of plastic hinges is incorporated and the corresponding deformations may be bounded.

ÍNDICE

AGRADECIMENTOS.....	i
RESUMO	iii
ABSTRACT	v
ÍNDICE	vii
SIMBOLOGIA	xiii

CAPÍTULO 1 - INTRODUÇÃO

1.1 - OTIMIZAÇÃO DE ESTRUTURAS	1.2
1.2 - CLASSIFICAÇÃO DE PROBLEMAS DE OTIMIZAÇÃO DE ESTRUTURAS	1.3
1.3 - RESOLUÇÃO DE PROBLEMAS DE OTIMIZAÇÃO DE ESTRUTURAS	1.5

CAPÍTULO 2 - PROGRAMAÇÃO MATEMÁTICA

2.1 - COMPONENTES DE UM PROGRAMA MATEMÁTICO	2.4
2.1.1 - Variáveis de projecto	2.5
2.1.2 - Função objectivo	2.6
2.1.3 - Restrições igualdade	2.7
2.1.4 - Restrições desigualdade.....	2.10
2.2 - CONDIÇÕES DE OPTIMALIDADE.....	2.11
2.2.1 - Função com uma variável e sem restrições	2.13
2.2.2 - Função com n variáveis e sem restrições	2.15
2.2.3 - Restrições igualdade - Lagrangeano	2.19
2.2.4 - Restrições desigualdade	2.25
2.2.5 - Significado dos multiplicadores de Lagrange.....	2.29
2.2.6 - Condições de Karush-Kuhn-Tucker	2.33
2.3 - OBSERVAÇÕES FINAIS	2.38

CAPÍTULO 3 - MÉTODOS DE OPTIMIZAÇÃO

3.1 - MINIMIZAÇÃO DE FUNÇÕES COM UMA VARIÁVEL.....	3.5
3.1.1 - Métodos de ordem zero	3.5
3.1.2 - Métodos de primeira ordem.....	3.7
3.1.3 - Métodos de segunda ordem	3.10

3.2 - MINIMIZAÇÃO DE FUNÇÕES COM n VARIÁVEIS E SEM RESTRIÇÕES	3.11
3.2.1 - Métodos de ordem zero	3.12
3.2.2 - Métodos de primeira ordem	3.16
3.2.3 - Métodos de segunda ordem.....	3.18
3.3 - MINIMIZAÇÃO DE FUNÇÕES COM n VARIÁVEIS E COM RESTRIÇÕES	3.23
3.3.1 - Métodos primais.....	3.24
3.3.2 - Métodos de transformação	3.29
3.4 - CRITÉRIOS DE SELECÇÃO DE MÉTODOS DE OPTIMIZAÇÃO	3.37

CAPÍTULO 4 - MÉTODO DE NEWTON

4.1 - ALGORITMO DE OPTIMIZAÇÃO.....	4.1
4.2 - TÉCNICAS DE ARMAZENAMENTO E DERIVAÇÃO DE FUNÇÕES	4.6
4.3 - TÉCNICAS DE <i>SCALING</i>	4.9
4.4 - RESOLUÇÃO DO SISTEMA DE EQUAÇÕES LINEARES.....	4.17
4.4.1 - Método directo	4.17
4.4.2 - Método iterativo	4.22
4.5 - PESQUISA UNIDIMENSIONAL.....	4.27
4.6 - INTERRUPÇÃO DO PROCESSO ITERATIVO	4.29
4.7 - OBSERVAÇÕES FINAIS.....	4.32

CAPÍTULO 5 - PROGRAMA DE CÁLCULO AUTOMÁTICO

5.1 - DESCRIÇÃO DO PROGRAMA MATEMÁTICO.....	5.2
5.2 - CÁLCULO DO VALOR INICIAL DAS VARIÁVEIS DEPENDENTES	5.6
5.3 - PRÉ-SUBSTITUIÇÃO DE VARIÁVEIS.....	5.8
5.4 - DESCRIÇÃO DO PROGRAMA NEWTOP	5.12
5.5 - SOLUÇÃO INICIAL	5.19
5.6 - COMPARAÇÃO COM OS RESULTADOS DE OUTROS AUTORES.....	5.22
5.7 - OBSERVAÇÕES FINAIS	5.24

CAPÍTULO 6 - OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO LINEAR

6.1 - CRITÉRIOS DE OPTIMALIDADE.....	6.5
6.1.1 - <i>Fully Stressed Design</i>	6.5
6.1.2 - Optimização de uma treliça com 10 barras.....	6.8
6.1.3 - Restrições de deslocamento.....	6.10
6.1.4 - Optimização de uma treliça com 47 barras.....	6.14
6.1.5 - Observações finais	6.18
6.2 - ANÁLISE DE SENSIBILIDADES	6.19
6.2.1 - Formulação	6.20
6.2.2 - Exemplos	6.22

6.2.3 - Observações finais.....	6.26
6.3 - OPTIMIZAÇÃO DE TRELIÇAS PELO MÉTODO DE NEWTON - FORMULAÇÃO INTEGRADA.....	6.27
6.3.1- Geração do programa matemático.....	6.28
6.3.2 - Treliza simétrica com três barras e dois casos de carga.....	6.36
6.3.3 - Trelizas parametrizadas.....	6.46
6.3.3.1 - Formulação geral.....	6.47
6.3.3.2 - Preparação da solução inicial.....	6.53
6.3.3.3 - Apresentação dos resultados.....	6.55
6.3.4 - Comparação com os resultados de outros autores.....	6.75
6.3.4.1 - Treliza de 10 barras.....	6.77
6.3.4.2 - Treliza de 200 barras.....	6.80
6.3.4.3 - Treliza de 72 barras.....	6.85
6.3.4.4 - Conclusões do estudo comparativo.....	6.89

CAPÍTULO 7 - OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO NÃO LINEAR

7.1 - FORMULAÇÃO DO COMPORTAMENTO ELASTOPLÁSTICO.....	7.2
7.2 - ANÁLISE DE ESTRUTURAS COM COMPORTAMENTO ELASTOPLÁSTICO.....	7.10
7.3 - OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO ELASTOPLÁSTICO.....	7.18
7.4 - OBSERVAÇÕES FINAIS.....	7.22

CAPÍTULO 8 - CONCLUSÕES

8.1 - OBSERVAÇÕES FINAIS 8.1

8.2 - SUGESTÕES PARA FUTUROS DESENVOLVIMENTOS..... 8.3

REFERÊNCIAS BIBLIOGRÁFICAS R.1

SIMBOLOGIA

SIMBOLOGIA NA PROGRAMAÇÃO MATEMÁTICA

$\alpha \rightarrow$	parâmetro de pesquisa unidimensional
$\varepsilon \rightarrow$	erro ou tolerância; assume em geral valores positivos de ordem de grandeza muito pequena
$\varepsilon^g \rightarrow$	variável que é adicionada a uma restrição desigualdade
$\varepsilon^h \rightarrow$	variável que é adicionada a uma restrição igualdade
$\Phi \rightarrow$	pseudo função objectivo utilizada na minimização sem restrições sequencial e no método do Lagrangeano aumentado
$\lambda \rightarrow$	multiplicador de Lagrange
$\bar{\lambda} \rightarrow$	multiplicador de Lagrange transformado
$\lambda^g \rightarrow$	multiplicador de Lagrange associado a restrições desigualdade
$\lambda^h \rightarrow$	multiplicador de Lagrange associado a restrições igualdade
$\nabla f \rightarrow$	gradiente de uma função
$\nabla_x f \rightarrow$	gradiente de uma função relativamente às variáveis x
$\Delta x \rightarrow$	direcção ou variação da solução
$A \rightarrow$	matriz dos coeficientes

$b \rightarrow$	vector dos termos independentes
$C \rightarrow$	matriz de pré-condicionamento
$d \rightarrow$	direcção ou variação da solução
$D(\tilde{A}) \rightarrow$	matriz diagonal; os elementos da diagonal de D coincidem com os da diagonal de \tilde{A}
$E \rightarrow$	erro que é minimizado na pesquisa unidimensional
$f \rightarrow$	função objectivo
$F(s) \rightarrow$	folga de uma restrição desigualdade
$g \rightarrow$	restrição desigualdade
$h \rightarrow$	restrição igualdade
$H \rightarrow$	matriz Hessiana
$I \rightarrow$	matriz identidade
$l \rightarrow$	número de restrições igualdade
$L \rightarrow$	Lagrangeano
$L' \rightarrow$	função objectivo dual
$m \rightarrow$	número de restrições desigualdade
$n \rightarrow$	número de variáveis de projecto
$nt \rightarrow$	soma do número de variáveis de projecto com o número de restrições igualdade ($n + l$)
$q \rightarrow$	número da iteração
$r \rightarrow$	vector cujas componentes são os resíduos $\tilde{A}\tilde{x} - \tilde{b}$
$s \rightarrow$	variável de desvio
$\bar{s} \rightarrow$	variável de desvio transformada

$w \rightarrow$	vector auxiliar utilizado no algoritmo correspondente ao método dos gradientes conjugados
$x \rightarrow$	variável de projecto
$x_{\min}, x_{\max} \rightarrow$	valor mínimo/máximo de uma variável de projecto (<i>simple bounds</i> ou <i>side constraints</i>)
$x^0 \rightarrow$	solução inicial
$x^* \rightarrow$	solução óptima
$\bar{x} \rightarrow$	variável de projecto transformada
$X \rightarrow$	vector que contém todas as variáveis do Lagrangeano $\left(\underset{\sim}{s}, \underset{\sim}{\lambda^g}, \underset{\sim}{x}, \underset{\sim}{\lambda^h} \right)$
$Z \rightarrow$	coeficientes associados às técnicas de <i>scaling</i>

SIMBOLOGIA NO CAPÍTULO 6

$\gamma \rightarrow$	peso específico do material
$\sigma \rightarrow$	tensão normal
$\sigma_{\min, \max} \rightarrow$	tensão mínima/máxima de um material
$\sigma_{cr} \rightarrow$	tensão de encurvadura de Euler
$\Lambda \rightarrow$	variáveis adjuntas
$a \rightarrow$	área da secção transversal da barra
$a_{\min, \max} \rightarrow$	área mínima/máxima da secção transversal da barra
$c \rightarrow$	custo por unidade de volume de um material
$C \rightarrow$	custo global da solução

d	→	deslocamentos dos nós
$d_{\min, \max}$	→	deslocamento mínimo/máximo de um nó
E	→	módulo de Young do material
F	→	forças nodais equivalentes
I	→	momento de inércia da secção transversal da barra
K	→	matriz de rigidez
K_{FSD}	→	factor que multiplicado pela solução FSD dá origem a uma solução que respeita as restrições deslocamento
L	→	comprimento da barra
M	→	momento flector
N	→	esforço axial numa barra
NB	→	número de barras
p	→	carga uniformemente distribuída na barra
u	→	deslocamentos dos nós

SIMBOLOGIA NO CAPÍTULO 7

Variáveis utilizadas no programa matemático

Z	→	factor de carga
L_i	→	comprimento do tramo
a_i	→	comprimento da barra esquerda do tramo
b_i	→	comprimento da barra direita do tramo

$E_i \rightarrow$	módulo de Young do material do tramo
$G_i \rightarrow$	peso específico do material do tramo
$c_i \rightarrow$	custo por unidade de volume do material do tramo
$S_i \rightarrow$	tensão de cedência do material do tramo
$e_i \rightarrow$	máxima rotação relativa numa rótula plástica
$B_i \rightarrow$	largura da secção do tramo
$H_i \rightarrow$	altura da secção do tramo
$I_i \rightarrow$	momento de inércia da secção do tramo
$M_i \rightarrow$	momento plástico da secção do tramo
$p_i \rightarrow$	acção exterior distribuída no tramo
$d_{is} \rightarrow$	deslocamento no tramo
$D_{jt} \rightarrow$	deslocamento no nó
$F_{is} \rightarrow$	força no tramo
$Q_{jt} \rightarrow$	acção exterior concentrada no nó
$R_{jt} \rightarrow$	reação de apoio

Variáveis auxiliares utilizadas na formulação

$K \rightarrow$	matriz de rigidez da barra
$P \rightarrow$	acção nodal equivalente à acção exterior distribuída

Índices

$n \rightarrow$	número de tramos
$m \rightarrow$	número de nós

CAPÍTULO 1

INTRODUÇÃO

O principal objectivo do projectista de estruturas devia ser sempre o da obtenção da solução mais vantajosa para cada problema. De um modo geral esta tarefa implica uma pesquisa exhaustiva, que obrigaria o projectista a ter em consideração todas as soluções possíveis. Para que esta tarefa seja realizável é necessário que o número de soluções possíveis seja finito e que o estudo de todas essas soluções possa ser efectuado num intervalo de tempo razoável. Esta situação só ocorre em problemas muito simples e com um reduzido número de parâmetros, que por sua vez só possam assumir um também reduzido número de valores. Nos restantes casos torna-se necessário abordar o problema de um modo mais ou menos simplificado de modo a ser possível a sua resolução. É tarefa do projectista ajuizar se o problema simplificado possui uma solução que aproxima com um rigor satisfatório a solução do problema original.

Na formulação de um problema de engenharia estrutural a primeira fase consiste habitualmente na identificação de um conjunto de parâmetros gerais, cuja quantificação antecede a intervenção do projectista de estruturas. Referem-se como exemplos os seguintes casos: selecção do perfil de uma estrada antecedendo o projecto de uma ponte, cota de uma albufeira fixada antes de projectar a barragem, etc. Segue-se a selecção do tipo de estrutura. No caso da ponte pode ser adoptada uma solução com múltiplos pilares, em arco, etc. No caso da barragem a decisão a tomar poderá ser entre uma solução em abóbada ou do tipo gravidade. Uma vez que em cada situação existe um reduzido número de tipos de estrutura possíveis de serem concretizadas, as decisões relativas a esta fase são geralmente baseadas na comparação entre estudos preliminares. Após a definição geral do problema e a selecção do tipo de estrutura, segue-se uma fase que, na generalidade dos casos, pode ser matematicamente formulada como uma minimização de um custo. No presente trabalho apenas serão aprofundados os aspectos relativos a esta última fase.

1.1 - OPTIMIZAÇÃO DE ESTRUTURAS

Optimizar uma estrutura significa na generalidade dos casos determinar a solução que torna mínimo o valor de uma função escalar que traduz, directa ou indirectamente, um custo e que é designada função objectivo. Para que a estrutura tenha utilidade, devem ser satisfeitas algumas exigências relativas à sua segurança e *performance*. Estas exigências constituem um conjunto de igualdades e/ou desigualdades designadas restrições. Em certos casos particulares a formulação do problema de optimização pode apresentar alguns aspectos diferentes dos atrás referidos. Se o objectivo for a maximização da segurança ou da *performance* considerando o custo fixo, resulta um problema que pode ser resolvido de um modo semelhante e ao qual está também associado o conceito de optimização da estrutura. Existem problemas em que o objectivo pode não ser único, sendo a respectiva optimização designada multi-objectivo. Estes problemas podem ser resolvidos com uma única função objectivo que resulta da combinação ponderada dos diversos objectivos iniciais.

Nos problemas de optimização está quase sempre presente a necessidade de quantificar um custo. Se a estrutura for constituída por um único material, pode-se em geral considerar que o seu custo depende do volume do material que vai ser utilizado na sua construção. Nestes casos, se em vez do volume for considerado o peso do material, a solução do problema é obviamente a mesma. Se estiverem presentes diversos materiais, o custo da estrutura pode ser facilmente calculado a partir dos respectivos custos unitários. Qualquer um destes procedimentos fornece apenas o custo associado à quantidade de material utilizado na própria estrutura. A necessidade de executar estruturas auxiliares durante as fases construtivas tem custos adicionais que dependem da solução adoptada e que podem influenciar significativamente o respectivo custo global. A consideração destes custos na função objectivo pode nalguns casos ser simples e noutras muito complexa. Os custos de cofragens de peças de betão é em geral fácil de incluir no custo global da solução, desde que se conheça o respectivo custo por unidade de superfície. Contrastando com esta simplicidade estão por exemplo os cimbramentos, cavaletes ou vigas de lançamento, cujo custo depende da solução adoptada para a estrutura de um modo difícil de quantificar. Resta por último referir um custo cuja avaliação e inclusão na função objectivo se revela ainda mais complexa. É evidente que uma solução que apresente uma grande regularidade e repetitividade é mais fácil e economicamente executada do que uma solução muito heterogénea. Se não forem impostas restrições que limitem essa heterogeneidade, a solução do problema de optimização pode ser de tal forma complexa que a sua execução teria custos adicionais incontroláveis. A consideração de restrições adicionais que limitem a heterogeneidade da estrutura conduzirá necessariamente a uma solução de custo mais elevado. A ponderação destes aspectos - heterogeneidade e simplicidade de execução - é muito difícil de quantificar e de incluir na

função objectivo. As decisões relativas à imposição de restrições que limitem a heterogeneidade da solução final são em geral baseadas na intuição e experiência do projectista.

1.2 - CLASSIFICAÇÃO DE PROBLEMAS DE OPTIMIZAÇÃO DE ESTRUTURAS

Quer na análise, quer na optimização, é essencial considerar o comportamento da estrutura de um modo simplificado. O procedimento mais versátil consiste na discretização da estrutura em elementos finitos, formulando o respectivo comportamento por intermédio do método dos deslocamentos. Deste modo, um campo de deslocamentos contínuo passa a ser caracterizado por um número finito de deslocamentos em pontos nodais. As tensões passam também a ser apenas verificadas num número finito de pontos de amostragem. Nas estruturas com comportamento linear, os deslocamentos são considerados como variáveis e as tensões como funções que dependem dos deslocamentos e de algumas características da estrutura. Nos problemas de análise de estruturas o respectivo comportamento é traduzido por um sistema de equações lineares em que apenas os deslocamentos dos nós figuram como incógnitas. As tensões são calculadas *a posteriori*. Nos problemas de optimização de estruturas, os deslocamentos dos nós e as tensões são variáveis que na generalidade dos casos podem figurar na formulação de um modo implícito.

Nos problemas de optimização, além dos deslocamentos dos nós e das tensões, existem outras variáveis que dependem do tipo de problema e do grau de complexidade que o projectista pretende incluir no estudo da estrutura. No conjunto de parâmetros que definem a solução estrutural, é necessário seleccionar previamente aqueles que podem ser modificados pelo algoritmo de optimização. Estes parâmetros passam a pertencer ao conjunto das variáveis e os restantes assumem valores constantes durante o processo de optimização da estrutura. O tipo de parâmetros que não são fixados designam-se por variáveis de projecto e constituem o principal critério de classificação de problemas de optimização de estruturas. Apresenta-se em seguida uma breve descrição de cada um desses tipos de problemas.

- 1) **Optimização de secções transversais** - as variáveis de projecto são as características das secções transversais das barras em estruturas reticuladas (treliças ou pórticos). As coordenadas dos nós e as propriedades dos materiais utilizados são fixas. As secções transversais podem ser caracterizadas por intermédio da respectiva área, momento de inércia ou dimensões transversais. Este tipo de problema é o mais simples de formular e consequentemente é o que mais vezes é utilizado para avaliar e exemplificar as características dos métodos de optimização.
- 2) **Optimização de espessuras** - este tipo de problema encontra-se associado às estruturas laminares (membranas, lajes e cascas). As variáveis de projecto são as espessuras dos elementos finitos ou a espessura em cada ponto nodal. As coordenadas dos nós e as propriedades dos materiais utilizados são fixas. Existem muitas semelhanças entre este tipo de problema e a optimização de secções transversais atrás descrita.
- 3) **Optimização de forma** - a principal diferença entre este tipo de problema e os outros dois já descritos consiste no facto de as coordenadas dos nós serem também variáveis de projecto. No caso da optimização de forma em sólidos tridimensionais, apenas as coordenadas dos nós são variáveis, porque na respectiva formulação não são consideradas secções nem dimensões transversais de elementos finitos. Na prática é aconselhável manter um número significativo de nós com as coordenadas fixas para que a estrutura não seja radicalmente modificada. Em certos casos pode interessar que os nós cuja posição é livre fiquem situados sobre uma linha ou superfície de geometria simples. Os parâmetros que definem esta linha ou superfície são também incluídos no conjunto das variáveis do problema de optimização. Estas restrições adicionais destinam-se a evitar que a solução óptima apresente grandes irregularidades.
- 4) **Optimização da localização de acções** - o problema que melhor exemplifica este tipo de optimização é o da localização e quantificação do pré-esforço numa estrutura. Podem também estar presentes variáveis de outro tipo, tais como dimensões de secções transversais, espessuras ou coordenadas de nós.
- 5) **Optimização de materiais** - esta designação é utilizada sempre que uma ou mais propriedades dos materiais são consideradas como variáveis do problema de optimização. Estas variáveis podem traduzir o facto de diferentes materiais apresentarem propriedades distintas ou podem estar associadas à variação de propriedades que ocorre entre classes de resistência de um mesmo material. Uma vez que não é possível dispor de materiais com uma variação contínua de propriedades, as respectivas variáveis devem ser consideradas discretas.

- 6) **Optimização da topologia** - são assim designados os problemas cuja resolução envolve a supressão ou o acréscimo de componentes da estrutura. No caso das estruturas reticuladas, para além de retirar e adicionar barras, o algoritmo de optimização deve ser capaz de modificar as conexões existentes. Nos meios contínuos é importante a possibilidade de criar ou suprimir aberturas. Também são consideradas como optimização da topologia as situações em que o número de ligações ao exterior e a sua posição são variáveis. Estes problemas são em geral muito complexos, só sendo possível a resolução de casos simples. Apresenta-se como exemplo o caso de uma treliça em que cada nó se encontra inicialmente ligado a todos os outros. A tarefa essencial do algoritmo de optimização será a de decidir quais são as barras que devem ser suprimidas.

1.3 - RESOLUÇÃO DE PROBLEMAS DE OPTIMIZAÇÃO DE ESTRUTURAS

Em muitos problemas de dimensionamento de estruturas, cuja resolução é efectuada recorrendo a técnicas de optimização, a solução final apresenta características simples. Refere-se como exemplo o facto de as soluções óptimas relativas a treliças hiperstáticas serem geralmente configurações isostáticas em que os pontos de aplicação das cargas e os apoios se encontram ligados pelo trajecto mais curto. Nestes casos as soluções óptimas parecem estar ao alcance de qualquer projectista que recorra apenas à intuição e à experiência adquirida na resolução de problemas semelhantes. Em muitos outros casos, soluções quase óptimas podem ser obtidas recorrendo apenas a regras heurísticas e a critérios de dimensionamento elementares. Existem contudo situações em que não é fácil chegar às soluções mais convenientes. Referem-se como exemplo problemas de dimensionamento em que estão presentes diversos casos de carga a solicitar a estrutura de distintos modos, diversos materiais com custos diferentes, restrições contraditórias, etc. Nestes casos é fundamental o recurso a técnicas de optimização mais ou menos elaboradas.

A utilização de métodos de optimização de estruturas não deve limitar-se à resolução de problemas complexos. Apresentam-se em seguida algumas situações em que a obtenção da solução óptima é fundamental:

- estruturas invulgares ou mesmo inéditas, em relação às quais não exista experiência acumulada;

- estruturas importantes e de custo elevado;
- quando o projecto da estrutura se destinar a ser executado em múltiplos exemplares;
- quando é exigida à estrutura uma elevada *performance*.

Para que um projectista possa recorrer a métodos de optimização para resolver problemas de dimensionamento de estruturas, é fundamental colocar à sua disposição códigos computacionais eficientes, robustos, versáteis e fáceis de utilizar. Os métodos de optimização apresentam quase sempre uma complexidade considerável, que faz com que este conjunto de objectivos seja dificilmente alcançado. No passado, devido a limitações nos recursos informáticos, a eficiência foi um dos objectivos prioritários de modo a permitir a resolução de problemas com interesse prático. Este aumento da eficiência foi quase sempre conseguido à custa de uma diminuição da robustez e da versatilidade dos algoritmos [Aro87b]. As simplificações introduzidas vieram também dificultar a obtenção de resultados precisos. No presente trabalho foi dada prioridade à versatilidade, precisão e facilidade de utilização do código computacional. O constante aumento da *performance* dos computadores tem contribuído para que a eficiência deixe de ser um aspecto problemático. Se no futuro passarem a estar disponíveis programas de computador suficientemente robustos, eficientes e fáceis de utilizar, poderá vir a ser possível recorrer à optimização sempre que for necessário dimensionar uma estrutura. Espera-se que o presente trabalho corresponda a um progresso nesse sentido.

CAPÍTULO 2

PROGRAMAÇÃO MATEMÁTICA

A formulação matemática dos problemas de optimização em geral, e de optimização de estruturas em particular, traduz-se habitualmente no seguinte problema:

$$\text{Minimizar } f(x_1, \dots, x_n) \quad (2.1a)$$

sujeito a

$$\begin{aligned} g_1(x_1, \dots, x_n) &\leq 0 \\ \vdots & \\ g_m(x_1, \dots, x_n) &\leq 0 \end{aligned} \quad (2.1b)$$

$$\begin{aligned} h_1(x_1, \dots, x_n) &= 0 \\ \vdots & \\ h_l(x_1, \dots, x_n) &= 0 \end{aligned} \quad (2.1c)$$

Nesta descrição, as variáveis x_i ($i = 1, \dots, n$) são as variáveis de projecto, a função escalar f é a função objectivo, $g_j \leq 0$ ($j = 1, \dots, m$) são as restrições do tipo desigualdade e por último $h_k = 0$ ($k = 1, \dots, l$) são as restrições do tipo igualdade. O problema (2.1) é designado programa matemático e pode ser escrito do seguinte modo mais compacto

$$\text{Min. } f(\underline{x}) \quad (2.2a)$$

s.a

$$\underline{g}(\underline{x}) \leq \underline{0} \quad (2.2b)$$

$$\underline{h}(\underline{x}) = \underline{0} \quad (2.2c)$$

A solução do problema (2.2) é o vector \tilde{x}^* que torna mínimo o valor da função objectivo f , respeitando as restrições $\tilde{g} \leq 0$ e $\tilde{h} = 0$. Esta solução é designada solução óptima. Em casos particulares pode existir mais do que uma solução óptima. As soluções que apenas respeitam as restrições (2.2b) e (2.2c) designam-se soluções admissíveis, sendo o domínio por elas definido designado região admissível. As soluções que não respeitam a totalidade das restrições são soluções inadmissíveis. Se o problema de optimização que se pretende resolver for correctamente formulado, admite-se que a sua solução coincide com a solução óptima do programa matemático (2.2). Os problemas de maximização são também contemplados pela formulação (2.2) desde que se atenda à seguinte equivalência

$$\text{Maximizar } f(\tilde{x}) \Leftrightarrow \text{Minimizar } -f(\tilde{x}) \quad (2.3)$$

As restrições desigualdade do tipo $g \geq 0$ podem ser convertidas em restrições do tipo (2.2b) por intermédio de uma multiplicação por (-1)

$$g \geq 0 \Leftrightarrow -g \leq 0 \quad (2.4)$$

No programa matemático (2.2) considera-se geralmente que as variáveis x podem assumir qualquer valor real e que as funções f , g e h são contínuas e reais. É habitual considerar que as respectivas derivadas são também contínuas. A máxima ordem em relação à qual as derivadas devem ser contínuas depende do método de optimização utilizado. Nestas circunstâncias o programa matemático (2.2) é designado programa não linear. No Quadro 2.1 encontram-se algumas características das variáveis, função objectivo e restrições que dão origem a programas matemáticos com determinadas particularidades [Gil81]. Enquanto que nuns casos estes aspectos particulares dificultam a respectiva resolução, noutros casos permitem o desenvolvimento de métodos mais eficientes e mais robustos do que os destinados à resolução de programas não lineares.

Quadro 2.1 - Casos particulares de variáveis, função objectivo e restrições.

	Particularidade	Descrição/Exemplo
Variáveis	contínuas	Assumem qualquer valor real
	discretas	Só assumem alguns valores reais
	inteiras	Só assumem valores inteiros
	binárias	Só assumem os valores 0 e 1
Função objectivo	função de uma variável	Ex: $f(x) = 3x^4 + \sqrt{x^2 + 1}$
	linear	Ex: $f(\tilde{x}) = 3x_1 - 6x_2$
	quadrática	Ex: $f(\tilde{x}) = 3x_1^2 - 6x_1x_2$
	polinomial	Ex: $f(\tilde{x}) = 4x_1^2x_2 + 6x_2^3x_3x_5^4 + 7x_4^2 - 1$
	não linear com continuidade	Ex: $f(\tilde{x}) = (x_1 + 1)(x_2^4 + 5x_3^2 + 1)^{-2.3}$
	não linear sem continuidade	Ex: $f(\tilde{x}) = \ 5x_1x_2^3\ $
Restrições igualdade e desigualdade	sem restrições	----
	limites nos valores das variáveis	$x_{\min} \leq x_i \leq x_{\max}$
	lineares	Ex: $g(\tilde{x}) = 3x_1 - 6x_2 \leq 0$
	polinomiais	Ex: $g(\tilde{x}) = 4x_1^2x_2 + 6x_2^3x_3x_5^4 + 7x_4^2 - 1 \leq 0$
	não lineares com continuidade	Ex: $g(\tilde{x}) = (x_1 + 1)(x_2^4 + 5x_3^2 + 1)^{-2.3} \leq 0$
	não lineares sem continuidade	Ex: $g(\tilde{x}) = \ 5x_1x_2^3\ - x_3 \leq 0$

Algumas das particularidades referidas no Quadro 2.1 dão origem a programas matemáticos com designações específicas. Quando a função objectivo e as restrições são funções lineares, o programa matemático (2.2) recebe a designação de programa linear e pode ser resolvido pelo método *simplex* [Dan63]. Este é considerado o mais antigo método de resolução de programas matemáticos, tendo sido desenvolvido por George Dantzig em 1947. A programação quadrática apresenta como única diferença relativamente à programação linear o facto de a função objectivo poder ser quadrática. Quando a função objectivo e as restrições de desigualdade são funções polinomiais generalizadas (signómios), o programa matemático passa a designar-se programação geométrica, sendo neste caso necessário considerar mais algumas particularidades [Kir81]. Se não existir nenhuma restrição, o problema é designado minimização sem restrições. Uma classe importante de métodos de optimização baseia-se na resolução de problemas com restrições recorrendo a sequências de minimizações sem restrições [Hil90]. Torna-se portanto fundamental dispor de métodos eficientes para resolver este caso particular. Quando as variáveis do problema só podem assumir valores inteiros, a designação adoptada é a de programação inteira e quando só podem assumir os valores 0 e 1 trata-se de programação inteira binária.

Os métodos destinados à resolução de problemas de optimização de estruturas só em situações muito particulares podem tirar partido das simplificações que resultam do facto das funções serem lineares ou quadráticas. Alguns métodos de optimização baseiam-se em sequências de programas lineares ou quadráticos, que são mais robustos quando o problema inicial se apresenta com um elevado número de restrições lineares ou quase lineares. Este objectivo é por vezes alcançado recorrendo à utilização de variáveis recíprocas, i.e., substituindo x_i por $1/x_i$ [Fle87]. Na optimização de estruturas depara-se frequentemente com uma dificuldade adicional, que consiste no facto de algumas variáveis serem discretas.

2.1 - COMPONENTES DE UM PROGRAMA MATEMÁTICO

Os problemas de optimização de estruturas podem de um modo geral ser formulados como um programa matemático, que apresente a forma indicada em (2.2). Conforme o tipo de problema que está a ser abordado, as variáveis, função objectivo e restrições têm diferentes significados. Apresentam-se em seguida algumas características destas componentes do programa matemático, bem como exemplos da sua concretização em problemas de optimização de estruturas.

2.1.1 - Variáveis de projecto

As grandezas que em cada caso são consideradas variáveis dependem do tipo de problema de optimização de estruturas (ver Capítulo 1). Na generalidade dos casos os deslocamentos dos nós estão presentes como variáveis contínuas. No Capítulo 6 será apresentado um procedimento que faz com que os deslocamentos passem a ser considerados de um modo implícito, deixando de figurar no vector das variáveis de projecto x . Na optimização de secções transversais é em geral conveniente considerar as respectivas características como variáveis discretas para evitar uma excessiva heterogeneidade da solução óptima e para que seja possível construir a estrutura com secções standardizadas. Na prática revela-se quase sempre mais simples resolver o problema de optimização com as características das secções transversais como variáveis contínuas. Em seguida é possível passar para a solução discreta recorrendo a técnicas rigorosas ou aproximadas. Na optimização de espessuras é mais frequente considerar as variáveis contínuas. Em certos casos pode ter interesse considerar as espessuras como variáveis discretas, pelos mesmos motivos indicados no caso das secções transversais.

Na optimização de forma as variáveis de projecto são as coordenadas dos nós, sendo conveniente considerá-las como variáveis contínuas. Se os nós cujas coordenadas são variáveis forem obrigados a situar-se sobre uma linha ou superfície (por exemplo B-Spline ou Bezier), as posições dos respectivos pontos de controle passam a ser as variáveis de projecto [Bec87]. Neste caso as coordenadas dos nós passam a ser variáveis dependentes e podem figurar na formulação do problema de um modo implícito. Quando as posições dos pontos de controle das superfícies de Bezier ou dos B-Splines forem as variáveis de projecto, estas devem ser consideradas contínuas.

No Capítulo 1 foi apresentado como exemplo de optimização da localização de acções a minimização do custo do pré-esforço a aplicar numa estrutura. Neste caso as variáveis de projecto são as coordenadas dos pontos que definem o traçado do cabo e o valor do pré-esforço. Estes dois tipos de variáveis podem ser consideradas contínuas.

Devido ao facto de não existirem materiais com uma variação contínua de propriedades e por ser habitual considerar um número reduzido de classes de resistência, a optimização de materiais deve ser abordada com variáveis discretas. Também são considerados como optimização de materiais os problemas em que as variáveis de projecto são as orientações das fibras em compósitos. Neste caso os ângulos que definem essas orientações podem ter uma variação contínua. Para facilitar o processo construtivo é também

conveniente restringir o número de ângulos possíveis, passando nesse caso as respectivas variáveis a ser discretas.

Os problemas de otimização da topologia podem em certos casos ser formulados com variáveis binárias, que também são designadas variáveis 0-1. Cada elemento da estrutura deve ser multiplicado por uma variável binária, bem como as restrições relativas ao seu comportamento. Na solução ótima os elementos aos quais está associada uma variável binária nula são suprimidos e os restantes mantidos.

2.1.2 - Função objectivo

A função objectivo $f(\tilde{x})$ que figura no programa matemático (2.2) depende do tipo de problema de otimização e da grandeza que se pretende minimizar. Na generalidade dos casos é facilmente explicitada a partir das variáveis de projecto. Se a estrutura for constituída por um único material é habitual considerar que ao minimizar o volume está-se a minimizar o custo. Se estiver presente mais do que um material a função objectivo deve representar o custo da estrutura, sendo calculada a partir do custo por unidade de volume de cada material. Se se pretender minimizar o peso de uma estrutura constituída por diversos materiais é suficiente considerar como custo por unidade de volume o peso específico de cada um deles.

Em certos problemas pouco frequentes pode haver interesse em maximizar a performance de uma estrutura com custo fixo. Nestes casos a função objectivo representa a grandeza associada a essa performance. Apresenta-se como exemplo a maximização da rigidez, da carga crítica de instabilidade, da menor frequência própria, etc. Nestes problemas a solução ótima representa a distribuição de recursos que maximiza a performance, respeitando as restrições.

Os problemas de análises de estruturas com comportamento linear podem ser formulados como uma minimização da energia, considerando como variáveis os deslocamentos e as reacções. Se se pretender efectuar a análise de uma estrutura com comportamento elastoplástico o objectivo passa a ser a maximização do factor de carga [Ada80].

2.1.3 - Restrições igualdade

As restrições igualdade que figuram no programa matemático (2.2) podem ser eliminadas desde que em cada uma delas seja possível explicitar uma variável de projecto distinta. Estas variáveis podem em seguir ser substituídas nas restantes expressões do programa matemático. Para clarificar este procedimento apresenta-se o seguinte exemplo

$$\text{Min. } x_1^2 + x_2^2 + x_3^2 \quad (2.5a)$$

s.a

$$-5x_1 + x_2 \leq 0 \quad (2.5b)$$

$$x_1 - x_2 + x_3 = 0 \quad (2.5c)$$

$$x_3 - 2 = 0 \quad (2.5d)$$

Na expressão (2.5c) é possível explicitar a variável x_2 , que em seguida pode ser substituída nas restantes expressões do programa matemático. Após a substituição a restrição (2.5c) deixa de ser necessária. Na expressão (2.5d) é possível explicitar x_3 e efectuar o mesmo procedimento. Após a realização de algumas simplificações resulta o seguinte programa matemático

$$\text{Min. } 2x_1^2 + 4x_1 + 8 \quad (2.6a)$$

s.a

$$-2x_1 + 1 \leq 0 \quad (2.6b)$$

A solução de (2.6) coincide necessariamente com a solução de (2.5). Este procedimento conduz a um programa matemático com menos variáveis e com menos restrições, que provavelmente será resolvido de um modo mais eficiente. Se no programa matemático estiverem presentes muitas restrições igualdade e se cada uma delas relacionar um elevado número de variáveis, as sucessivas substituições conduzem a expressões de exagerada complexidade. Nestes casos o aumento de eficiência atrás referido pode não ocorrer. Por este motivo apenas devem ser substituídas restrições igualdade em que figure um número muito reduzido de variáveis. Quando o número de restrições é elevado torna-se fundamental que as substituições sucessivas e as consequentes simplificações sejam efectuadas com o auxílio de um programa de computador.

Nos problemas de optimização de estruturas algumas restrições igualdade relacionam um número reduzido de variáveis e devem ser substituídas nas restantes expressões do programa matemático. Apresenta-se como exemplo as restrições do tipo

$$x_i = \text{constante} \quad (2.7)$$

Estas restrições estão presentes quando é necessário impor valores fixos a deslocamentos, nos problemas em que eles são considerados como variáveis de projecto. Também aparecem restrições deste tipo nos casos em que se pretende fixar algumas variáveis de projecto em problemas que foram inicialmente formulados com todas as variáveis livres.

Devem também ser substituídas as restrições do tipo

$$x_i = x_j \quad (2.8)$$

Estas restrições são necessárias nos casos em que se pretende uma solução simétrica ou quando se reduz a heterogeneidade da solução recorrendo ao agrupamento de variáveis (variable linking).

Nos problemas em que figuram áreas e momentos de inércia de secções transversais de barras é necessário estabelecer relações do tipo

$$I_i = p a_i^q \quad (2.9)$$

Os valores p e q são reais e dependem da forma da secção transversal. No caso dos perfis metálicos standardizados a expressão (2.9) não é exacta, podendo no entanto constituir uma aproximação aceitável.

A formulação do comportamento da estrutura por intermédio do método dos deslocamentos implica a inclusão no programa matemático de tantas restrições igualdade quantas as equações de equilíbrio dos nós. Estas restrições relacionam os deslocamentos dos nós com as restantes variáveis de projecto. Uma vez que o número de equações de equilíbrio é igual ao número de deslocamentos seria possível explicitá-los e substituí-los nas restantes expressões do programa matemático. Exceptuando casos muito simples, este procedimento é impraticável porque implica um elevado número de operações e o armazenamento de um grande volume de informação. Existe ainda a agravante de as operações efectuadas sobre este tipo de informação não serem cálculos numéricos mas sim substituições e simplificações de expressões. No Capítulo 6 será apresentado um método designado análise de sensibilidades que permite considerar de um modo implícito os deslocamentos dos nós e as equações de equilíbrio. Nos problemas de optimização resolvidos no âmbito deste trabalho as equações de equilíbrio foram consideradas explicitamente como restrições do tipo (2.2b) ou (2.2c). A

inclusão das equações de equilíbrio como restrições desigualdade foi efectuada recorrendo à seguinte equivalência

$$h(\tilde{x}) = 0 \Leftrightarrow \begin{cases} h(\tilde{x}) \leq 0 \\ -h(\tilde{x}) \geq 0 \end{cases} \quad (2.10)$$

Esta substituição de cada restrição igualdade por um par de restrições desigualdade apresenta algumas vantagens, que serão referidas no Capítulo 6. A seguinte equivalência permite a substituição de uma restrição igualdade por apenas uma restrição desigualdade

$$h(\tilde{x}) = 0 \Leftrightarrow [h(\tilde{x})]^2 \leq 0 \quad (2.11)$$

Quando esta substituição é efectuada, o programa matemático (2.2) torna-se muito difícil de resolver [Bel82].

De um modo geral é mais simples formular o programa matemático relativo a um problema de optimização de estruturas se se considerarem todos os elementos finitos desligados entre si. Para que esta formulação traduza a realidade é necessário acrescentar um conjunto de restrições igualdade do seguinte tipo

$$d_{ij} = d_{kl} \quad (2.12)$$

A expressão (2.12) significa a igualdade entre o deslocamento segundo o grau de liberdade j do elemento i e o deslocamento segundo o grau de liberdade l do elemento k . A substituição deste tipo de restrições nas restantes expressões do programa matemático constitui uma operação que corresponde ao convencional espalhamento das matrizes de rigidez dos elementos.

Quando num programa matemático estão presentes duas restrições igualdade

$$\begin{aligned} h_1(\tilde{x}) &= 0 \\ h_2(\tilde{x}) &= 0 \end{aligned} \quad (2.13)$$

está implícito que na solução se deve verificar ser $h_1 = 0$ e $h_2 = 0$. Sempre que a formulação de um problema exija que na solução se verifique ser $h_1 = 0$ ou $h_2 = 0$ no programa matemático deve estar presente a seguinte restrição

$$h_1(\tilde{x}) h_2(\tilde{x}) = 0 \quad (2.14)$$

A expressão (2.14) designa-se restrição de complementaridade e garante que as funções h_1 ou h_2 ou ambas serão nulas na solução óptima. Contudo, permanece a possibilidade de uma delas ser não nula. Nos problemas de optimização de estruturas existe a necessidade de considerar restrições de complementaridade quando estão presentes apoios unilaterais. Neste caso h_1 representa a reacção de apoio e h_2 o deslocamento do nó. Quando o comportamento da estrutura é elastoplástico existe uma complementaridade entre a deformação plástica e a distância à superfície de cedência [Ada80].

2.1.4 - Restrições desigualdade

Alguns métodos de optimização requerem que as restrições desigualdade do tipo (2.2b) sejam convertidas em restrições igualdade. Esta conversão pode ser efectuada adicionando a $g(\underline{x})$ uma função $F(s)$ que tem de ser sempre positiva

$$g(\underline{x}) \leq 0 \rightarrow g(\underline{x}) + F(s) = 0 \quad (2.15)$$

A cada restrição g é adicionada uma função F que depende de uma variável s . A função F representa a folga da restrição g e a variável s é designada variável de desvio (slack variable). Esta nova variável é acrescentada às restantes variáveis do programa matemático. O valor das variáveis de desvio só será conhecido após a resolução do problema de optimização. Se o valor final de $F(s)$ for nulo a restrição desigualdade é designada activa. Se $F(s)$ for maior que zero, a restrição desigualdade é inactiva e a sua supressão não afecta a solução do problema de optimização.

Existem diversas alternativas para a função $F(s)$, sendo as mais comuns as seguintes

$$F(s) = s \quad (2.16)$$

$$F(s) = |s| \quad (2.17)$$

$$F(s) = s^2 \quad (2.18)$$

A função (2.16) só pode ser utilizada se houver a garantia de a variável s só assumir valores positivos. Quando o programa matemático é linear e é utilizado o método simplex na sua resolução todas as variáveis são positivas, sendo portanto possível utilizar (2.16). A função (2.17) não deve ser utilizada porque provoca descontinuidades no problema que

podem dificultar a sua resolução. Na programação não linear é em geral preferível adoptar a função (2.18). No presente trabalho foi sempre esta a função utilizada.

Nos problemas de optimização as restrições desigualdade são devidas ao facto de ser necessário limitar tensões, deslocamentos, frequências próprias de vibração, etc. É também aconselhável limitar os valores das próprias variáveis de projecto para evitar o aparecimento de soluções impossíveis (e.g., áreas negativas) ou indesejadas (e.g., forma muito diferente da correspondente à solução inicial). Estas restrições apresentam-se com a seguinte forma

$$-x_i + x_{\min} \leq 0 \quad (2.19)$$

$$x_i - x_{\max} \leq 0 \quad (2.20)$$

e são designadas simple bounds [Gil81] ou side constraints [Van84].

2.2 - CONDIÇÕES DE OPTIMALIDADE

Na generalidade dos casos os problemas de optimização são formulados como um programa matemático constituído por funções não lineares contínuas e de derivadas também contínuas. Nestas circunstâncias e de acordo com o que foi atrás exposto, a sua forma genérica é a seguinte

$$\text{Min. } f(\underline{x}) \quad \underline{x} = (x_1, \dots, x_n) \quad (2.21a)$$

s.a

$$g_j(\underline{x}) \leq 0 \quad (j = 1, \dots, m) \quad (2.21b)$$

$$h_k(\underline{x}) = 0 \quad (k = 1, \dots, l) \quad (2.21c)$$

Alguns métodos destinados à resolução de programas matemáticos procuram obter directamente a solução de (2.21). Outros métodos procuram uma solução que satisfaça um conjunto de condições designadas condições de optimalidade. Neste caso é necessário demonstrar que uma solução que verifique as condições de optimalidade é a solução do programa matemático (2.21). Qualquer que seja o método de resolução utilizado, é sempre importante verificar se a solução obtida satisfaz ou não as condições de optimalidade. O

estudo efectuado com base na solução fornecida por um método de resolução de (2.21) designa-se genericamente análise pós-optimização ou análise de sensibilidades da solução óptima. A análise pós-optimização permite uma avaliação da qualidade da solução obtida e consiste também no cálculo da influência no custo final da variação de certos parâmetros que foram considerados com valor fixo [Van84]. A análise pós-optimização fornece respostas para, por exemplo, as seguintes questões:

- quanto aumenta o custo da estrutura se o deslocamento admissível diminuir Δu ?
- quanto diminui o custo da estrutura se a tensão admissível no material aumentar $\Delta \sigma$?

É evidente que o objectivo da análise pós-optimização consiste em obter respostas para este tipo de questões sem ter de repetir a resolução do problema de optimização. A análise pós-optimização pode ser efectuada recorrendo às condições de optimalidade.

Apresenta-se em seguida uma classificação das condições de optimalidade baseada no facto de serem necessárias ou suficientes e de se referirem ao mínimo global ou a um mínimo local:

- A) condições necessárias para que uma solução seja um mínimo local ou global;
- B) condições suficientes para que uma solução seja um mínimo local;
- C) condições suficientes para que uma solução seja um mínimo global.

Uma solução do programa matemático (2.21) é um mínimo global quando é admissível e não existe nenhuma outra solução admissível à qual corresponda um valor inferior da função objectivo f . Esta é portanto a verdadeira solução do programa matemático. Quando uma solução é admissível e numa sua vizinhança não existe nenhuma outra solução admissível à qual corresponda um valor inferior da função objectivo f , está-se na presença de um mínimo local, que poderá eventualmente ser um mínimo global.

As condições suficientes para que uma solução seja um mínimo global (C) só podem ser estabelecidas se a função objectivo e a região admissível apresentarem determinadas características relacionadas com o conceito de convexidade [Bel82]. Uma vez que na generalidade dos casos os problemas de optimização de estruturas não apresentam estas características, as condições de optimalidade do tipo C) apenas serão referidas em casos muito simples. A verificação das condições suficientes para que uma solução seja um mínimo local (B) obriga nuns casos a verificar se a matriz Hessiana da função objectivo é positiva definida e noutros casos a verificar as características de um elevado número de direcções admissíveis [Bel82]. Quer num caso quer noutro, este procedimento revela-se pouco prático, não sendo

habitualmente tido em consideração nos problemas de optimização de estruturas. De um modo geral a resolução deste tipo de problemas envolve apenas a procura de soluções que verifiquem algumas das condições necessárias relativas a mínimos locais (A).

A formulação das condições de optimalidade necessárias e suficientes constituiu em muitos casos a base de desenvolvimento de importantes métodos de optimização. Estes métodos são geralmente processos iterativos que partem de uma solução inicial e que a modificam de modo a que ela venha a satisfazer as condições de optimalidade. Quando se utilizam métodos que apenas garantem a verificação de algumas das condições necessárias relativas a mínimos locais (A) é conveniente repetir o processo iterativo a partir de distintas soluções iniciais. Deste modo aumenta-se a probabilidade de obter o mínimo global ou pelo menos um mínimo local. Se se obtiverem diversas soluções distintas, deve ser considerada apenas aquela que apresentar um menor valor da função objectivo.

Alguns problemas de optimização apresentam múltiplas soluções com o mesmo valor da função objectivo. Estas soluções podem ser múltiplos mínimos locais ou múltiplos mínimos globais. As condições de optimalidade podem ser formuladas de modo a distinguir soluções isoladas de soluções múltiplas [Gil81]. Nos problemas de optimização de estruturas é satisfatório determinar uma solução de custo mínimo, independentemente de existirem ou não outras soluções com um custo igual. Por este motivo as questões relativas à distinção entre soluções isoladas e soluções múltiplas não serão aqui abordadas.

2.2.1 - Função com uma variável e sem restrições

A minimização sem restrições de uma função que apenas depende de uma variável é um caso muito simples, mas que serve para introduzir alguns conceitos fundamentais. O programa matemático relativo a este problema apresenta a seguinte formulação

$$\text{Min. } f(x) \quad (2.22)$$

Designando a solução óptima por x^* , as condições de optimalidade são neste caso [Gil81]

A) condições necessárias mas não suficientes para que x^* seja um mínimo local ou global

$$f'(x^*) = 0 \quad (2.23a)$$

$$f''(x^*) \geq 0 \quad (2.23b)$$

B) condições suficientes mas não necessárias para que x^* seja um mínimo local

$$f'(x^*) = 0 \quad (2.24a)$$

$$f''(x^*) > 0 \quad (2.24b)$$

C) condições suficientes mas não necessárias para que x^* seja um mínimo global

$$f'(x^*) = 0 \quad (2.25a)$$

$$f''(x^*) \geq 0, \forall x \Leftrightarrow f \text{ é convexa} \quad (2.25b)$$

Uma vez que é possível representar graficamente o problema (2.22), as condições de optimalidade revelam-se neste caso intuitivas. As respectivas demonstrações encontram-se em [Gil81]. Na Figura 2.1 está representada uma função que possui diversos pontos com a primeira derivada nula.

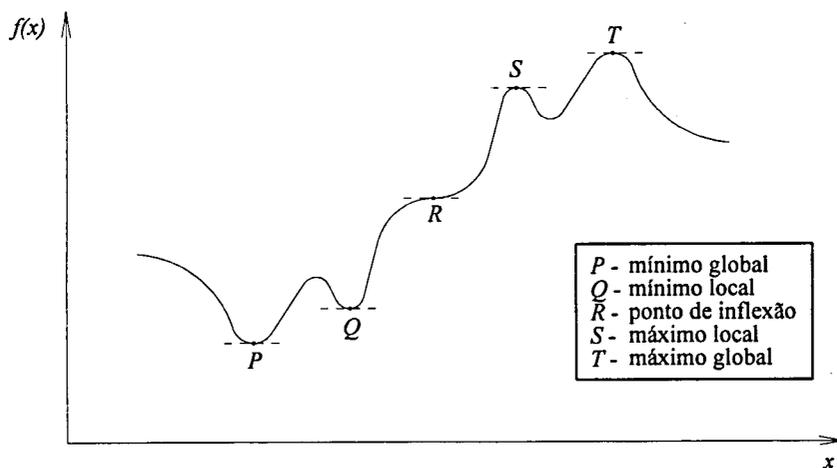


Figura 2.1 - Função com uma variável - pontos com primeira derivada nula.

Os pontos P e Q verificam as condições de optimalidade B), enquanto que R apenas verifica as condições necessárias A). Apesar de satisfazer estas condições, o ponto R não é mínimo nem máximo local. As considerações relativas aos pontos S e T são semelhantes. Alguns métodos de otimização apenas procuram um ponto que satisfaça $f'(x^*) = 0$, podendo portanto convergir para qualquer um dos pontos indicados na Figura 2.1. As condições de

optimalidade C) só são satisfeitas em situações particulares simples, uma vez que na generalidade dos casos não existe convexidade.

2.2.2 - Função com n variáveis e sem restrições

A minimização sem restrições de funções de n variáveis apresenta muitas semelhanças com o caso unidimensional. A respectiva formulação é a seguinte

$$\text{Min. } f(x_1, \dots, x_n) \quad (2.26)$$

Nas condições de optimalidade são referidos o vector gradiente de f e a matriz Hessiana de f cuja definição é a seguinte

$$\nabla f(\underline{x}) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right) \quad (2.27)$$

$$H_{ij}(\underline{x}) = \frac{\partial^2 f}{\partial x_i \partial x_j} \quad (2.28)$$

Se a função f e as respectivas derivadas forem contínuas a matriz Hessiana é simétrica.

As condições de optimalidade apresentam-se como uma generalização do caso unidimensional [Gil81] [Van84].

- A) condições necessárias mas não suficientes para que \underline{x}^* seja um mínimo local ou global

$$\nabla f(\underline{x}^*) = \underline{0} \quad (2.29a)$$

$$H(\underline{x}^*) \text{ positiva semidefinida ou positiva definida} \quad (2.29b)$$

- B) condições suficientes mas não necessárias para que \underline{x}^* seja um mínimo local

$$\nabla f(\underline{x}^*) = \underline{0} \quad (2.30a)$$

$$H(\underline{x}^*) \text{ positiva definida} \quad (2.30b)$$

C) condições suficientes mas não necessárias para que \tilde{x}^* seja um mínimo global

$$\nabla f\left(\tilde{x}^*\right) = 0 \quad (2.31a)$$

$$H\left(\tilde{x}\right) \text{ positiva semidefinida ou positiva definida, } \forall \tilde{x} \Leftrightarrow f \text{ é convexa} \quad (2.31b)$$

Uma matriz é positiva definida quando todos os seus valores próprios são maiores do que zero, é positiva semidefinida quando todos os seus valores próprios são não negativos e pelo menos um é nulo e é indefinida quando possui alguns valores próprios maiores do que zero e outros menores do que zero [Aro89]. A definição de matriz negativa definida e negativa semidefinida é semelhante à de matriz positiva definida e positiva semidefinida, bastando considerar uma troca de sinal.

As condições de optimalidade A), B) e C) têm em comum o anulamento do gradiente da função objectivo. Os pontos em que tal se verifica são designados pontos estacionários de f . A diferença entre as condições de optimalidade B) e C) consiste na exigência de convexidade da função objectivo $f(\tilde{x})$. Quando esta função é convexa um mínimo local é também mínimo global. As condições de optimalidade C) contemplam a possibilidade de numa vizinhança de \tilde{x}^* existirem múltiplos mínimos com um valor da função objectivo igual ao mínimo global. Em problemas práticos raramente se consegue demonstrar a existência de convexidade, sendo portanto difícil saber se a solução obtida é o mínimo global ou apenas um mínimo local.

Nas Figuras 2.2 e 2.3 encontra-se representada graficamente a seguinte função polinomial

$$\begin{aligned} f(x_1, x_2) = & -0.13x_1^3 x_2^3 - 0.03x_1^3 x_2^2 + 0.65x_1^3 x_2 - 0.04x_1^2 x_2^2 \\ & + 0.65x_1 x_2^3 + 0.16x_1^3 + 0.16x_1 x_2^2 + 0.17x_1^2 \\ & - 2.84x_1 x_2 + 0.17x_2^2 - 0.66x_1 - 0.04x_2 - 0.70 \end{aligned} \quad (2.32)$$

No domínio representado esta função apresenta o gradiente nulo em cinco pontos. No Quadro 2.2 encontram-se os valores de x_1 e x_2 correspondentes aos cinco pontos estacionários, bem como as suas características.

Quadro 2.2 - Pontos estacionários de (2.32).

Ponto	x_1	x_2	$f(x_1, x_2)$	Observações
P	1.109642	1.041041	-2.510023	Mínimo global * (1)
Q	-1.104724	-1.177210	-1.729227	Mínimo local (1)
R	-0.043018	-0.237351	-0.681238	Ponto de sela (2)
S	1.249587	-1.334671	1.265513	Máximo local (3)
T	-1.225442	1.140193	1.762034	Máximo global * (3)

(1) Matriz Hessiana positiva definida.

(2) Matriz Hessiana indefinida.

(3) Matriz Hessiana negativa definida.

* Considerando apenas o domínio representado nas Figuras 2.2 e 2.3.

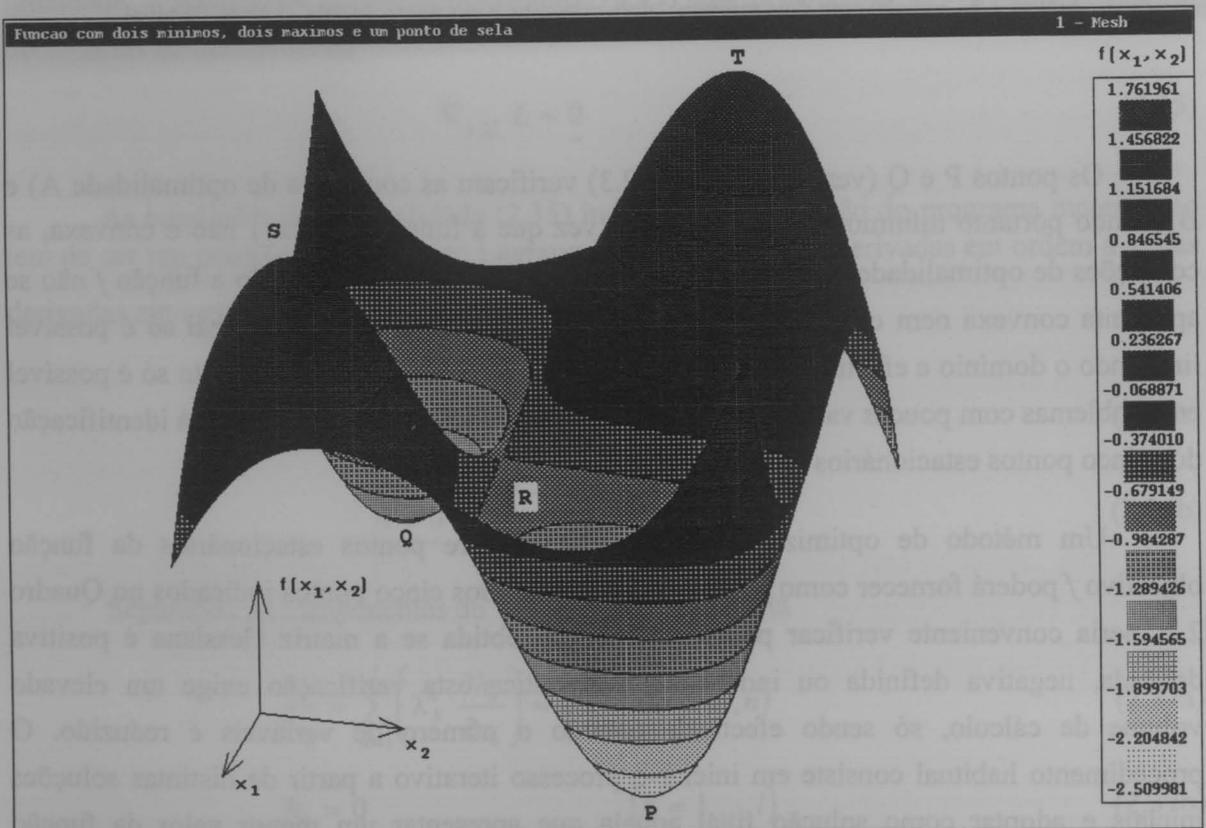


Figura 2.2 - Função com dois mínimos, dois máximos e um ponto de sela - perspectiva das curvas de nível.

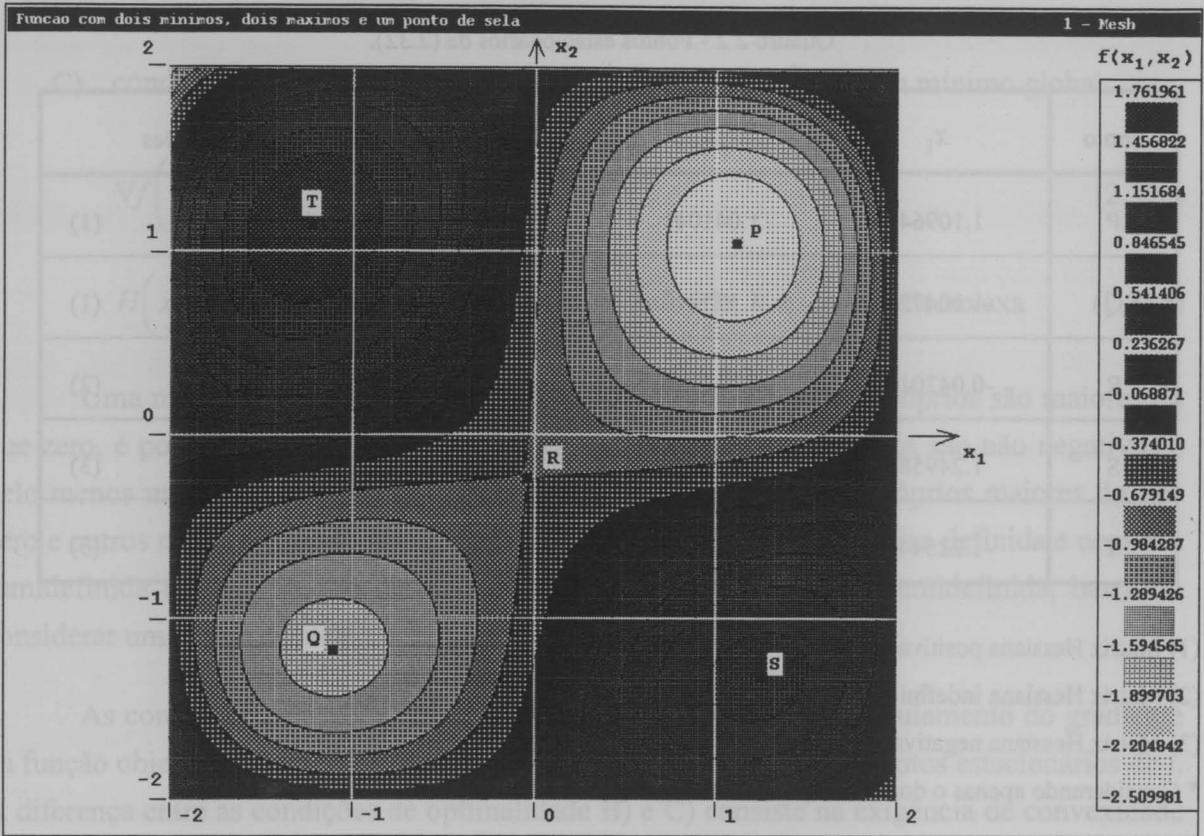


Figura 2.3 - Função com dois mínimos, dois máximos e um ponto de sela - curvas de nível.

Os pontos P e Q (ver Figuras 2.2 e 2.3) verificam as condições de optimalidade A) e B), sendo portanto mínimos locais de f . Uma vez que a função $f(x_1, x_2)$ não é convexa, as condições de optimalidade C) não se verificam em nenhum ponto. Quando a função f não se apresenta convexa nem côncava, a identificação do mínimo e máximo global só é possível limitando o domínio e efectuando aí uma pesquisa exhaustiva. Este procedimento só é possível em problemas com poucas variáveis, tendo no caso da função (2.32) conduzido à identificação dos cinco pontos estacionários indicados nas Figuras 2.2 e 2.3.

Um método de optimização que apenas procure pontos estacionários da função objectivo f poderá fornecer como solução qualquer um dos cinco pontos indicados no Quadro 2.2. Seria conveniente verificar para cada solução obtida se a matriz Hessiana é positiva definida, negativa definida ou indefinida. Na prática esta verificação exige um elevado volume de cálculo, só sendo efectuada quando o número de variáveis é reduzido. O procedimento habitual consiste em iniciar o processo iterativo a partir de distintas soluções iniciais e adoptar como solução final aquela que apresentar um menor valor da função objectivo.

2.2.3 - Restrições igualdade - Lagrangeano

A minimização de uma função sujeita a um conjunto de restrições igualdade é representada pelo seguinte programa matemático

$$\text{Min. } f(\underline{x}) \quad \underline{x} = (x_1, \dots, x_n) \quad (2.33a)$$

s.a

$$h_k(\underline{x}) = 0 \quad (k = 1, \dots, l) \quad (2.33b)$$

A resolução do programa matemático (2.33) pode ser efectuada recorrendo a uma função designada Lagrangeano, cuja definição é a seguinte [Lue84]

$$L(\underline{x}, \underline{\lambda}^h) = f(\underline{x}) + \sum_{k=1}^l [\lambda_k^h h_k(\underline{x})] \quad (2.34)$$

Em (2.34) $\underline{\lambda}^h$ é o vector dos multiplicadores de Lagrange associados a restrições igualdade. A solução do programa matemático (2.33) tem de satisfazer as seguintes condições necessárias de optimalidade

$$\nabla_{\underline{x}, \underline{\lambda}^h} L = \underline{0} \quad (2.35)$$

As condições de optimalidade (2.35) indicam que a solução do programa matemático tem de ser um ponto estacionário do Lagrangeano. Separando as derivadas em ordem a \underline{x} das derivadas em ordem a $\underline{\lambda}^h$ obtém-se

$$\nabla_{\underline{x}} f + \sum_{k=1}^l (\lambda_k^h \nabla_{\underline{x}} h_k) = \underline{0} \quad (2.36a)$$

$$\underline{h} = \underline{0} \quad (2.36b)$$

Separando as componentes do vector gradiente resulta

$$\frac{\partial f}{\partial x_i} + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial h_k}{\partial x_i} \right) = 0 \quad (i = 1, \dots, n) \quad (2.37a)$$

$$h_k = 0 \quad (k = 1, \dots, l) \quad (2.37b)$$

As expressões (2.37) correspondem a um sistema de $n + l$ equações não lineares a $n + l$ incógnitas, cuja solução são as n componentes de \underline{x}^* e as l componentes do vector $\underline{\lambda}^h$ associado à solução \underline{x}^* . Uma vez que (2.37) são condições necessárias mas não suficientes de optimalidade, a respectiva solução pode ser um mínimo local, ponto de sela ou máximo local. Por este motivo a solução de (2.37) é designada solução candidata a mínimo global [Aro89].

Para que seja possível efectuar a representação gráfica do Lagrangeano, é necessário considerar um problema com apenas uma variável e uma restrição igualdade. O seguinte programa matemático possui essas características.

$$\text{Min. } f(x) = x^2 - 2x + 4 \quad (2.38a)$$

s.a

$$h(x) = -x + 2 = 0 \quad (2.38b)$$

O Lagrangeano correspondente a (2.38) é a seguinte função

$$L(x, \lambda^h) = (x^2 - 2x + 4) + \lambda^h(-x + 2) \quad (2.39)$$

Uma vez que (2.39) é uma função de duas variáveis, é possível visualizá-la graficamente recorrendo a curvas de nível (ver Figura 2.4). A região admissível do programa matemático (2.38) é apenas o ponto $x = 2$, que é forçosamente a solução óptima. O valor do único multiplicador de Lagrange pode ser obtido de (2.37a), sendo neste caso $\lambda^h = 2$. Na Figura 2.4 pode-se verificar que o ponto $(\underline{x}^*, \underline{\lambda}^h) = (2, 2)$ é um ponto de sela, ao qual corresponde a seguinte matriz Hessiana do Lagrangeano

$$\underline{H}(\underline{x}^*, \underline{\lambda}^h) = \begin{bmatrix} 2 & -1 \\ -1 & 0 \end{bmatrix} \quad (2.40)$$

Esta matriz possui um valor próprio positivo e o outro negativo, sendo portanto indefinida. Em problemas com múltiplas variáveis e múltiplas restrições igualdade a matriz Hessiana correspondente à solução óptima $\underline{H}(\underline{x}^*, \underline{\lambda}^h)$ é também indefinida [Lue69].

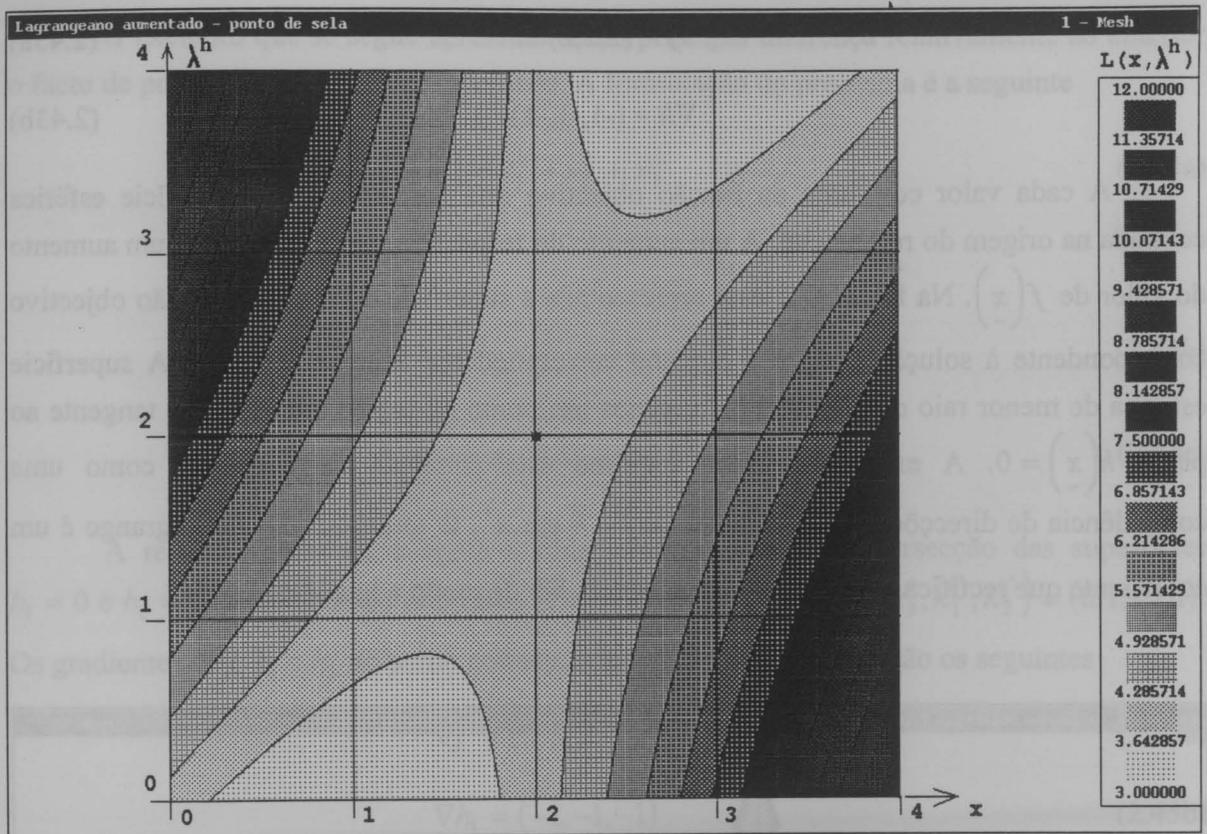


Figura 2.4 - Representação gráfica do Lagrangeano (2.39).

Apresentam-se em seguida alguns exemplos simples, cuja visualização gráfica esclarece alguns aspectos relativos à necessidade de anular o gradiente do Lagrangeano para obter a solução de um problema de minimização com restrições. O primeiro exemplo consiste na minimização de uma função de três variáveis sujeita a uma restrição igualdade.

$$\text{Min. } f(\underline{x}) = x_1^2 + x_2^2 + x_3^2 \tag{2.41a}$$

s.a

$$h(\underline{x}) = -x_1 - x_2 - x_3 + 3 = 0 \tag{2.41b}$$

As condições de optimalidade (2.36a) reduzem-se neste caso à seguinte expressão

$$\nabla f = -\lambda^h \nabla h \tag{2.42}$$

A resolução do sistema de equações (2.36) conduz à solução ótima $(x_1^*, x_2^*, x_3^*, \lambda^h) = (1, 1, 1, 2)$. Neste ponto os gradientes de f e h são

$$\nabla f = (2, 2, 2) \quad (2.43a)$$

$$\nabla h = (-1, -1, -1) \quad (2.43b)$$

A cada valor constante da função objectivo está associada uma superfície esférica centrada na origem do referencial. A um aumento do respectivo raio corresponde um aumento do valor de $f(\tilde{x})$. Na Figura 2.5 está representada a superfície de nível da função objectivo correspondente à solução óptima e o plano correspondente à única restrição. A superfície esférica de menor raio que contém pelo menos um ponto da região admissível é tangente ao plano $h(\tilde{x}) = 0$. A expressão (2.42) exprime esta condição de tangência como uma coincidência de direcções dos gradientes de f e h em \tilde{x}^* . O multiplicador de Lagrange é um coeficiente que rectifica o sentido e a grandeza de ∇h de modo a igualá-lo a $-\nabla f$.

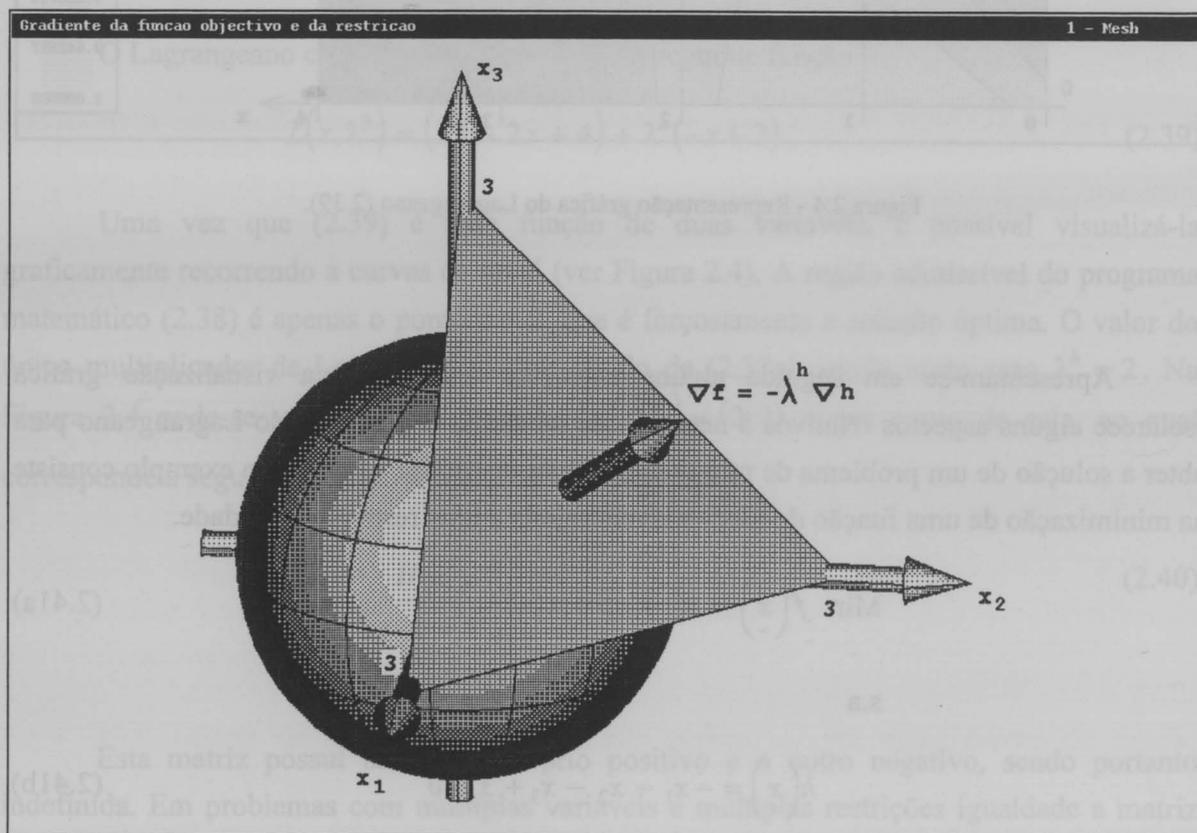


Figura 2.5 - Visualização do significado do anulamento do gradiente do Lagrangeano.

O exemplo que se segue apresenta como principal diferença relativamente ao anterior o facto de possuir duas restrições igualdade. A formulação do problema é a seguinte

$$\text{Min. } f(\tilde{x}) = x_1^2 + x_2^2 + x_3^2 \quad (2.44a)$$

s.a

$$h_1(\tilde{x}) = -x_1 - x_2 - x_3 + 2 = 0 \quad (2.44b)$$

$$h_2(\tilde{x}) = x_1 - x_2 - x_3 + 2 = 0 \quad (2.44c)$$

A região admissível é agora a linha correspondente à intersecção das superfícies $h_1 = 0$ e $h_2 = 0$. A solução do programa matemático (2.44) é $(x_1^*, x_2^*, x_3^*, \lambda_1^*, \lambda_2^*) = (0, 1, 1, 1, 1)$. Os gradientes de f , h_1 e h_2 no ponto correspondente à solução óptima são os seguintes

$$\nabla f = (0, 2, 2) \quad (2.45a)$$

$$\nabla h_1 = (-1, -1, -1) \quad (2.45b)$$

$$\nabla h_2 = (1, -1, -1) \quad (2.45c)$$

As condições de optimalidade (2.36a) apresentam-se neste caso com a seguinte forma

$$\nabla f = -\lambda_1^h \nabla h_1 - \lambda_2^h \nabla h_2 \quad (2.46)$$

Na Figura 2.6 encontra-se a representação gráfica das superfícies e vectores correspondentes a este problema. A resolução do programa matemático (2.44) consiste na determinação da superfície esférica de menor raio que contém pelo menos um ponto da região admissível. Neste caso a solução é o ponto de tangência entre a referida superfície esférica e a linha correspondente à região admissível. De acordo com a Figura 2.6, a condição de optimalidade (2.46) pode ser interpretada como a imposição de que ∇f pertença a um plano normal à linha correspondente à região admissível. Este plano é definido pelo conjunto das combinações lineares de ∇h_1 e ∇h_2 . Deste modo ∇f é normal à linha correspondente à região admissível no ponto de tangência com a superfície de nível da função objectivo.

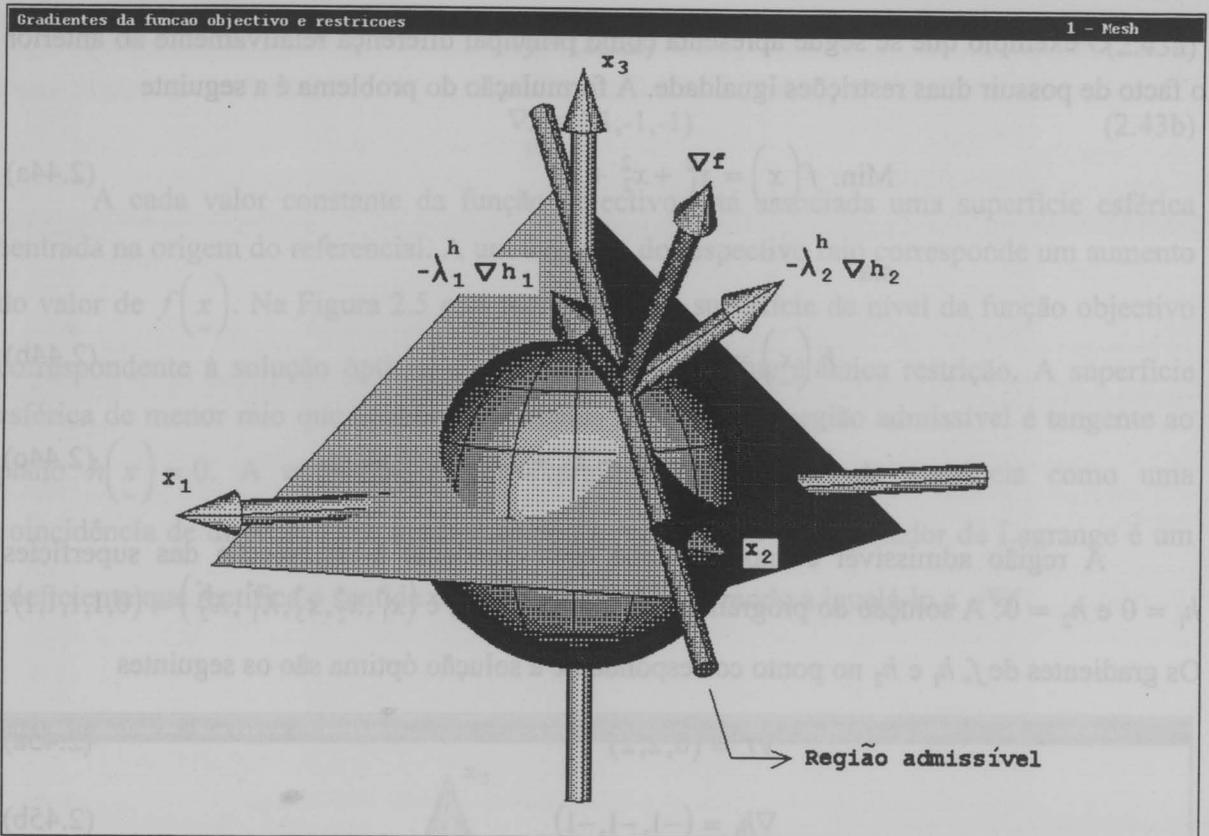


Figura 2.6 - Visualização do significado do anulamento do gradiente do Lagrangeano.

Apresentam-se em seguida algumas observações relativas aos multiplicadores de Lagrange associados a restrições igualdade (λ^h)

- A) Multiplicando por -1 ambos os membros de uma restrição $h = 0$ obtém-se a expressão equivalente $-h = 0$. O respectivo gradiente passa a ser $-\nabla h$ e de acordo com (2.36a) o correspondente multiplicador de Lagrange λ^h passará a ter na solução óptima o valor $-\lambda^h$. Verifica-se portanto que os multiplicadores de Lagrange associados a restrições igualdade podem ser positivos ou negativos. O respectivo sinal depende do modo como a restrição é formulada ($h = 0$ ou $-h = 0$).
- B) Se na solução óptima um multiplicador de Lagrange apresentar um valor nulo, a respectiva restrição pode em geral ser retirada do programa matemático. Se o método de optimização apenas procurar resolver o sistema de equações (2.36), a supressão de uma restrição com λ^h nulo não afecta a respectiva solução. Existem no entanto alguns problemas particulares cujo mínimo global pode ser condicionado por uma restrição com λ^h nulo. Apresenta-se como exemplo o seguinte programa matemático

$$\text{Min. } f(\underline{x}) = x_1 x_2 \quad (2.47a)$$

s.a

$$h(\underline{x}) = x_1 - x_2 = 0 \quad (2.47b)$$

A solução de (2.47) é $(x_1^*, x_2^*, \lambda^h) = (0, 0, 0)$. Se se suprimir a restrição, o mínimo global passa a ser $-\infty$ e o ponto $(x_1, x_2) = (0, 0)$ passa a ser um ponto de sela da função objectivo.

2.2.4 - Restrições desigualdade

As restrições desigualdade que figuram no programa matemático (2.21) podem ser convertidas em restrições igualdade recorrendo a uma variável de desvio. No caso da programação não linear deve ser efectuada a seguinte substituição

$$g_j(\underline{x}) \leq 0 \rightarrow g_j(\underline{x}) + s_j^2 = 0 \quad (j = 1, \dots, m) \quad (2.48)$$

Com este procedimento são acrescentadas ao programa matemático m variáveis. Os multiplicadores de Lagrange associados a restrições desigualdade são designados λ^g para que possam ser distinguidos dos multiplicadores λ^h referidos na secção anterior. Por motivos que só serão apresentados no Capítulo 4, as variáveis que figuram no Lagrangeano consideram-se ordenadas de um modo diferente daquele que seria mais lógico. Com a inclusão das restrições (2.48) o Lagrangeano (2.34) passa a ser o seguinte

$$L(\underline{s}, \underline{\lambda}^g, \underline{x}, \underline{\lambda}^h) = f(\underline{x}) + \sum_{j=1}^m \left[\lambda_j^g (g_j(\underline{x}) + s_j^2) \right] + \sum_{k=1}^l \left[\lambda_k^h h_k(\underline{x}) \right] \quad (2.49)$$

As condições de optimalidade que resultam do anulamento do gradiente do Lagrangeano incluem derivadas parciais em ordem aos quatro tipos de variáveis $(\underline{s}, \underline{\lambda}^g, \underline{x}, \underline{\lambda}^h)$. Ao efectuar a derivação do Lagrangeano por esta ordem obtém-se o seguinte sistema de equações não lineares

$$2s_j \lambda_j^g = 0 \quad (j = 1, \dots, m) \quad (2.50a)$$

$$g_j + s_j^2 = 0 \quad (j = 1, \dots, m) \quad (2.50b)$$

$$\frac{\partial f}{\partial x_i} + \sum_{j=1}^m \left(\lambda_j^g \frac{\partial g_j}{\partial x_i} \right) + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial h_k}{\partial x_i} \right) = 0 \quad (i = 1, \dots, n) \quad (2.50c)$$

$$h_k = 0 \quad (k = 1, \dots, l) \quad (2.50d)$$

Em (2.50) figuram $2m + n + l$ equações e igual número de incógnitas. A solução de (2.50) pode ser obtida recorrendo a um método de resolução de sistemas de equações não lineares.

Das expressões que figuram em (2.50) é possível extrair algumas conclusões. Assim, quando na solução óptima uma variável s_j^* apresenta um valor nulo, a respectiva restrição desigualdade encontra-se activa. Quando o valor de s_j^* for não nulo, a restrição é inactiva e a equação de complementaridade (2.50a) obriga λ_j^g a adoptar um valor nulo. Conclui-se portanto que os multiplicadores de Lagrange associados a restrições desigualdade inactivas são nulos. Se numa solução de (2.50) se substituir s_j^* por $-s_j^*$, obtém-se uma solução alternativa do sistema de equações não lineares. Uma vez que as equações (2.50a) e (2.50b) são respeitadas quer por s_j^* como por $-s_j^*$, conclui-se que o sinal da variável de desvio é arbitrário.

A resolução de (2.50) tem de um modo geral que ser efectuada recorrendo a um método iterativo. Em certos casos particulares é possível resolver o sistema de equações por substituição, desde que sejam consideradas todas as hipóteses de complementaridade impostas pelas equações (2.50a). Uma vez que estas equações apenas impõe o anulamento de s_j ou λ_j^g , existem 2^m modos diferentes de elas serem respeitadas. Em problemas com poucas restrições é possível considerar separadamente cada uma destas hipóteses e em seguida resolver o sistema de equações constituído pelas restrições desigualdade consideradas activas, pelas equações (2.50c) e pelas restrições igualdade. Apresenta-se em seguida um programa matemático cuja resolução foi efectuada por este processo

$$\text{Min. } f(\underline{x}) = x_1^2 + x_2^2 \quad (2.51a)$$

s.a

$$g_1(\underline{x}) = -x_1 - x_2 + 4 \leq 0 \quad (2.51b)$$

$$g_2(\underline{x}) = -x_2 + 1 \leq 0 \quad (2.51c)$$

O sistema de equações não lineares (2.50) é neste caso o seguinte

$$2s_1\lambda_1^g = 0 \quad (2.52a)$$

$$2s_2\lambda_2^g = 0 \quad (2.52b)$$

$$-x_1 - x_2 + 4 + s_1^2 = 0 \quad (2.52c)$$

$$-x_2 + 1 + s_2^2 = 0 \quad (2.52d)$$

$$2x_1 - \lambda_1^g = 0 \quad (2.52e)$$

$$2x_2 - \lambda_1^g - \lambda_2^g = 0 \quad (2.52f)$$

No Quadro 2.3 encontram-se as soluções de (2.52) que correspondem às $2^2 = 4$ possibilidades de satisfazer as equações (2.52a) e (2.52b).

Quadro 2.3 - Soluções do sistema de equações não lineares (2.52).

Solução	s_1^2	s_2^2	λ_1^g	λ_2^g	x_1	x_2	$f(\underline{x})$	Observações
A	0	0	6	-4	3	1	10	λ_2^g negativo
B	0	1	4	0	2	2	8	Solução óptima
C	-3	0	0	2	0	1	1	Impossível
D	-4	-1	0	0	0	0	0	Impossível

Na Figura 2.7 encontra-se representado graficamente o programa matemático (2.51), as soluções do sistema de equações (2.52) e os gradientes da função objectivo e das restrições nos pontos A e B.

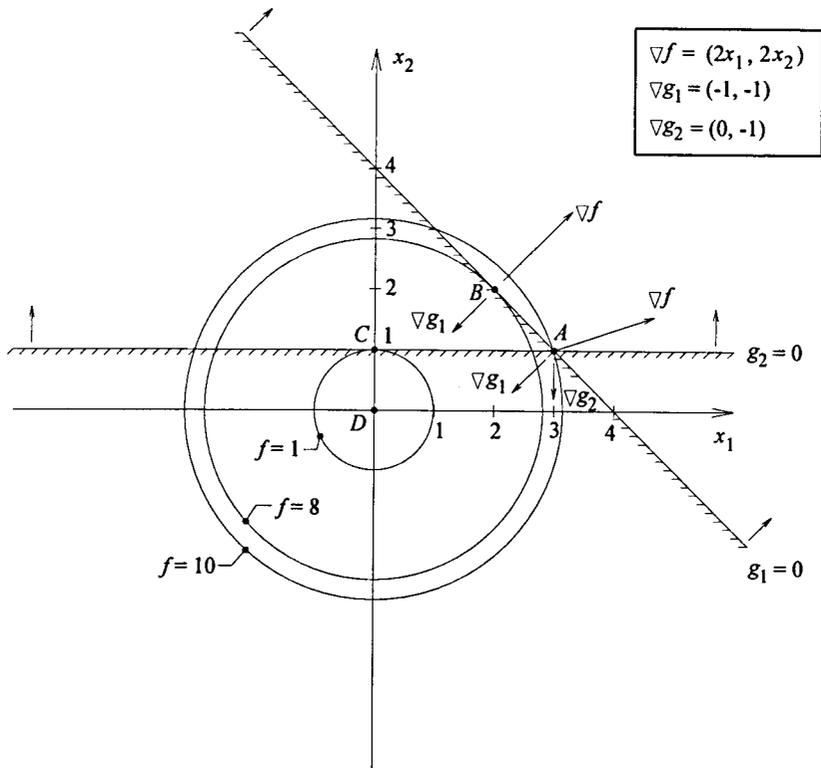


Figura 2.7 - Representação gráfica do programa matemático (2.51).

Os pontos C e D encontram-se fora da região admissível, implicando este facto a presença de pelo menos um s_j^2 negativo. Por este motivo as soluções C e D não devem ser consideradas. As soluções A e B são ambas admissíveis e a cada uma delas corresponde um valor distinto da função objectivo. A solução óptima é o ponto B porque apresenta um valor inferior de $f(\underline{x})$. De acordo com (2.50c), a solução óptima deve respeitar a seguinte condição

$$\nabla f + \sum_{j=1}^m (\lambda_j^g \nabla g_j) = \underline{0} \quad (2.53)$$

No ponto B (ver Figura 2.7) esta condição é satisfeita com um valor positivo de λ_1^g e com um valor nulo de λ_2^g , sendo a restrição g_1 activa e a restrição g_2 inactiva. No ponto A as duas restrições são activas e a condição (2.53) só é respeitada se λ_1^g for positivo e λ_2^g negativo. A obtenção de um multiplicador de Lagrange negativo indica que a respectiva restrição não devia ser activa e que existe outra solução admissível à qual corresponde um valor inferior da função objectivo. Este assunto será desenvolvido na secção 2.2.5. No Capítulo 4 será apresentado um método de resolução do sistema de equações (2.50) que não implica a consideração das 2^m combinações de restrições desigualdade activas/inactivas.

2.2.5 - Significado dos multiplicadores de Lagrange

A solução do programa matemático (2.33) pode ser obtida recorrendo à resolução do sistema de equações (2.37). Neste tipo de problemas a solução é constituída pelos n valores das variáveis x_i e pelos l valores dos multiplicadores de Lagrange λ_k^h . Depois de obtida esta solução pode haver interesse em saber quanto varia o valor da função objectivo (custo) quando uma constante ε_k^h é adicionada a cada uma das restrições $h_k(\underline{x}) = 0$ ($k = 1, \dots, l$). Deste modo é possível avaliar quais são as restrições cuja modificação mais influencia o valor final da função objectivo. O estudo que é em seguida apresentado destina-se a possibilitar a obtenção destes resultados sem ser necessário repetir a resolução do programa matemático. As conclusões deste estudo fornecem também uma explicação relativa ao significado físico dos multiplicadores de Lagrange.

Apresenta-se em seguida o cálculo da derivada da função objectivo f em ordem ao parâmetro ε^h que é adicionado à única restrição igualdade que neste caso é considerada. A extensão deste estudo a programas matemáticos com múltiplas restrições encontra-se em [Lue84] e consiste apenas numa generalização do caso aqui tratado.

O cálculo de $df/d\varepsilon^h$ é efectuado com base na solução óptima do seguinte programa matemático

$$\text{Mín. } f(x_1, \dots, x_n) \quad (2.54a)$$

s.a

$$\bar{h}(x_1, \dots, x_n, \varepsilon^h) = h(x_1, \dots, x_n) + \varepsilon^h = 0 \quad (2.54b)$$

Em (2.54) apenas é considerada uma restrição igualdade $h(\underline{x}) = 0$ à qual é adicionado o parâmetro ε^h , sendo o resultado designado \bar{h} . O programa matemático (2.54) possui uma solução que depende do valor de ε^h e que é designada $\underline{x}^*(\varepsilon^h)$. A esta solução corresponde um valor do multiplicador de Lagrange que também depende de ε^h e é designado $\lambda^h(\varepsilon^h)$. No ponto correspondente à solução óptima o gradiente do Lagrangeano de (2.54) é nulo, resultando

$$\frac{\partial f}{\partial x_i} + \lambda^h \frac{\partial \bar{h}}{\partial x_i} = 0 \quad (i = 1, \dots, n) \quad (2.55a)$$

$$\bar{h} = 0 \quad (2.55b)$$

Multiplicando ambos os membros de (2.55a) pelo vector $dx_i / d\varepsilon^h$ e substituindo $\partial \bar{h} / \partial x_i$ por $\partial h / \partial x_i$ obtém-se

$$\sum_{i=1}^n \left(\frac{\partial f}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) + \lambda^h \cdot \sum_{i=1}^n \left(\frac{\partial h}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) = 0 \quad (2.56)$$

Uma vez que $\bar{h} = 0$ qualquer que seja o valor de ε^h conclui-se que

$$\frac{d\bar{h}}{d\varepsilon^h} = 0 \quad (2.57)$$

$$\frac{d\bar{h}}{d\varepsilon^h} = \sum_{i=1}^n \left(\frac{\partial \bar{h}}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) + \frac{\partial \bar{h}}{\partial \varepsilon^h} = 0 \quad (2.58)$$

$$\sum_{i=1}^n \left(\frac{\partial h}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) = -1 \quad (2.59)$$

Substituindo (2.59) em (2.56) obtém-se

$$\sum_{i=1}^n \left(\frac{\partial f}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) = \lambda^h \quad (2.60)$$

A expressão (2.60) fornece o resultado final pretendido porque

$$\frac{df}{d\varepsilon^h} = \sum_{i=1}^n \left(\frac{\partial f}{\partial x_i} \frac{dx_i}{d\varepsilon^h} \right) = \lambda^h \quad (2.61)$$

Assim se conclui que para um valor pequeno de ε^h

$$\frac{\Delta f}{\Delta \varepsilon^h} \cong \lambda^h \quad (2.62)$$

A expressão (2.62) indica que a variação do custo quando é adicionado à restrição h um ε^h pequeno é aproximadamente igual ao valor do multiplicador de Lagrange correspondente à solução óptima obtida com $\varepsilon^h = 0$. O valor de ε^h deve ser pequeno porque em (2.62) é considerada uma aproximação linear de uma relação que em geral é não linear. O estudo da influência na solução óptima da variação de certos parâmetros é genericamente designado análise pós-optimização.

Apresenta-se em seguida um exemplo simples destinado a clarificar as características do cálculo de $df / d\varepsilon^h$. No seguinte programa matemático apenas existem duas variáveis e uma restrição igualdade à qual é adicionado o parâmetro ε^h

$$\text{Min. } f(x_1, x_2) = x_1^2 + x_2^2 \tag{2.63a}$$

s.a

$$\bar{h}(x_1, x_2, \varepsilon^h) = -x_1 - x_2 + 4 + \varepsilon^h = 0 \tag{2.63b}$$

Neste caso simples é possível obter a solução de (2.63) em função de ε^h

$$x_1^*(\varepsilon^h) = x_2^*(\varepsilon^h) = \varepsilon^h / 2 + 2 \tag{2.64a}$$

$$\lambda^h(\varepsilon^h) = \varepsilon^h + 4 \tag{2.64b}$$

$$f(x_1^*(\varepsilon^h), x_2^*(\varepsilon^h)) = (\varepsilon^h)^2 / 2 + 4\varepsilon^h + 8 \tag{2.64c}$$

Derivando f em ordem a ε^h obtém-se

$$\frac{df}{d\varepsilon^h} = \varepsilon^h + 4 \tag{2.65}$$

Comparando (2.64b) com (2.65) pode-se comprovar o resultado atrás obtido e expresso em (2.61).

O estudo aqui efectuado pode ser generalizado a problemas de optimização em que estão presentes diversas restrições igualdade e desigualdade [Aro89]. Com este fim foi considerado o seguinte programa matemático, a cujas restrições foram adicionados os parâmetros $\tilde{\varepsilon}^g$ e $\tilde{\varepsilon}^h$

$$\text{Min. } f(\tilde{x}) \tag{2.66a}$$

s.a

$$g_j(\tilde{x}) + \varepsilon_j^g \leq 0 \quad (j = 1, \dots, m) \tag{2.66b}$$

$$h_k(\tilde{x}) + \varepsilon_k^h = 0 \quad (k = 1, \dots, l) \tag{2.66c}$$

No ponto correspondente à solução óptima $\tilde{x}^*(\tilde{\varepsilon}^g, \tilde{\varepsilon}^h)$, $\tilde{\lambda}^g(\tilde{\varepsilon}^g, \tilde{\varepsilon}^h)$, $\tilde{\lambda}^h(\tilde{\varepsilon}^g, \tilde{\varepsilon}^h)$ as derivadas da função objectivo em ordem às variáveis $\tilde{\varepsilon}^g$ e $\tilde{\varepsilon}^h$ são

$$\frac{\partial f}{\partial \tilde{\varepsilon}^g} = \tilde{\lambda}^g \tag{2.67a}$$

$$\frac{\partial f}{\partial \varepsilon_j^g} = \lambda_j^g \quad (2.67b)$$

Estas expressões só são válidas se a solução \underline{x}^* , $\underline{\lambda}^g$, $\underline{\lambda}^h$ respeitar as condições de optimalidade de Karush-Kuhn-Tucker que serão mais adiante apresentadas. Se uma restrição do tipo (2.66b) for activa para todos os valores de ε_j^g pertencentes a um intervalo centrado em zero, o estudo de $\partial f / \partial \varepsilon_j^g$ pode ser efectuado como se se tratasse de uma restrição igualdade. Se a restrição for inactiva para ε_j^g pertencente a um intervalo centrado em zero, o valor óptimo da função objectivo não é condicionado por essa restrição. Neste caso a uma variação de ε_j^g corresponde apenas uma variação do valor da variável de desvio associada a essa restrição. Por este motivo $\partial f / \partial \varepsilon_j^g$ é nulo. Uma vez que as restrições inactivas possuem multiplicador de Lagrange nulo, conclui-se que a expressão (2.67a) é também válida neste caso.

A partir de (2.66b) e de (2.67a) é possível deduzir uma condição necessária de optimalidade relativa ao sinal dos multiplicadores de Lagrange associados às restrições desigualdade. As considerações que se seguem referem-se apenas a restrições desigualdade que se mantêm activas para ε_j^g pertencente a um intervalo centrado em zero. De acordo com (2.66b), um aumento do valor de ε_j^g implica que alguns pontos da região admissível deixem de lhe pertencer. Existe portanto uma diminuição da região admissível que implica um aumento do valor da função objectivo ou a sua manutenção. Conclui-se portanto que $\partial f / \partial \varepsilon_j^g$ não pode ser negativo e atendendo a (2.67a) resulta que

$$\lambda_j^g \geq 0 \quad (j = 1, \dots, m) \quad (2.68)$$

De acordo com as conclusões apresentadas pode ainda observar-se que o valor de um multiplicador λ_j^g indica qual o acréscimo no custo associado a uma diminuição da região admissível. Obviamente também indica quanto se beneficia ao aumentar a região admissível. De acordo com (2.62) o valor do multiplicador de Lagrange é aproximadamente igual à variação da função objectivo que é provocada por uma variação unitária da constante associada à restrição. O erro desta aproximação só é aceitável se ε_j^g e ε_k^h forem suficientemente pequenos de modo a não ser alterado o conjunto das restrições activas. Cada multiplicador de Lagrange encontra-se numa unidade correspondentes ao quociente entre as unidades da função objectivo e as unidades da respectiva restrição. Quando existe um conjunto de restrições do mesmo tipo, a comparação directa dos valores dos respectivos multiplicadores de Lagrange indica quais são as restrições que mais estão a contribuir para o custo correspondente à solução óptima. Se o método de optimização utilizado fornecer uma

solução com um ou mais multiplicadores λ_j^g negativos, deve haver uma intervenção que conduza ao seu desaparecimento. O procedimento mais simples consiste em suprimir a restrição que apresentar o menor valor de λ_j^g e repetir o cálculo da solução óptima. É também possível resolver o problema relaxando a restrição ou adoptando uma nova solução inicial que implique a existência de grandes folgas nas restrições que apresentaram valores negativos de λ_j^g .

2.2.6 - Condições de Karush-Kuhn-Tucker

As condições de optimalidade relativas ao programa matemático (2.21) são em seguida apresentadas em duas versões distintas mas equivalentes. O referido programa matemático apresenta a seguinte forma genérica

$$\text{Min. } f(\underline{x}) \quad \underline{x} = (x_1, \dots, x_n) \quad (2.69a)$$

s.a

$$g_j(\underline{x}) \leq 0 \quad (j = 1, \dots, m) \quad (2.69b)$$

$$h_k(\underline{x}) = 0 \quad (k = 1, \dots, l) \quad (2.69c)$$

Algumas das condições necessárias de optimalidade foram atrás deduzidas (2.50) com base no anulamento do gradiente do Lagrangeano. As restrições desigualdade foram previamente convertidas em restrições igualdade por intermédio de variáveis de desvio s_j^2 ($j = 1, \dots, m$). Na secção anterior foi deduzida uma condição necessária de optimalidade (2.68), que requer a não negatividade dos multiplicadores de Lagrange associados a restrições desigualdade. A primeira versão das condições necessárias de optimalidade é constituída pelas referidas condições (2.50) e (2.68). Estas expressões são aqui reproduzidas.

$$\frac{\partial f}{\partial x_i} + \sum_{j=1}^m \left(\lambda_j^g \frac{\partial g_j}{\partial x_i} \right) + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial h_k}{\partial x_i} \right) = 0 \quad (i = 1, \dots, n) \quad (2.70a)$$

$$2s_j \lambda_j^g = 0 \quad (j = 1, \dots, m) \quad (2.70b)$$

$$\lambda_j^g \geq 0 \quad (j = 1, \dots, m) \quad (2.70c)$$

$$g_j + s_j^2 = 0 \quad (j = 1, \dots, m) \quad (2.70d)$$

$$h_k = 0 \quad (k = 1, \dots, l) \quad (2.70e)$$

As condições (2.70b) apenas impõe que s_j e λ_j^g não sejam simultaneamente não nulos, ie, pelo menos uma das variáveis tem de ser nula. Também se verifica que, sendo uma das variáveis nula, a outra pode assumir qualquer valor. A expressão (2.70b) pode ser substituída pela seguinte condição que possui as mesmas características

$$-s_j^2 \lambda_j^g = 0 \quad (j = 1, \dots, m) \quad (2.71)$$

A expressão (2.70d) permite efectuar em (2.71) a substituição $-s_j^2 = g_j$. Reconvertendo (2.70d) para a sua forma original (2.69b) as variáveis de desvio deixam de estar presentes nas condições de optimalidade. Obtêm-se assim as condições de Karush-Kuhn-Tucker na sua versão habitual [Van 84] [Aro 89], que é a seguinte

$$\frac{\partial f}{\partial x_i} + \sum_{j=1}^m \left(\lambda_j^g \frac{\partial g_j}{\partial x_i} \right) + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial h_k}{\partial x_i} \right) = 0 \quad (i = 1, \dots, n) \quad (2.72a)$$

$$\lambda_j^g g_j = 0 \quad (j = 1, \dots, m) \quad (2.72b)$$

$$\lambda_j^g \geq 0 \quad (j = 1, \dots, m) \quad (2.72c)$$

$$g_j \leq 0 \quad (j = 1, \dots, m) \quad (2.72d)$$

$$h_k = 0 \quad (k = 1, \dots, l) \quad (2.72e)$$

As condições (2.72d) e (2.72e) podem ser substituídas pela afirmação de que a solução óptima tem de ser admissível.

Para referir uma situação de excepção relativa às condições de Karush-Kuhn-Tucker apresenta-se em seguida a definição de ponto regular. Uma solução de (2.69) é um ponto regular se nesse ponto os gradientes das restrições igualdade e das restrições desigualdade activas forem linearmente independentes [Lue84]. Quando o mínimo global ou um mínimo local de um problema de optimização não é um ponto regular as condições necessárias (2.70) ou (2.72) podem não ser verificadas. Apresenta-se em seguida um problema de optimização cujo mínimo global não é um ponto regular e não verifica as condições (2.72) [McC67].

$$\text{Min. } f(x_1, x_2) = x_1^2 + x_2^2 - 4x_1 + 4 \quad (2.73a)$$

s.a

$$g_1(x_1, x_2) = -x_1 \leq 0 \quad (2.73b)$$

$$g_2(x_1, x_2) = -x_2 \leq 0 \quad (2.73c)$$

$$g_3(x_1, x_2) = x_1^3 - 3x_1^2 + 3x_1 + x_2 - 1 \leq 0 \quad (2.73d)$$

A representação gráfica deste problema encontra-se na Figura 2.8.

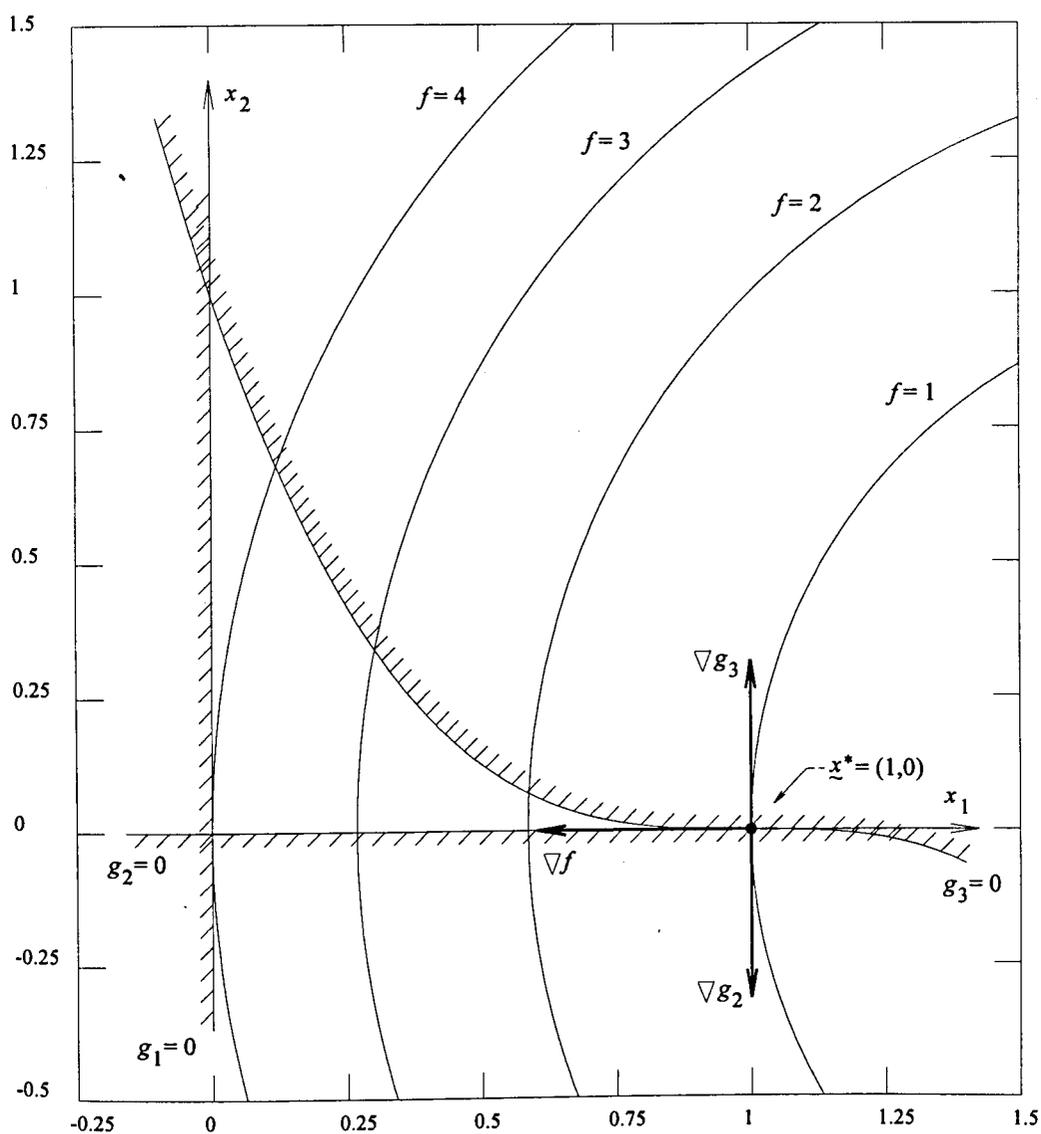


Figura 2.8 - Representação gráfica do programa matemático (2.73).

Pode-se verificar graficamente que o mínimo global de (2.73) é o ponto $\tilde{x}^* = (1, 0)$. Nesse ponto a restrição g_1 é inactiva e as restantes são activas. Uma vez que $g_1 \neq 0$ a condição (2.72b) obriga que λ_1^g seja nulo. De acordo com (2.72a), $\nabla f(\tilde{x}^*) = (-2, 0)$ tem de ser uma combinação linear de $\nabla g_2(\tilde{x}^*) = (0, -1)$ e de $\nabla g_3(\tilde{x}^*) = (0, 1)$. Uma vez que estes dois vectores são normais a $\nabla f(\tilde{x}^*)$, não existe nenhum par $(\lambda_2^g, \lambda_3^g)$ que verifique (2.72a). Pode também verificar-se que no ponto $\tilde{x}^* = (1, 0)$ os gradientes das restrições activas ∇g_2 e ∇g_3 são linearmente dependentes. Assim se constatou que o mínimo global de (2.73) não é um ponto regular e não verifica as condições necessárias de Karush-Kuhn-Tucker (2.72). Presume-se que na generalidade dos problemas de optimização esta situação particular raramente ocorre.

No Capítulo 6 serão apresentados alguns problemas de optimização cuja solução não é um ponto regular. Nesse tipo de problemas é efectuada a substituição de cada restrição igualdade $h = 0$ por um par de restrições desigualdade $h \leq 0$, $-h \leq 0$. Estas duas restrições têm de ser simultaneamente activas, porque se uma delas for inactiva a outra é violada. Os gradientes destas restrições (∇h e $-\nabla h$) são linearmente dependentes em qualquer ponto que satisfaça simultaneamente $h \leq 0$, $-h \leq 0$. Por este motivo e uma vez que tem de ser admissível, a solução óptima do problema não é um ponto regular. Apesar de se verificar este facto, foi possível em cada caso obter uma solução x^* coincidente com a do problema inicial com restrições igualdade. Apresenta-se em seguida o desenvolvimento destas questões no caso de um programa matemático com uma única restrição igualdade. As considerações em seguida efectuadas podem facilmente ser generalizadas a programas matemáticos com múltiplas restrições.

$$\text{Min. } f(\tilde{x}) \quad \tilde{x} = (x_1, \dots, x_n) \quad (2.74a)$$

s.a

$$h(\tilde{x}) = 0 \quad (2.74b)$$

Efectuando a substituição da restrição igualdade por um par de restrições desigualdade equivalentes e abreviando a notação, o programa matemático passa a ser o seguinte

$$\text{Min. } f \quad (2.75a)$$

s.a

$$g_1 = h \leq 0 \quad (2.75b)$$

$$g_2 = -h \leq 0 \quad (2.75c)$$

O sistema de equações não lineares correspondente ao anulamento do gradiente do Lagrangeano (2.50) é neste caso o seguinte

$$\nabla f + \lambda_1^g \nabla h - \lambda_2^g \nabla h = 0 \quad (2.76a)$$

$$h + s_1^2 = 0 \quad (2.76b)$$

$$-h + s_2^2 = 0 \quad (2.76c)$$

$$2s_1 \lambda_1^g = 0 \quad (2.76d)$$

$$2s_2 \lambda_2^g = 0 \quad (2.76e)$$

Com todo o método de optimização iterativo utilizado no presente trabalho é possível obter a solução deste sistema de $n + 4$ equações a $n + 4$ incógnitas $(\tilde{x}, s_1, s_2, \lambda_1^g, \lambda_2^g)$.

O sistema de equações (2.76) apresenta algumas características particulares que são em seguida indicadas. Uma vez que as restrições (2.75b) e (2.75c) têm de ser ambas activas, as respectivas variáveis de desvio s_1 e s_2 são nulas. Efectuando a substituição de $s_1 = s_2 = 0$ em (2.76) as equações (2.76d) e (2.76e) podem ser suprimidas. O sistema de equações resultante é o seguinte

$$\nabla f + (\lambda_1^g - \lambda_2^g) \nabla h = 0 \quad (2.77a)$$

$$h = 0 \quad (2.77b)$$

$$-h = 0 \quad (2.77c)$$

Este sistema é agora de $n + 2$ equações a $n + 2$ incógnitas e é indeterminado porque apresenta uma equação repetida. Eliminando (2.77c) e considerando a substituição de variáveis

$$\lambda_1^g - \lambda_2^g = \lambda^h \quad (2.78)$$

resulta o seguinte sistema de $n + 1$ equações a $n + 1$ incógnitas

$$\nabla f + \lambda^h \nabla h = 0 \quad (2.79a)$$

$$h = 0 \quad (2.79b)$$

Este sistema de equações coincide com o que resulta do anulamento do gradiente do Lagrangeano do problema inicial com uma restrição igualdade (2.74).

O método de optimização utilizado no Capítulo 6 consegue resolver o sistema de equações indeterminado (2.76), fornecendo uma das suas soluções. Se forem utilizadas soluções iniciais diferentes obtêm-se múltiplas soluções que apenas diferem nos valores de λ_1^g e λ_2^g . Em cada solução obtida o valor de $\lambda_1^g - \lambda_2^g$ corresponde ao multiplicador de Lagrange λ^h do problema inicial (2.74). De acordo com (2.77) e (2.79) qualquer par $(\lambda_1^g, \lambda_2^g)$ que satisfaça (2.78) pode figurar numa solução válida. O facto de λ_1^g ou λ_2^g ser negativo não apresenta qualquer inconveniente porque em todas as soluções $\lambda_1^g - \lambda_2^g = \lambda^h$ e este último não possui qualquer restrição de sinal.

O aparecimento de valores negativos para λ_1^g ou λ_2^g constitui uma violação das condições de Karush-Kuhn-Tucker (2.72) justificada pelo facto de a solução não ser um ponto regular. Foi possível verificar que este facto não inviabiliza a obtenção de soluções correctas de problemas de optimização.

2.3 - OBSERVAÇÕES FINAIS

As condições de Karush-Kuhn-Tucker (2.72) foram desenvolvidas independentemente por W. Karush [Kar39] e por H.W. Kuhn e A. W. Tucker [Kuh51]. Durante um longo período foram designadas apenas condições de Kuhn-Tucker, só tendo sido recentemente reconhecido o trabalho anterior de W. Karush. As condições K.K.T. são também designadas condições necessárias de primeira ordem, porque são apenas referidas primeiras derivadas das funções f , g e h , que têm obviamente de ser deriváveis. Pelo facto de serem condições necessárias, uma solução que apenas respeite as condições K.K.T. pode ser um mínimo local, ponto de sela ou máximo local. Em [Lue84] são apresentadas condições suficientes que garantem que a solução é um mínimo local. Estas condições requerem que seja verificado se a matriz Hessiana do Lagrangeano é positiva definida num determinado sub espaço. Como esta matriz apresenta geralmente um número elevado de linhas e colunas esta verificação não é efectuada, porque exigiria um elevado volume de cálculo. Existem também condições suficientes que garantem que uma solução é o mínimo global. Neste caso é necessário que a região admissível

e a função objectivo sejam convexas [Aro89]. Uma vez que os problemas de optimização de estruturas de um modo geral não são convexas, estas condições suficientes têm pouco interesse prático. Para beneficiar das vantagens da existência de convexidade, é possível abordar a optimização de estruturas recorrendo a aproximações convexas [Fle87]. No presente trabalho apenas foram consideradas as condições necessárias de optimalidade na versão (2.70). Nestas circunstâncias, para que a solução obtida seja o mínimo global do problema ou pelo menos uma solução próxima é conveniente iniciar o método de optimização a partir de distintas soluções iniciais. A preparação destas soluções iniciais deve ser efectuada atendendo ao significado físico dos componentes do programa matemático. Nesta fase, a intuição e experiência de quem formulou o problema e preparou a solução inicial são fundamentais. Com o método de optimização utilizado neste trabalho foi possível resolver diversos tipos de problemas de optimização de estruturas recorrendo apenas às condições necessárias (2.70).

CAPÍTULO 3

MÉTODOS DE OPTIMIZAÇÃO

A resolução de problemas de optimização formulados como um programa matemático (ver Capítulo 2) é na generalidade dos casos efectuada com recurso a um método iterativo. Estes métodos requerem a preparação de uma solução inicial, que vai sendo sucessivamente modificada até ser satisfeito um determinado critério de convergência. Na Figura 3.1 encontra-se um algoritmo que inclui algumas das fases que são comuns à generalidade dos métodos de optimização.

No algoritmo da Figura 3.1 q é o contador de iterações e a solução inicial \tilde{x}^0 é considerada como um dado. O critério de convergência em que é baseada a interrupção do processo iterativo depende das características do método de optimização. Nos casos em que não se dispõe de um critério de avaliação da qualidade da solução corrente, o processo iterativo deve ser interrompido quando deixarem de ocorrer variações significativas da solução. Deve também ser prevista a interrupção do processo iterativo quando for alcançado um número limite de iterações antecipadamente fixado. Assim se evita que o processo iterativo evolua indefinidamente nos casos em que não ocorrer a convergência para a solução óptima. Ao vector $\tilde{\Delta x}^q$ está tradicionalmente associado o conceito de direcção no espaço, pelo facto de ele deslocar o ponto \tilde{x}^{q-1} em direcção ao ponto \tilde{x}^q . A grandeza deste deslocamento pode ser modificada por intermédio do parâmetro α^q , que é designado parâmetro de pesquisa unidimensional (*line search*). O parâmetro α^q deve adoptar um valor que minimize a função objectivo ou o erro na direcção $\tilde{\Delta x}^q$. De um modo geral revela-se vantajoso que o parâmetro α^q adopte um valor que conduza a uma nova solução corrente $\left(\tilde{x}^{q-1} + \alpha^q \tilde{\Delta x}^q\right)$ admissível. Alguns métodos possuem a capacidade de fazer com que a solução regresse à região admissível, caso ocorram soluções intermédias não admissíveis.

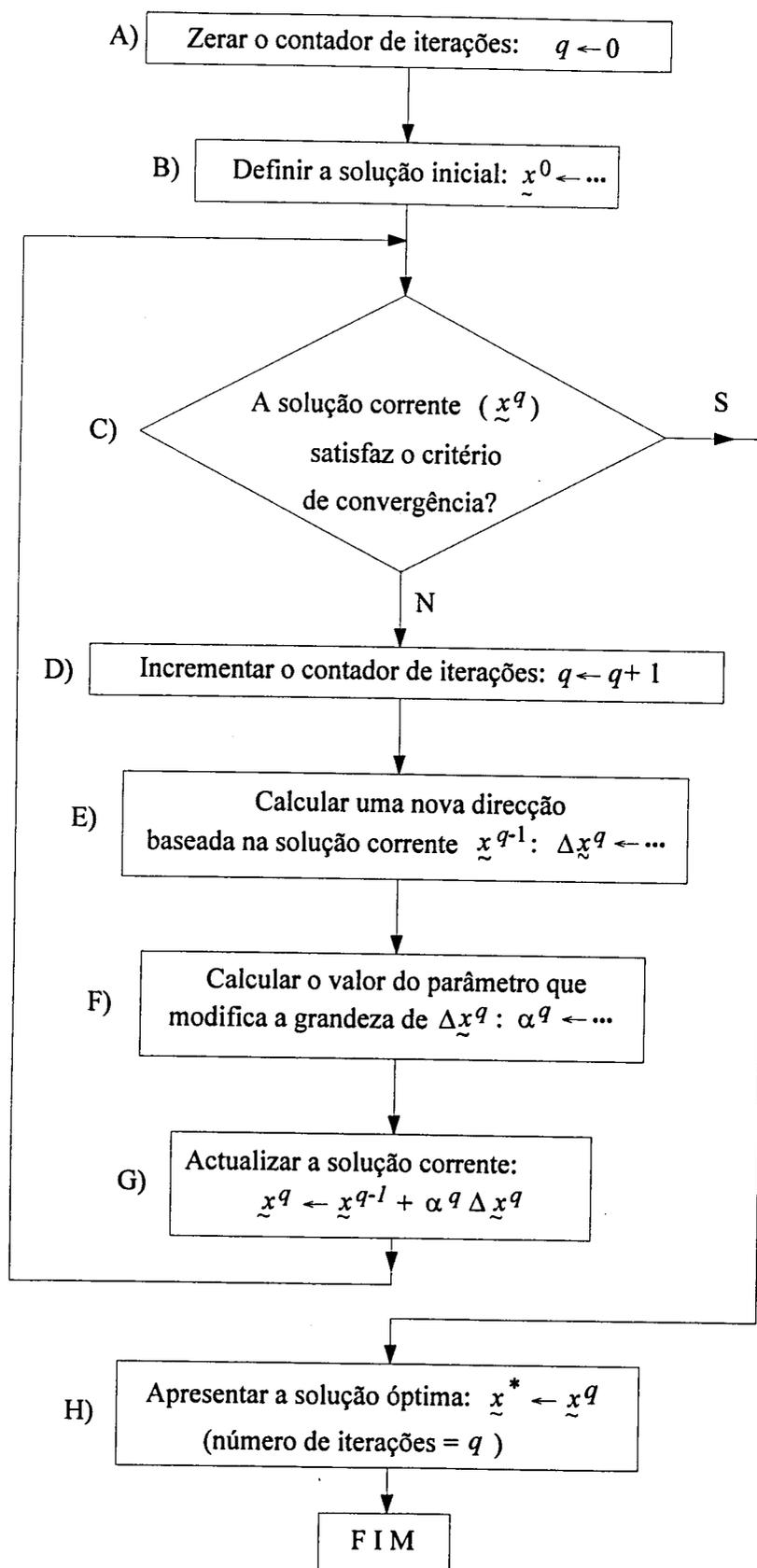


Figura 3.1 - Algoritmo comum à generalidade dos métodos de optimização.

No algoritmo da Figura 3.1 a fase que fundamentalmente caracteriza cada um dos métodos de optimização é a correspondente ao cálculo de Δx^q . Este vector é na generalidade dos casos calculado depois de serem efectuadas modificações no programa matemático que descreve o problema. Estas modificações são em geral simplificações que transformam o problema inicial num problema aproximado. Quanto mais próxima da solução óptima estiver a solução corrente, menor será o erro associado à referida aproximação. Por este motivo é desejável que a solução inicial se encontre o mais próximo possível da solução óptima. Esta questão será clarificada com a apresentação de alguns métodos de optimização mais adiante neste capítulo.

Durante as últimas décadas foram desenvolvidos e aperfeiçoados por diversos investigadores um número relativamente elevado de métodos de optimização. Estes métodos podem ser agrupados em classes de acordo com determinadas características comuns. É contudo difícil definir um único critério de classificação de métodos de optimização, porque quase todos envolvem conceitos ou procedimentos que estão também presentes nos restantes métodos. Existem no entanto alguns critérios de classificação óbvios como por exemplo:

A1 - minimização de funções com uma variável

A2 - minimização de funções com n variáveis

Podem também ser considerados separadamente os dois seguintes casos:

B1 - minimização de funções sem restrições

B2 - minimização de funções com restrições

É evidente que os métodos destinados a resolver problemas do tipo A2 também resolvem os problemas do tipo A1. O caso B1 podem também ser considerado como um caso particular de B2.

No presente capítulo, os métodos de minimização sem restrições são classificados de acordo com a ordem das derivadas por ele utilizadas. De acordo com este critério são considerados os seguintes casos:

C1 - métodos de ordem zero - não requerem qualquer derivada

C2 - métodos de primeira ordem - requerem primeiras derivadas

C3 - métodos de segunda ordem - requerem segundas derivadas

No Capítulo 2 (ver Quadro 2.1) foi apresentada uma classificação de problemas de optimização com base no tipo de variáveis de projecto e no tipo de funções que constituem o

programa matemático (função objectivo e restrições). De acordo com o tipo de variáveis presentes no problema podem ser considerados os seguintes casos [Hil90]:

- D1 - caso geral - variáveis reais e contínuas
- D2 - programação dinâmica - variáveis discretas
- D3 - programação inteira - variáveis inteiras
- D4 - programação inteira binária - variáveis binárias

Os tipos de funções que constituem a função objectivo e restrições permitem distinguir os seguintes casos de programação matemática [Gil81]:

- E1 - programação linear - função objectivo e restrições lineares
- E2 - programação quadrática - função objectivo quadrática e restrições lineares
- E3 - programação geométrica - função objectivo e restrições polinomiais
- E4 - programação não linear - função objectivo e restrições não lineares

Os métodos de resolução de problemas com restrições dividem-se em dois grandes grupos [Bel85]:

- F1 - métodos primais - as restrições são explicitamente consideradas
- F2 - métodos de transformação - o problema com restrições é previamente transformado num problema sem restrições

Os métodos de transformação podem por sua vez ser classificados em três subgrupos, de acordo com o tipo de transformação que é efectuada:

- F2a - funções penalidade
- F2b - Lagrangeano
- F2c - Lagrangeano aumentado

Alguns dos métodos atrás referidos são por vezes aplicados mais do que uma vez a um mesmo problema. No fim de cada minimização a aproximação do programa matemático é recalculada e/ou alguns parâmetros são actualizados. Esta técnica é utilizada nos seguintes métodos

- programação linear sequencial - *sequential linear programming* (SLP) [Van84]
- programação quadrática sequencial - *sequential quadratic programming* (SQP) [Kao94]
- minimização sem restrições sequencial - *sequential unconstrained minimization techniques* (SUMT) (utilizados com F2a) [Van84]

- actualização de multiplicadores de Lagrange - *multiplier update methods* (utilizados com F2c) [Aro91]

Para a generalidade das classes de métodos aqui referidos existem diversos algoritmos e diversas variantes de cada algoritmo. Esta grande diversidade de técnicas de optimização implica uma certa dificuldade na selecção do método mais adequado à resolução de cada tipo de problema. Neste capítulo serão apresentados alguns métodos de optimização, as vantagens e desvantagens de cada um deles e por fim alguns critérios de selecção de algoritmos. Pelo facto de serem os mais utilizados na optimização de estruturas, apenas serão referidos métodos aplicáveis a problemas com variáveis reais e contínuas. Serão apresentados com mais pormenor os métodos que foram utilizados no presente trabalho.

3.1 - MINIMIZAÇÃO DE FUNÇÕES COM UMA VARIÁVEL

A minimização de funções com uma variável e sem restrições é o caso mais simples de todos os que são em seguida referidos. Uma vez que possui uma representação gráfica também simples, é possível visualizar as características e o comportamento dos diversos métodos. O problema genérico de minimização de uma função escalar reduz-se neste caso à seguinte formulação

$$\text{Min. } f(x) \quad (3.1)$$

Devido às suas características particulares e ao facto de serem utilizados no passo F) do algoritmo da Figura 3.1, são em seguida apresentados alguns métodos de minimização de funções com uma variável. Estes métodos são classificados de acordo com o grau das derivadas utilizadas no respectivo algoritmo. São referidos métodos de ordem zero, de primeira e de segunda ordem.

3.1.1 - Métodos de ordem zero

Os métodos de ordem zero apenas requerem a avaliação da função $f(x)$ em diversos pontos. Em vez de uma solução inicial estes métodos necessitam em geral do conhecimento prévio de um intervalo que contenha o mínimo local que se pretende calcular. É também conveniente que nesse intervalo exista um só mínimo local. Em certos problemas particulares

cujo significado forneça o referido intervalo, evita-se a necessidade de calcular os respectivos limites recorrendo a um algoritmo auxiliar [Van84].

Apresenta-se em seguida o método de aproximação polinomial de segundo grau, que se revela na generalidade dos casos um dos métodos mais eficientes na resolução do problema (3.1) [Gil81]. Este método fornece um resultado mais rigoroso se se partir de um intervalo $[x_A, x_B]$ que contenha o mínimo local procurado. A função f é avaliada em x_A , em x_B e no ponto médio do intervalo $x_M = (x_A + x_B)/2$ (ver Figura 3.2).

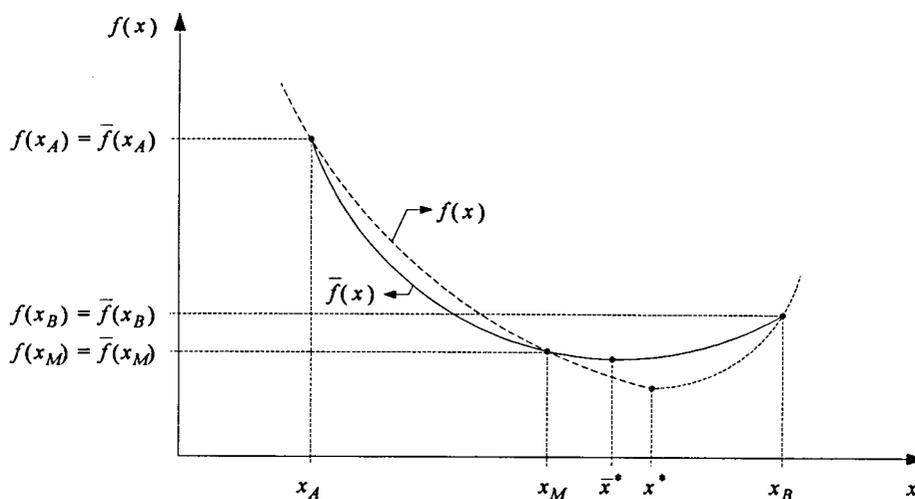


Figura 3.2 - Minimização de $f(x)$ por aproximação polinomial.

Depois de avaliar a função f nos três referidos pontos calculam-se os coeficientes a , b e c dos termos do polinómio do segundo grau que passa por $(x_A, f(x_A))$, $(x_M, f(x_M))$ e $(x_B, f(x_B))$.

$$\bar{f}(x) = ax^2 + bx + c \quad (3.2)$$

A solução aproximada \bar{x}^* é o ponto em que se anula a primeira derivada de \bar{f} . A respectiva expressão é neste caso facilmente obtida

$$\bar{x}^* = -\frac{b}{2a} \quad (3.3)$$

A solução \bar{x}^* obtida com (3.3) só apresenta uma precisão aceitável se a função $f(x)$ for aproximada com suficiente rigor por $\bar{f}(x)$. Em geral obtêm-se soluções precisas desde que o intervalo $[x_A, x_B]$ seja de pequena amplitude e envolva um único mínimo local. Nos casos

em que não seja conhecido um intervalo com estas características, tem de se recorrer a um outro algoritmo que forneça os valores de x_A e x_B . Se se pretender uma solução \bar{x}^* mais precisa, é possível recorrer a sucessivas aplicações do método de aproximação polinomial. Sempre que este método é reiniciado, o intervalo $[x_A, x_B]$ deve passar a ter uma amplitude consideravelmente menor e deve ser centrado na solução \bar{x}^* obtida na iteração anterior.

A minimização de funções com uma variável pode ser efectuada por outros métodos de ordem zero. Em [Box69] são descritos os métodos de Fibonacci e da secção de ouro (*golden section*). Em ambos estes métodos é necessário fornecer um intervalo $[x_A, x_B]$ que contenha um único mínimo local. Este intervalo é sucessivamente diminuído com umas proporções que permitem aproveitar na iteração corrente alguns cálculos de $f(x)$ efectuados nas iterações anteriores. O método de Fibonacci possui a vantagem de se poder determinar antecipadamente o número de iterações que será necessário efectuar para obter uma solução com o erro pretendido. Apesar de fornecerem soluções precisas com um número reduzido de cálculos de $f(x)$, estes métodos na generalidade dos casos práticos não são tão eficientes como o método de aproximação polinomial.

No presente trabalho os métodos de minimização de funções com uma variável apenas foram utilizados na fase F) do algoritmo da Figura 3.1. Atendendo à performance dos computadores actuais e ao tipo de problemas abordados, a esta fase corresponde uma percentagem pouco significativa do tempo total de resolução de um problema de optimização com múltiplas variáveis e com restrições. Por este motivo, no algoritmo de minimização de $f(x)$ foi dada mais importância à fiabilidade e precisão do que à eficiência.

3.1.2 - Métodos de primeira ordem

Na resolução do problema (3.1) por um método de primeira ordem são utilizadas primeiras derivadas da função f . Se não estiver disponível a expressão de $f'(x)$, as derivadas de f podem ser calculadas por diferenças finitas recorrendo à seguinte expressão

$$f'(x) \cong \frac{f(x + \Delta x) - f(x)}{\Delta x} \quad (3.4)$$

O valor de Δx deve ser escolhido tendo em consideração a precisão utilizada no cálculo automático e a ordem de grandeza dos valores de x e de $f(x)$. Se $[x_A, x_B]$ for o intervalo inicial que contém o mínimo local procurado e se os cálculos forem efectuados com 15 algarismos significativos sugere-se para Δx um valor com a seguinte ordem de grandeza

$$\Delta x \cong (x_B - x_A) \times 10^{-6} \quad (3.5)$$

Se a perturbação Δx conduzir a uma variação do valor de f fora ou próxima da precisão utilizada no cálculo automático, o valor de Δx deve ser superior ao indicado em (3.5).

No presente trabalho é utilizado na fase F) do algoritmo da Figura 3.1 o método das bissecções sucessivas aplicado ao cálculo do zero da função $f'(\alpha)$. Este método requer uma prévia determinação de um intervalo $[x_A, x_B]$ que contenha um único mínimo local. Tal como foi referido na secção anterior, este intervalo pode ser definido atendendo ao significado físico do problema ou calculado com um algoritmo auxiliar [Van84].

Na Figura 3.3 encontra-se esquematizado o procedimento correspondente a cada iteração do algoritmo das bissecções sucessivas. O intervalo $[x_A, x_B]$ é truncado à esquerda ou à direita de acordo com o sinal da derivada de $f(x)$ em x_A , x_B e x_M . A adaptação do método das bissecções sucessivas à minimização de $f(x)$ dá origem ao algoritmo indicado na Figura 3.4.

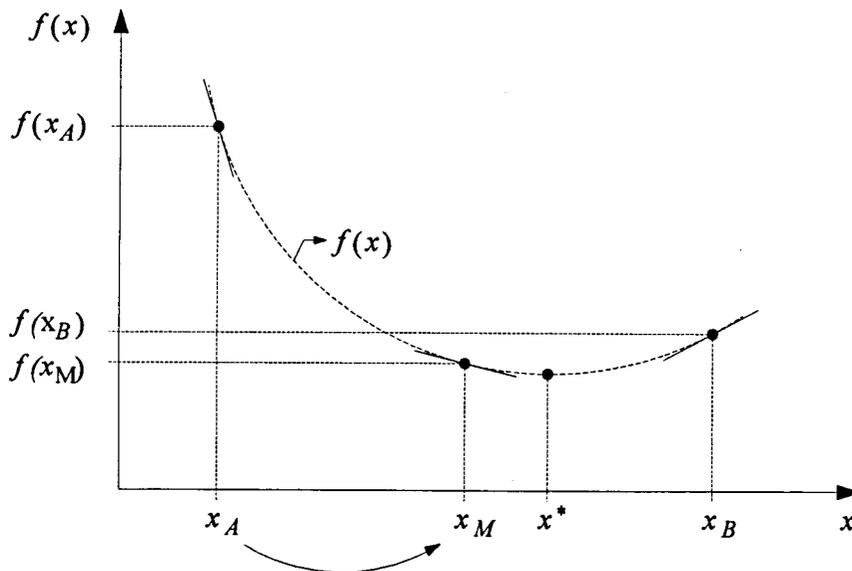


Figura 3.3 - Minimização de $f(x)$ pelo método das bissecções sucessivas.

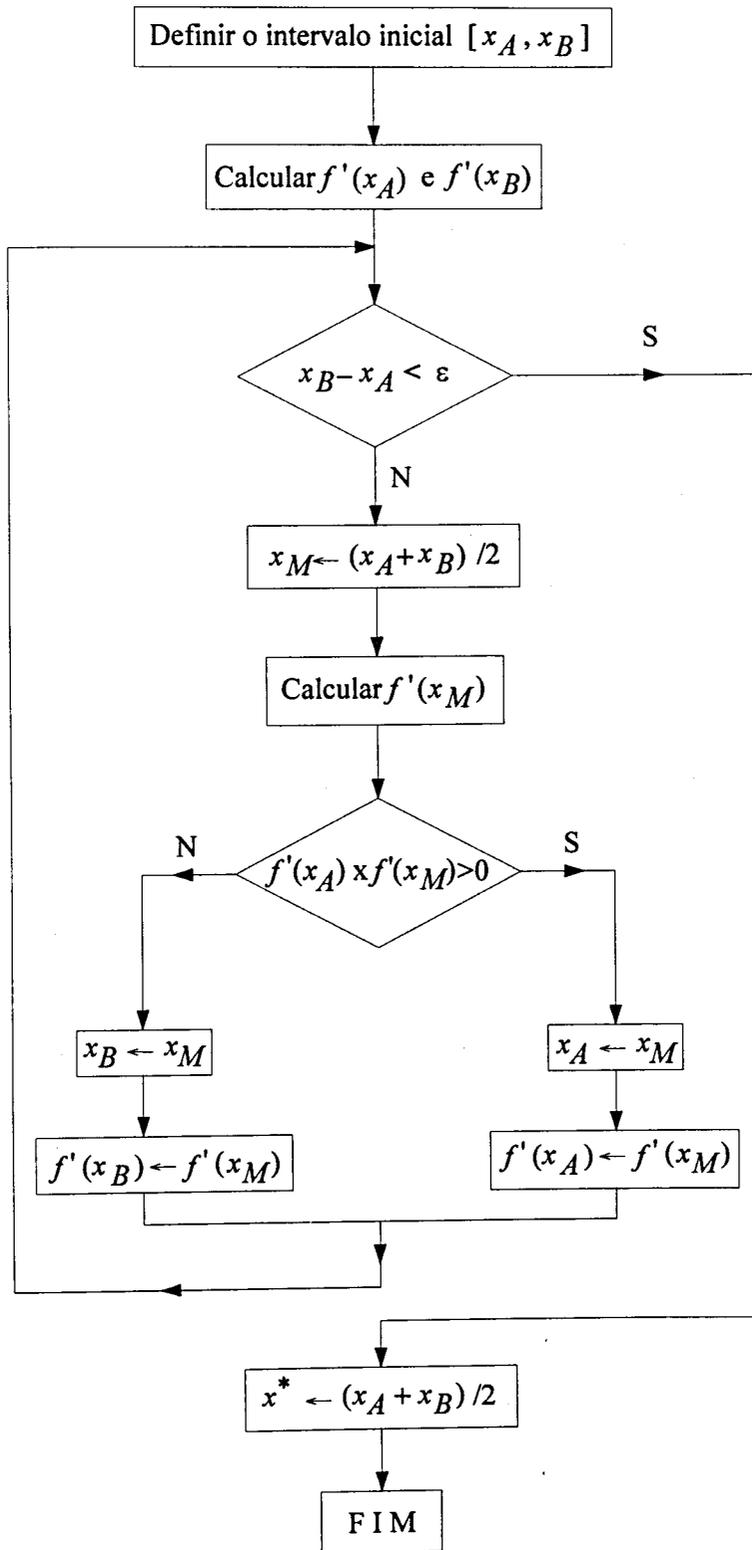


Figura 3.4 - Algoritmo correspondente à minimização de $f(x)$ pelo método das bissecções sucessivas.

O processo iterativo deve ser interrompido quando $x_B - x_A$ for inferior a uma tolerância ε previamente fixada ou quando $f'(x_M)$ for suficientemente pequeno.

Antes de iniciar o processo iterativo pelo método das bissecções sucessivas é necessário verificar se $f'(x_A)$ é negativo e $f'(x_B)$ positivo. Nos casos em que tal não se verificar o intervalo $[x_A, x_B]$ deve ser modificado ou então o mínimo de $f(x)$ deve ser procurado no intervalo inicial recorrendo a outro método. Os casos mais comuns em que não se verifica ser $f'(x_A)$ negativo e $f'(x_B)$ positivo são os correspondentes a uma função $f(x)$ constante, linear crescente, linear decrescente ou côncava no intervalo $[x_A, x_B]$. A estratégia que deve ser adoptada nestes casos será indicada no Capítulo 4, conjuntamente com a adaptação do método das bissecções sucessivas à fase F) do algoritmo da Figura 3.1.

3.1.3 - Métodos de segunda ordem

O clássico método de Newton-Raphson, que se destina ao cálculo do zero de uma função, pode ser adaptado à resolução do problema (3.1). Dessa adaptação resulta um método de minimização de $f(x)$ em que são utilizadas segundas derivadas de f e que é habitualmente designado método de Newton. A aplicação deste método à minimização de uma função com uma variável é efectuada de acordo com uma versão simplificada do algoritmo da Figura 3.1. As simplificações consistem na eliminação da fase F), na consideração de α^q unitário e na substituição dos vectores \underline{x} e $\Delta \underline{x}$ pelos escalares x e Δx .

Uma vez que a solução procurada é um valor x^* que anule a primeira derivada de f , o critério de convergência (fase C) é neste caso o seguinte

$$|f'(x^q)| < \varepsilon \quad (3.6)$$

Uma vez que o método de Newton possui convergência quadrática [Gil81], é habitual atribuir a ε um valor pequeno (por exemplo 10^{-6}).

No algoritmo da Figura 3.1 a fase E) baseia-se no cálculo do valor de Δx^q de tal forma que

$$f'(x^{q-1} + \Delta x^q) = 0 \quad (3.7)$$

Desenvolvendo f' em série de Taylor e conservando apenas os dois primeiros termos obtém-se

$$f'(x^{q-1} + \Delta x^q) \cong f'(x^{q-1}) + f''(x^{q-1})\Delta x^q \quad (3.8)$$

Substituindo (3.8) em (3.7) obtém-se a seguinte equação linear com uma incógnita, cuja resolução fornece o valor de Δx^q

$$f''(x^{q-1})\Delta x^q + f'(x^{q-1}) = 0 \quad (3.9)$$

A equação (3.9) é apresentada com esta forma de modo a possibilitar mais adiante a sua comparação com o caso relativo à minimização de uma função com n variáveis. Os valores de f' e f'' podem ser calculados analiticamente ou por diferenças finitas com base na solução corrente x^{q-1} . Uma vez que no desenvolvimento de f' em série de Taylor apenas foram conservados os dois primeiros termos, a equação (3.9) fornece um valor aproximado de Δx^q . Por este motivo torna-se necessário recorrer ao método iterativo indicado na Figura 3.1. Se f for uma função quadrática é suficiente efectuar uma iteração para obter a solução óptima, porque nestas circunstâncias a aproximação em série de Taylor é exacta.

O algoritmo da Figura 3.1 com as modificações atrás indicadas constitui o método de Newton, que neste caso é aplicado à minimização de uma função com uma variável. Este método necessita apenas de uma solução inicial x^0 e possui convergência quadrática [Gil81], desde que x^0 adopte um valor suficientemente próximo de x^* . O método de Newton apresenta como principal desvantagem o facto de poder divergir se a solução inicial x^0 não for criteriosamente escolhida. Uma vez que apenas é procurado um ponto com a primeira derivada nula a solução x^* pode ser um mínimo local, máximo local ou um ponto de inflexão (ver Figura 2.1). O significado físico do problema e a análise das características da função f na vizinhança de x^* podem contribuir para uma avaliação da qualidade da solução óptima.

3.2 - MINIMIZAÇÃO DE FUNÇÕES COM n VARIÁVEIS E SEM RESTRIÇÕES

A formulação de um problema de minimização sem restrições de uma função com n variáveis é a seguinte

$$\text{Min. } f(\underline{x}) \quad (3.10)$$

A generalidade dos métodos de resolução do problema (3.10) utilizam o algoritmo indicado na Figura 3.1. A fase que fundamentalmente caracteriza cada um dos métodos é a fase E), sendo habitual distinguir métodos de ordem zero, primeira e segunda ordem de

acordo com o grau das derivadas que são utilizadas no cálculo de $\Delta \underline{x}^q$. A pesquisa unidimensional correspondente à fase F) pode ser efectuada de distintos modos e em certos casos depende do método que é utilizado na fase E). O critério de convergência ou de interrupção do processo iterativo pode também apresentar diversas variantes. Se as derivadas de f estiverem disponíveis deve ser verificada a seguinte condição de optimalidade (ver Capítulo 2)

$$\left\| \nabla f \left(\underline{x}^q \right) \right\| < \varepsilon \quad (3.11)$$

O valor de ε depende da precisão pretendida, podendo em geral assumir valores no intervalo $[10^{-3}, 10^{-6}]$. Qualquer que seja o método de optimização utilizado, deve-se interromper o processo iterativo quando a solução corrente deixar de apresentar variações significativas ou quando for alcançado um número limite de iterações.

3.2.1 - Métodos de ordem zero

Um método de resolução do problema (3.10) é de ordem zero quando não requer o cálculo de derivadas da função f . Estes métodos são portanto adequados à resolução de problemas em que o cálculo das derivadas de f exige grandes recursos informáticos, ou sempre que a função f apresente descontinuidades.

O método de Powell [Pow64] é um método de ordem zero clássico que apresenta algumas características que o tornam vantajoso em certos tipos de problemas. Sendo n o número de variáveis de projecto, o algoritmo relativo ao método de Powell começa por efectuar n pesquisas unidimensionais segundo os eixos coordenados. Em seguida é calculada uma direcção $\Delta \underline{x}^{n+1}$ correspondente à soma dos n progressos $\alpha^q \Delta \underline{x}^q$ anteriormente efectuados. Segundo $\Delta \underline{x}^{n+1}$ é efectuada uma nova pesquisa unidimensional e o vector $\alpha^{n+1} \Delta \underline{x}^{n+1}$ substitui $\alpha^1 \Delta \underline{x}^1$ na lista de n vectores que tem de ser armazenada ao longo do processo iterativo. Segue-se um novo conjunto de n pesquisas unidimensionais segundo as direcções armazenadas e o cálculo de $\Delta \underline{x}^{2n+2}$ como sendo a soma das n progressões anteriores. Em seguida é calculado α^{2n+2} e o vector correspondente a esta última progressão vai substituir o vector mais antigo da lista. Este procedimento é repetido cerca de n vezes e na generalidade dos casos fornece a solução óptima.

O método de Powell revela-se vantajoso nos problemas em que o número de variáveis de projecto é reduzido, não se dispõe de derivadas e cada avaliação da função objectivo tem um custo elevado. Nestes casos a pesquisa unidimensional deve ser efectuada recorrendo ao método de aproximação polinomial, de modo a minimizar o número de avaliações de f . Nestas circunstâncias e desde que não ocorram situações particulares que dificultem a progressão para a solução óptima, o método de Powell revela-se muito eficiente [Van84].

Existe um conjunto de métodos de ordem zero que se baseiam em pesquisas aleatórias efectuadas no espaço das soluções. Estes métodos fundamentam-se em conceitos intuitivos e, de um modo geral, dão origem a algoritmos fáceis de programar. Na Figura 3.5 encontra-se uma versão simplificada no algoritmo correspondente à resolução do problema (3.10) pelo método das pesquisas aleatórias. A função $r(a,b)$ representa um gerador de números aleatórios distribuídos uniformemente no intervalo $[a,b]$.

O método das pesquisas aleatórias pode ser aplicado nos casos em que a função f é descontínua ou quando não está definida em todos os pontos do domínio. A característica mais interessante que o algoritmo da Figura 3.5 apresenta consiste na capacidade de pesquisa do mínimo global de f no domínio $[\underline{x}_{\min}, \underline{x}_{\max}]$. Esta característica é devida ao facto de o processo iterativo não depender de uma solução inicial e de a solução não ser atraída por eventuais mínimos locais. Devido à sua simplicidade o algoritmo da Figura 3.5 pode ser facilmente adaptado de modo a ser possível tirar partido do processamento em paralelo. Uma vez que as pesquisas aleatórias são independentes entre si, pode-se atribuir a cada processador a tarefa de procurar o mínimo global e no fim recolher e comparar a melhor solução obtida por cada um deles. Se não se encontrarem disponíveis meios de cálculo de elevada *performance*, o algoritmo da Figura 3.5 dificilmente resolverá problemas com mais do que poucas dezenas de variáveis. Quando o número de variáveis excede esta ordem de grandeza, para se obter uma solução com uma precisão aceitável torna-se necessário efectuar um número muito elevado de avaliações da função objectivo e o tempo de resolução do problema passa a ser inaceitável.

O algoritmo da Figura 3.5 foi aplicado à minimização da função (2.32), que apresenta diversos mínimos e máximos locais no domínio representado na Figura 2.3 (ver Quadro 2.2). O programa de computador correspondente ao referido algoritmo forneceu como solução o mínimo global (ponto P) com três algarismos significativos correctos. Foram realizadas 10 000 iterações às quais correspondeu num microcomputador um tempo total de resolução inferior a um segundo.

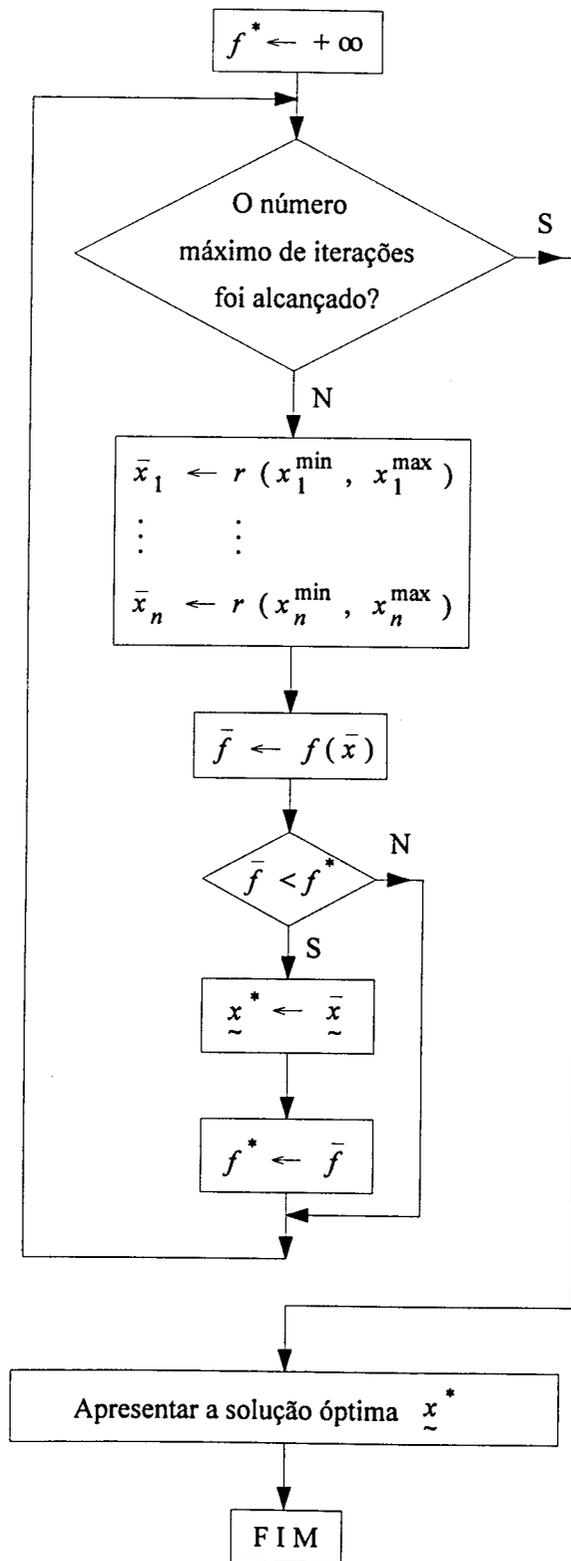


Figura 3.5 - Minimização de $f(x)$ pelo método das pesquisas aleatórias.

Nos casos em que o número de variáveis é elevado, pode ser vantajoso aplicar o algoritmo da Figura 3.5 diversas vezes e gerar poucas soluções aleatórias em cada uma dessas aplicações. Por este motivo as soluções aleatórias devem ser geradas num domínio restrito envolvendo a solução corrente. Nesta versão é necessário preparar uma solução inicial e substituir sucessivamente a solução corrente pela melhor solução que resultou das aplicações anteriores do algoritmo. Nesta versão alternativa o método das pesquisas aleatórias passa a depender da solução inicial e aumenta consideravelmente o risco de convergência para um mínimo local. Para aumentar a sua eficiência é possível modificar o gerador de números aleatórios de modo a aumentar a probabilidade de geração de soluções úteis. Estas modificações podem basear-se em elementos que vão sendo recolhidos durante a progressão da solução corrente [Val86].

Os métodos designados algoritmos genéticos baseiam-se também em pesquisas aleatórias e apresentam algumas características particulares. A sua designação é devida ao facto de procurarem representar a evolução de uma população de acordo com a teoria de Darwin [Gal93]. Os membros dessa população são soluções do problema de optimização que se reproduzem, cruzam e sofrem mutações. Estes três processos dão origem a novas soluções que apresentam características distintas das anteriores e que foram geradas com um determinado grau de aleatoriedade. As soluções são convertidas e manipuladas de acordo com um código binário que procura reproduzir o modo como a informação genética é armazenada nos cromossomas. É necessário definir um critério de avaliação do grau de adaptação ao meio ambiente de cada membro da população. No caso da minimização de funções as soluções mais inadaptadas são as que correspondem a um valor mais elevado da função objectivo. De geração para geração só os membros melhor adaptados devem sobreviver.

Em [Haj92] é apresentada uma aplicação de um algoritmo genético à optimização de estruturas e são indicadas outras referências relativas a este tipo de algoritmos, sendo a mais antiga de 1975. Os resultados publicados em [Haj92] referem-se à optimização de estruturas com um reduzido número de variáveis de projecto e de graus de liberdade. Apesar de se tratarem de problemas de reduzida dimensão, o algoritmo genético apresentou resultados pouco precisos e revelou-se pouco eficiente.

A principal vantagem apresentada pelos algoritmos genéticos e, de um modo geral, pelos métodos baseados em pesquisas aleatórias é a sua maior probabilidade de encontrarem o mínimo global. O facto de só apresentarem uma eficiência aceitável quando a precisão exigida é pequena sugere que se recorra a este tipo de métodos para calcular soluções iniciais destinadas a serem utilizadas em algoritmos de primeira ou segunda ordem.

3.2.2 - Métodos de primeira ordem

A resolução do problema (3.10) pode ser efectuada recorrendo a métodos que apenas requerem o cálculo de primeiras derivadas de f e que são designados métodos de primeira ordem. Nesta classe existem diversos métodos que por sua vez apresentam diversas variantes. O método do máximo declive (*steepest descent*) é o método de primeira ordem que apresenta uma formulação mais simples, sendo por esse motivo apresentado em primeiro lugar. Se as primeiras derivadas de f não se encontrarem disponíveis, é possível efectuar o respectivo cálculo por diferenças finitas.

O método do máximo declive recorre ao algoritmo genérico da Figura 3.1. O critério de convergência utilizado na fase C) consiste na verificação da condição de optimalidade (3.11). Se o máximo número de iterações for alcançado antes de esta condição ser verificada, o processo iterativo deve ser interrompido. Na fase E) o cálculo de $\Delta \tilde{x}^q$ é efectuado com base na solução corrente \tilde{x}^{q-1} , sendo utilizada a seguinte expressão

$$\Delta \tilde{x}^q = -\nabla f\left(\tilde{x}^{q-1}\right) \quad (3.12)$$

De acordo com (3.12), a direcção de pesquisa $\Delta \tilde{x}^q$ é a que numa vizinhança de \tilde{x}^{q-1} apresenta maior declive. O parâmetro de pesquisa unidimensional (α^q) calculado na fase F) deve adoptar um valor tal que a seguinte função \bar{f} seja minimizada

$$\bar{f}(\alpha^q) = f\left(\tilde{x}^{q-1} + \alpha^q \Delta \tilde{x}^q\right) \quad (3.13)$$

Uma vez que $\Delta \tilde{x}^q$ se encontra orientado para a descida, o parâmetro α^q adopta em geral valores positivos.

O método do máximo declive possui uma representação gráfica intuitiva quando é aplicado à minimização de funções com duas variáveis. Na Figura 3.6 encontra-se a trajectória iterativa correspondente à minimização da função (2.32) (ver Capítulo 2). A solução inicial é o ponto $\tilde{x}^0 = (1, -1)$ e verifica-se neste caso uma convergência para o mínimo global. A trajectória iterativa correspondente ao método do máximo declive apresenta direcções consecutivas ortogonais entre si (ver Figura 3.6). Esta característica provoca em muitos casos uma convergência exageradamente lenta [Van84].

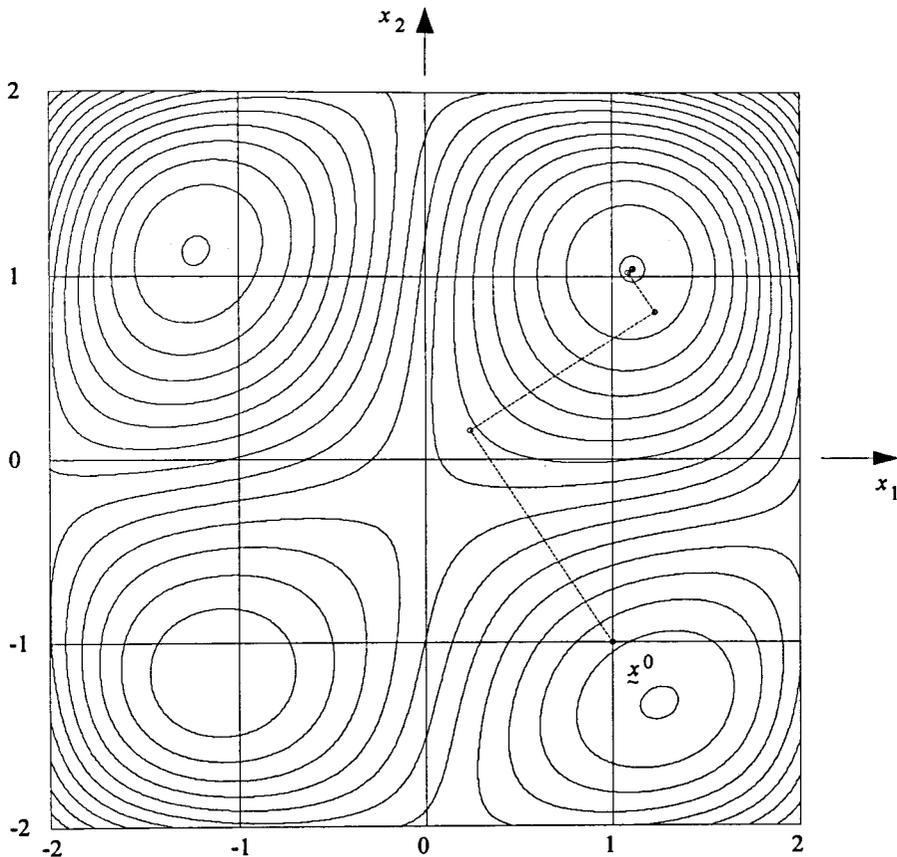


Figura 3.6 - Método do máximo declive (*steepest descent*) - trajectória iterativa.

Com uma pequena modificação é possível melhorar significativamente as características de convergência do método do máximo declive. Essa modificação consiste na substituição de (3.12) pela seguinte expressão também destinada ao cálculo de $\Delta \tilde{x}^q$ na fase E) do algoritmo da Figura 3.1

$$\Delta \tilde{x}^q = -\nabla f(\tilde{x}^{q-1}) + \frac{\|\nabla f(\tilde{x}^{q-1})\|^2}{\|\nabla f(\tilde{x}^{q-2})\|^2} \Delta \tilde{x}^{q-1} \quad (3.14)$$

Esta nova versão do algoritmo baseia-se no método de Fletcher-Reeves [Fle64], que constitui uma adaptação do método dos gradientes conjugados à minimização de uma função com n variáveis [Lue84]. A utilização de (3.14) requer o armazenamento da norma do gradiente de f e da direcção $\Delta \tilde{x}$ correspondentes à iteração anterior. Uma vez que esta informação não se encontra disponível quando o processo iterativo é iniciado, a primeira direcção utilizada no método de Fletcher-Reeves deve ser calculada com a expressão (3.12).

Ao ser adicionada à direcção de máximo declive uma parcela que depende da direcção anterior, as oscilações evidenciadas na Figura 3.6 são significativamente atenuadas [Van84]. Assim se consegue na generalidade dos casos uma mais rápida convergência para a solução óptima. O método de Fletcher-Reeves deve ser sempre preferido em relação ao método do máximo declive, porque é mais eficiente e não implica um acréscimo significativo na complexidade do algoritmo.

Existe um conjunto de métodos, designados genericamente de quasi-Newton ou de métrica variável que procuram aproximar o método de Newton (ver 3.2.3) sem exigirem o cálculo de segundas derivadas. Esses métodos permanecem como métodos de primeira ordem e apresentam características de convergência que os colocam próximo dos métodos de segunda ordem. Os métodos quasi-Newton baseiam-se na construção de uma matriz, que de iteração para iteração aproxima com maior rigor a inversa da matriz Hessiana. Se a função f for quadrática a inversa da matriz Hessiana é obtida num número finito de iterações possibilitando assim a obtenção de uma solução de elevada precisão. Quando a função f não é quadrática a convergência é em geral superlinear. Os métodos quasi-Newton apresentam uma formulação relativamente complexa que se encontra pormenorizadamente exposta em [Lue84]. Nesta classe os métodos mais utilizados são os de Davidon-Fletcher-Powell (DFP) e de Broydon-Fletcher-Goldfarb-Shanno (BFGS) [Van84], cujo desenvolvimento foi efectuado nos anos 60. Mais recentemente foram propostas algumas versões distintas destinadas à resolução de problemas com um elevado número de variáveis [Gil91].

Uma vez que quanto mais elevada for a ordem das derivadas utilizadas mais correcta é a aproximação do problema original, é óbvio que segundo este critério os métodos de primeira ordem são superiores aos de ordem zero. A generalidade dos autores refere que é proibitivo o cálculo e armazenamento de segundas derivadas, e conseqüentemente considera inviável a utilização de métodos de segunda ordem. Nestas circunstâncias os métodos de primeira ordem apresentam-se como um bom compromisso e são indiscutivelmente os mais utilizados na resolução de problemas de optimização não linear.

3.2.3 - Métodos de segunda ordem

A minimização de uma função f com n variáveis pode ser efectuada recorrendo à pesquisa de um ponto estacionário de f (ver 2.2.2). Uma solução com estas características apenas verifica a condição necessária de optimalidade (2.29a) e, por este motivo, pode corresponder a um mínimo local, máximo local ou ponto de sela. Mais adiante será mostrado

que em geral é possível obter soluções óptimas com um método que apenas procura pontos estacionários de f .

Designando por \underline{x} a solução corrente $\left(\underline{x}^{q-1}\right)$ e por $\Delta \underline{x}$ a variação da solução na iteração q $\left(\Delta \underline{x}^q\right)$, a condição de optimalidade (2.29a) que deve ser verificada no ponto $\underline{x} + \Delta \underline{x}$ é a seguinte

$$\nabla f\left(\underline{x} + \Delta \underline{x}\right) = \underline{0} \quad (3.15)$$

À expressão (3.15) corresponde um sistema de n equações não lineares a n incógnitas, que são as componentes do vector $\Delta \underline{x}$. Desenvolvendo em série de Taylor cada uma das equações e conservando apenas os dois primeiros termos obtém-se

$$\left(\frac{\partial f}{\partial x_i}\right)_{(\underline{x} + \Delta \underline{x})} \cong \left(\frac{\partial f}{\partial x_i}\right)_{(\underline{x})} + \sum_{j=1}^n \left[\left(\frac{\partial^2 f}{\partial x_i \partial x_j}\right)_{(\underline{x})} \Delta x_j\right] \quad (i = 1, \dots, n) \quad (3.16)$$

Substituindo o desenvolvimento em série de $\partial f / \partial x_i$ no sistema de equações (3.15) e designando por H a matriz Hessiana de f (ver 2.2.2) obtém-se o seguinte sistema de equações lineares

$$H(\underline{x}) \Delta \underline{x} + \nabla f(\underline{x}) = \underline{0} \quad (3.17)$$

Se a função f cujo ponto estacionário se pretende calcular for quadrática, os termos da série de Taylor que foram ignorados em (3.16) são nulos. Neste caso particular o sistema de equações lineares (3.17) fornece um vector $\Delta \underline{x}$ que conduz imediatamente à solução óptima pretendida, qualquer que seja \underline{x}

$$\underline{x}^* = \underline{x} + \Delta \underline{x} \quad (3.18)$$

Se a função f não for quadrática, o sistema de equações (3.17) fornece um valor aproximado de $\Delta \underline{x}$, que na generalidade dos casos conduz a uma nova solução corrente mais próxima da solução óptima. Para se obter uma solução final rigorosa, a resolução do sistema de equações (3.17) e a actualização da solução corrente (3.18) devem ser integrados num processo iterativo. O algoritmo genérico da Figura 3.1 pode ser de novo adoptado, recebendo neste caso a designação de método de Newton [Aro89]. O critério de convergência correspondente à fase C) (ver Figura 3.1) pode ser o indicado em (3.11), porque este método

requer que sejam calculadas as primeiras derivadas de f . Na fase E) o cálculo do vector Δx^q requer a resolução do seguinte sistema de equações lineares

$$H\left(\tilde{x}^{q-1}\right)\Delta\tilde{x}^q + \nabla f\left(\tilde{x}^{q-1}\right) = 0 \quad (3.19)$$

O método de Newton é um método de segunda ordem, porque na matriz Hessiana H figuram segundas derivadas de f . Se não for possível obter as expressões das segundas derivadas, o seu cálculo pode ser efectuado por diferenças finitas. O método de Newton torna-se significativamente mais robusto se em cada iteração for efectuada uma pesquisa unidimensional que forneça o valor óptimo de α^q (fases F) e G) do algoritmo da Figura 3.1). Este facto é evidenciado nas Figuras 3.7 e 3.8, que representam o comportamento do algoritmo relativo ao método de Newton. Nessas figuras encontra-se representado por intermédio de uma codificação por cores o tipo de convergência ou divergência que ocorre quando o algoritmo é executado a partir de soluções iniciais distintas. Ambos os casos correspondem à minimização da função (2.32) (ver 2.2.2), cujos pontos estacionários são indicados no Quadro 2.2. Nas Figuras 3.7 e 3.8 encontram-se também representadas as isocurvas da função (2.32), de modo a permitir a visualização das posições dos dois mínimos locais, do ponto de sela e dos dois máximos locais existentes no domínio considerado (ver também as Figuras 2.2 e 2.3). O algoritmo correspondente ao método de Newton foi executado um número de vezes igual ao número de pixels da imagem. A cada pixel encontra-se associada a solução inicial correspondente à respectiva posição no domínio representado. A cor atribuída a cada pixel depende do ponto estacionário que foi alcançado após um máximo de 50 iterações. Nos casos em que não ocorreu convergência não foi atribuída qualquer cor.

A Figura 3.7 corresponde ao método de Newton sem pesquisa unidimensional, verificando-se que na generalidade dos casos a solução final é o ponto estacionário mais próximo da solução inicial. Exceptuam-se algumas regiões distantes de pontos estacionários às quais corresponde a divergência do processo iterativo ou a convergência para uma solução final muito afastada da solução inicial. A Figura 3.8 corresponde ao método de Newton com pesquisa unidimensional e com a particularidade de α^q poder adoptar valores negativos. Foi considerado que, sempre que $\Delta\tilde{x}^q$ se encontra orientado para um máximo local, é atribuído a α^q um valor negativo, que faz com que a solução se desloque em sentido contrário. Por este motivo deixa de ocorrer convergência para os máximos locais. Com a introdução da pesquisa unidimensional aumenta significativamente a probabilidade de convergência para um dos mínimos locais.

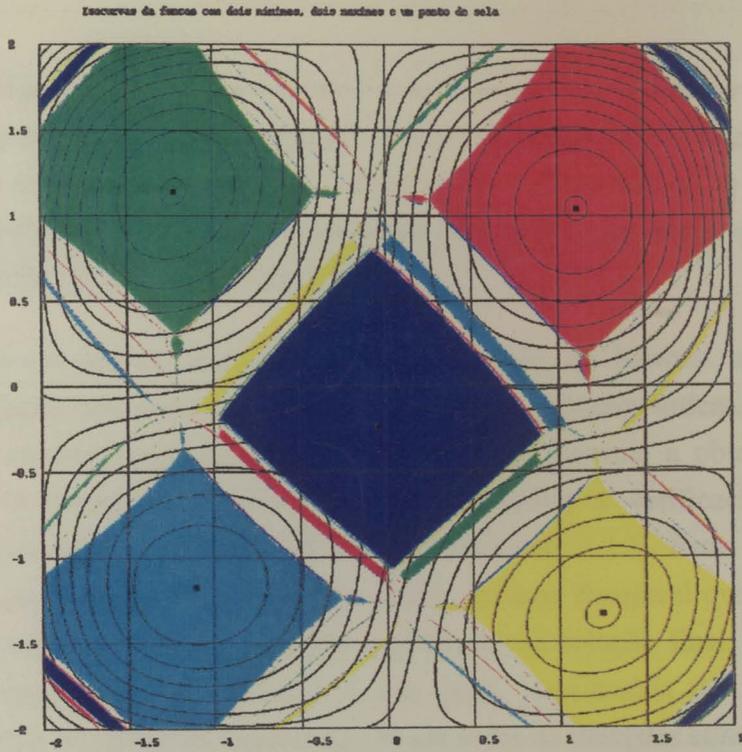


Figura 3.7 - Mapa de convergência correspondente ao método de Newton sem pesquisa unidimensional.

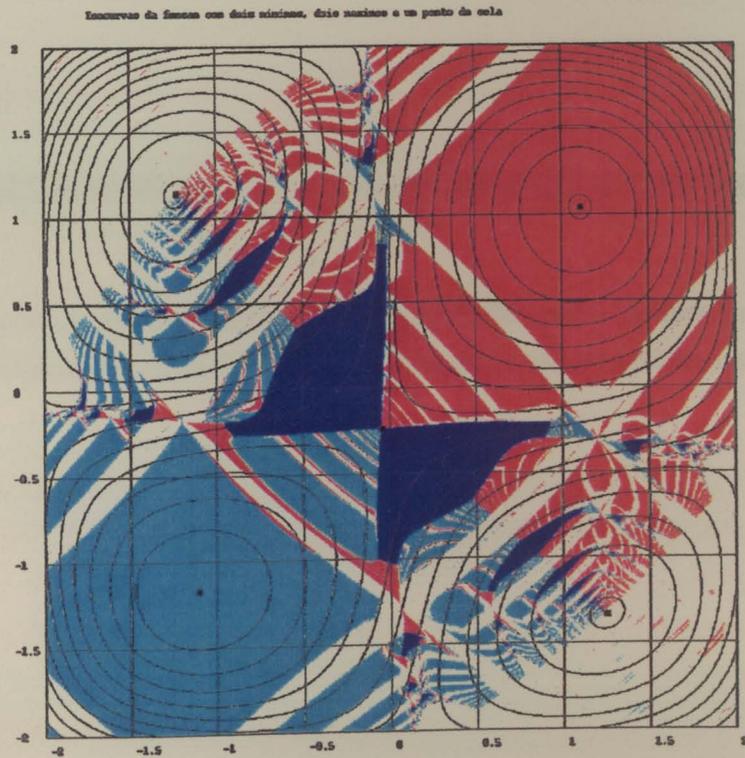


Figura 3.8 - Mapa de convergência correspondente ao método de Newton com pesquisa unidimensional.

Quer na Figura 3.7, quer na Figura 3.8 verifica-se que, quando a solução inicial se encontra distanciada das possíveis soluções finais, o sucesso ou o fracasso do processo iterativo torna-se dependente de pequenas variações da solução inicial. O estudo deste caso particular veio confirmar a importância da selecção de uma solução inicial que se encontre tão próximo quanto possível da solução final pretendida.

Pelo facto de ser de segunda ordem, o método de Newton apresenta algumas vantagens em relação aos métodos de ordem zero e de primeira ordem. De todas a mais significativa é a convergência quadrática [Gil81], porque permite a obtenção de soluções muito precisas. O cálculo da matriz Hessiana possibilita a verificação das condições suficientes de optimalidade relativas a um mínimo local (ver 2.2.2). Estas condições exigem que a matriz Hessiana seja positiva definida, sendo portanto desejável que os algoritmos de optimização efectuem esta verificação após a obtenção da solução final. A necessidade de calcular e armazenar segundas derivadas é, em geral, considerada proibitiva [Aro89]. A resolução do sistema de equações lineares (3.19) em todas as iterações contribui também para a necessidade de se dispor de meios informáticos relativamente poderosos. A constante evolução da informática e a utilização de algoritmos eficientes viabilizaram a resolução pelo método de Newton de problemas de optimização com um elevado número de variáveis. Estas questões serão pormenorizadamente apresentadas mais adiante neste trabalho.

3.3 - MINIMIZAÇÃO DE FUNÇÕES COM n VARIÁVEIS E COM RESTRIÇÕES

Na resolução de problemas de optimização estão quase sempre presentes restrições igualdade e/ou desigualdade. A formulação genérica deste tipo de problemas é a seguinte

$$\text{Min. } f(\underline{x}) \quad (3.20a)$$

s.a.

$$\underline{g}(\underline{x}) \leq \underline{0} \quad (3.20b)$$

$$\underline{h}(\underline{x}) = \underline{0} \quad (3.20c)$$

Os métodos de optimização referidos na secção anterior não contemplam a presença de restrições, sendo portanto necessário adaptá-los a este tipo de problemas ou proceder ao

desenvolvimento de novos métodos. Os métodos de minimização sem restrições podem ser utilizados desde que seja efectuada uma prévia transformação do problema com restrições num problema sem restrições. Quando um método apenas contempla a presença de restrições igualdade, é em geral possível incluir restrições desigualdade, recorrendo à adição de variáveis de desvio (ver secção 2.1.4). O caso contrário pode ser resolvido com a substituição de cada restrição igualdade por um par de restrições desigualdade (ver secção 2.1.3). Alguns métodos recorrem a uma estratégia baseada no conjunto das restrições activas (*active set*) [Gil81]. Esta estratégia refere-se às restrições desigualdade e baseia-se na identificação das restrições que para a solução corrente se encontram activas. É necessário dotar o algoritmo com a capacidade de ao longo do processo iterativo ir acrescentando e retirando restrições ao referido conjunto.

Apresenta-se em seguida uma breve descrição de alguns métodos de minimização com restrições, tendo sido efectuada a distinção entre métodos primais e métodos de transformação [Bel85]. Os métodos primais são aqueles que abordam directamente o problema de minimização com restrições, recebendo esta designação pelo facto de não considerarem variáveis duais (multiplicadores de Lagrange). Os métodos de transformação baseiam-se na substituição do problema de minimização com restrições num problema equivalente mas sem restrições. Após a transformação passa a ser possível recorrer a um dos métodos de minimização sem restrições referidos na secção anterior. A transformação num problema sem restrições pode ser efectuada recorrendo a funções penalidade, ao Lagrangeano ou ao Lagrangeano aumentado. Alguns métodos requerem que sejam efectuadas sucessivas minimizações sem restrições, acompanhadas de constantes actualizações de certos parâmetros.

3.3.1 - Métodos primais

Apresenta-se em seguida uma breve descrição dos métodos das direcções admissíveis, gradiente projectado, gradiente reduzido generalizado, programação linear sequencial e programação quadrática sequencial. Serão também referidas as principais vantagens e desvantagens de cada um deles. Devido à possibilidade de se efectuar diferentes combinações das inúmeras técnicas de optimização, revela-se praticamente impossível referir exhaustivamente todos os métodos que se encontram presentemente disponíveis. Os métodos que em seguida são apresentados foram seleccionados pelo facto de serem os mais frequentemente utilizados na optimização de estruturas.

Método das direcções admissíveis

A minimização de uma função sujeita a um conjunto de restrições desigualdade pode ser efectuada com um algoritmo que procure diminuir sucessivamente a função objectivo, mantendo a solução corrente dentro da região admissível. A solução inicial tem obviamente de ser admissível, sendo necessário recorrer a um algoritmo auxiliar quando esse facto não se verifica. É nestes princípios que se baseia o método das direcções admissíveis, cujo desenvolvimento remonta a 1960 [Zou60]. Recentemente tem sido utilizado na resolução de problemas de optimização por intermédio do programa DOT [Van89a].

O método das direcções admissíveis adopta o algoritmo genérico da Figura 3.1, sendo as suas características particulares incluídas na fase E). De acordo com o que foi atrás referido, a direcção $\Delta \underline{x}^q$ deve corresponder à máxima diminuição da função objectivo, com a restrição de a solução corrente permanecer admissível. O cálculo de $\Delta \underline{x}^q$ constitui um subproblema em que figuram os gradientes da função objectivo e das restrições activas. Este subproblema pode ser formulado como um programa linear [Van84] e conseqüentemente pode ser resolvido pelo método *simplex*. Para atender a uma eventual curvatura da superfície exterior da região admissível, deve-se evitar que a direcção $\Delta \underline{x}^q$ seja ortogonal a ∇g_j . Assim se evita um eventual deslocamento para fora da região admissível (ver Figura 3.9).

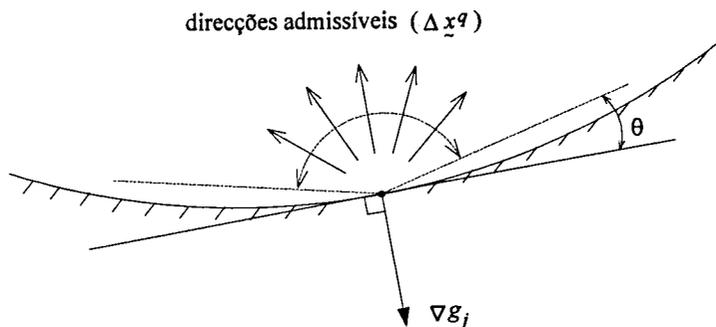


Figura 3.9 - Direcções admissíveis - influência do parâmetro θ .

Se o valor do parâmetro θ não for criteriosamente seleccionado, a eficiência e a robustez do algoritmo podem ser seriamente afectadas. Uma vez que no cálculo de $\Delta \underline{x}^q$ apenas são consideradas as restrições activas, existe também a necessidade de definir uma tolerância numérica associada à decisão de incluir ou excluir uma restrição do conjunto das restrições activas. O valor desta tolerância depende do problema e de cada restrição, devendo

assumir um valor inicial relativamente elevado. Ao longo do processo iterativo o valor da tolerância deve ser sucessivamente diminuído. Se os valores destes parâmetros não forem convenientemente escolhidos, poderá ocorrer um fenómeno designado zigzagueamento (*zigzagging*), que consiste no facto de uma restrição repetidamente entrar e sair do conjunto das restrições activas. Esta dependência dos valores atribuídos aos diversos parâmetros faz com que o método das direcções admissíveis seja considerado pouco robusto [Fle82]. Este método possui ainda o inconveniente de ser dificilmente adaptável a problemas com restrições igualdade [Aro89].

Método do gradiente projectado

O método do gradiente projectado é semelhante ao método das direcções admissíveis, tendo sido proposto por Rosen em 1961 [Ros61]. A motivação para o desenvolvimento deste novo método foi a de evitar a necessidade de resolver um programa linear em cada iteração. No método do gradiente projectado a direcção Δx^q é obtida projectando ∇f sobre o hiperplano tangente à região admissível. É em seguida efectuada uma pesquisa unidimensional destinada a minimizar f na direcção Δx^q . Se a superfície exterior da região admissível for não linear e convexa, a nova solução corrente é inadmissível (ver Figura 3.9 para $\theta = 0$). Torna-se então necessário deslocar iterativamente a solução corrente para a superfície da região admissível. Os algoritmos utilizados nesta fase correctiva revelam-se em geral pouco robustos [Bel85] e podem comprometer significativamente a eficiência do método. Por estes motivos o método do gradiente projectado é em geral pouco utilizado.

Método do gradiente reduzido generalizado

Em [Aba69] é apresentada uma generalização do método do gradiente reduzido de Wolfe [Wol63]. Na versão inicial o método tem como objectivo a minimização de funções sujeitas apenas a restrições igualdade. As variáveis do problema são distribuídas por dois grupos, o das variáveis independentes e o das variáveis dependentes. Neste último o número de variáveis é igual ao número de restrições igualdade. É possível calcular uma direcção no espaço das variáveis independentes, que simultaneamente seja uma direcção de descida e mantenha a solução corrente sobre uma aproximação linear da região admissível. Se, pelo facto de ter sido considerada uma aproximação linear, a solução deixar de ser admissível, deve-se recorrer a um método iterativo que desloque a solução para um ponto em que as restrições igualdade sejam de novo respeitadas. A generalização que permite a aplicação do método do gradiente reduzido a problemas com restrições desigualdade consiste no recurso a variáveis de desvio ou a uma estratégia em que apenas é considerado o conjunto das restrições

activas (*active set strategy*). O método do gradiente reduzido nesta versão que contempla a presença de restrições desigualdade é designado método do gradiente reduzido generalizado. Este método apresenta muitas semelhanças com o método do gradiente projectado e, por esse motivo, partilha com ele as principais vantagens e desvantagens [Aro89].

Programação linear sequencial

A resolução do programa matemático (3.20) pode basear-se numa aproximação linear da função objectivo e restrições. Efectuando o desenvolvimento em série de Taylor de cada uma das expressões e ignorando os termos de ordem superior à primeira, obtém-se o seguinte programa matemático

$$\text{Min. } f\left(\underset{\sim}{x}^{q-1}\right) + \nabla f\left(\underset{\sim}{x}^{q-1}\right) \Delta \underset{\sim}{x}^q \quad (3.21a)$$

s.a.

$$g_j\left(\underset{\sim}{x}^{q-1}\right) + \nabla g_j\left(\underset{\sim}{x}^{q-1}\right) \Delta \underset{\sim}{x}^q \leq \underset{\sim}{0} \quad (j = 1, \dots, m) \quad (3.21b)$$

$$h_k\left(\underset{\sim}{x}^{q-1}\right) + \nabla h_k\left(\underset{\sim}{x}^{q-1}\right) \Delta \underset{\sim}{x}^q = \underset{\sim}{0} \quad (k = 1, \dots, l) \quad (3.21c)$$

O desenvolvimento em série de Taylor é efectuado no ponto $\underset{\sim}{x}^{q-1}$ e as variáveis do problema passam a ser as componentes do vector $\Delta \underset{\sim}{x}^q$. O programa matemático (3.21) é um programa linear, que pode ser facilmente resolvido recorrendo por exemplo ao método *simplex*. Uma vez que na generalidade dos casos a aproximação considerada não é exacta, é conveniente resolver o programa linear mais do que uma vez, deslocando sucessivamente a solução corrente em direcção à solução óptima. Este procedimento pode ser integrado no algoritmo da Figura 3.1, correspondendo a resolução do programa linear (3.21) à fase E). O método assim definido é designado programação linear sequencial ou método do plano de corte de Kelley [Kel60].

Durante o processo iterativo podem ocorrer situações em que a aproximação linear considerada não permita a obtenção de direcções $\Delta \underset{\sim}{x}^q$ que conduzam a uma convergência para a solução óptima. Se a aproximação linear não reproduzir com suficiente rigor o problema original, o programa linear (3.21) pode não ser limitado (*unbounded*) [Van84]. Esta situação pode ser evitada acrescentando ao programa linear (3.21) um conjunto de restrições que impõem limites à variação da solução (*move limits*). Com estas restrições adicionais a solução corrente é obrigada a permanecer numa vizinhança do ponto em que foi efectuado o

desenvolvimento em série de Taylor. Nessa vizinhança a aproximação linear reproduz com suficiente rigor o problema original.

Na resolução de problemas práticos depara-se quase sempre com a dificuldade de definição da grandeza dos limites de variação da solução. Os respectivos valores dependem do tipo de problema, do tipo de variável e da iteração. Durante o processo iterativo é conveniente que os valores dos limites de variação da solução sejam sucessivamente diminuídos [Van84]. Se os seus valores se tornarem demasiado pequenos, a presença destas restrições adicionais pode fazer com que o programa linear (3.21) não possua nenhuma solução admissível. O algoritmo deve nestes casos aumentar automaticamente os valores dos limites de variação da solução.

A programação linear sequencial dificilmente pode ser utilizada na resolução de problemas práticos sem a intervenção de um utilizador experiente. A sua principal tarefa consiste na definição dos valores iniciais dos limites de variação da solução e dos factores associados ao seu decréscimo ao longo do processo iterativo. Se a estratégia previamente definida não se revelar conveniente, poderá ser necessário intervir durante o processo iterativo. A experiência adquirida na resolução de problemas semelhantes é nestes casos muito importante. As dificuldades associadas à definição dos limites de variação da solução constituem a principal desvantagem da programação linear sequencial como método de resolução de problemas genéricos de optimização.

Programação quadrática sequencial

Os métodos que recebem a designação genérica de programação quadrática sequencial apresentam diversas variantes baseadas no algoritmo da Figura 3.1. A principal característica que essas variantes possuem em comum consiste no facto de o vector $\Delta \tilde{x}^q$ (fase E) ser a solução de um programa quadrático. Se o problema de optimização consistir na minimização de uma função f sujeita a uma única restrição igualdade h , o programa quadrático que fornece a direcção $\Delta \tilde{x}^q$ é o seguinte [Kao94]

$$\text{Min. } \left[\nabla f \left(\tilde{x}^{q-1} \right) + \lambda \nabla h \left(\tilde{x}^{q-1} \right) \right] \Delta \tilde{x}^q + \frac{1}{2} \Delta \tilde{x}^{qT} \left[\nabla^2 f \left(\tilde{x}^{q-1} \right) + \lambda \nabla^2 h \left(\tilde{x}^{q-1} \right) \right] \Delta \tilde{x}^q \quad (3.22a)$$

s.a.

$$h \left(\tilde{x}^{q-1} \right) + \nabla h \left(\tilde{x}^{q-1} \right) \Delta \tilde{x}^q = 0 \quad (3.22b)$$

Em (3.22a) $\nabla^2 f$ e $\nabla^2 h$ são as matrizes Hessianas de f e h respectivamente. As considerações em seguida apresentadas são facilmente generalizáveis a programas matemáticos com múltiplas restrições igualdade e desigualdade.

O anulamento do gradiente do Lagrangeano associado ao programa matemático (3.22) constitui um sistema de equações lineares, cuja solução fornece o vector $\Delta \underline{x}^q$. Este sistema de equações coincide com o que é resolvido em cada iteração relativa ao método de Newton (ver Capítulo 4). Conclui-se portanto que a solução do programa quadrático (3.22) $\left(\Delta \underline{x}^q \right)$ coincide com a direcção que é utilizada no algoritmo relativo ao método de Newton.

A programação quadrática sequencial é referida nalguma bibliografia como programação quadrática recursiva [Lue84] ou com a designação de método do Lagrangeano projectado baseado em programação quadrática [Gil81]. Nas aplicações desta classe de métodos é habitual evitar o cálculo das segundas derivadas que figuram em (3.22a). Com esse fim é efectuada a substituição de $(\nabla^2 f + \lambda \nabla^2 h)$ por uma matriz que vai sendo actualizada de iteração para iteração. Esta operação pode ser realizada com a fórmula de actualização BFGS (ver secção 3.2.2.) [Pow77]. Esta aproximação compromete a convergência quadrática que o método possuiria se fosse calculada uma direcção $\Delta \underline{x}^q$ coincidente com a que é utilizada no método de Newton. Para resolver o problema de minimização associado à pesquisa unidimensional (*line search*) é necessário definir uma função auxiliar que permita avaliar a qualidade da solução para diferentes valores de α^q [Van84].

Em [Hin91] são apresentados diversos problemas de optimização de estruturas cuja solução foi obtida recorrendo à programação quadrática sequencial, tendo sido utilizada uma subrotina da biblioteca NAG [NAG91]. Na generalidade dos casos estes métodos revelam-se muito precisos e robustos, sendo estas qualidades conseguidas com algum sacrifício da eficiência [Ble92].

3.3.2 - Métodos de transformação

Os métodos de minimização sem restrições podem ser utilizados na resolução do problema (3.20) desde que este seja previamente transformado num problema equivalente mas sem restrições. Depende do método de transformação utilizado o facto de a solução do problema equivalente coincidir com a do problema original ou de ser apenas uma aproximação de precisão variável.

As transformações que podem ser efectuadas recorrem a funções penalidade, ao Lagrangeano ou a uma combinação destes dois tipos de funções. Com o objectivo de obter um algoritmo mais eficiente, alguns métodos exploram a dualidade que resulta da utilização do Lagrangeano.

Minimização sem restrições sequencial

O programa matemático (3.20) pode ser substituído pela minimização de uma pseudo função objectivo que traduz um custo que é agravado quando alguma restrição é violada. A seguinte função possui essas características [Van84]

$$\Phi(\tilde{x}) = f(\tilde{x}) + r \sum_{j=1}^m \left\{ \max \left[0, g_j(\tilde{x}) \right] \right\}^2 + r \sum_{k=1}^l \left[h_k(\tilde{x}) \right]^2 \quad (3.23)$$

Em (3.23) r é um factor que deve assumir um valor muito elevado, de modo a provocar um significativo agravamento do custo, mesmo que as restrições sejam apenas ligeiramente violadas. No caso das restrições desigualdade, só existe agravamento do custo se algum $g_j(\tilde{x})$ assumir um valor positivo. Por estes motivos, a minimização sem restrições da função Φ apresenta em geral uma solução \tilde{x} situada dentro da região admissível ou muito próximo dela.

Se a solução inicial se encontrar distanciada da solução óptima e o parâmetro r adoptar um valor demasiado elevado, a minimização da função Φ é um problema muito mal condicionado. Por este motivo é aconselhável atribuir a r um valor relativamente baixo e efectuar uma primeira minimização de Φ , obtendo assim uma solução mais próxima da solução óptima de (3.20). Pelo facto de o valor de r ser baixo, esta solução situa-se ainda significativamente afastada da região admissível. Uma vez que esta solução se encontra agora mais próxima da solução óptima de (3.20), é possível repetir a minimização de Φ com um valor de r mais elevado. Este procedimento pode ser repetido diversas vezes até r adoptar um valor de tal forma elevado que a solução seja forçada a situar-se muito próximo da região admissível. Este algoritmo é genericamente designado minimização sem restrições sequencial (*sequential unconstrained minimization techniques* - SUMT) [Fia68].

A utilização da função (3.23) no algoritmo SUMT conduz em geral a uma sucessão de soluções inadmissíveis. Na generalidade dos casos a solução final encontra-se também significativamente afastada da região admissível. Por este motivo a função que foi adicionada a $f(\tilde{x})$ em (3.23) é designada função penalidade exterior. Tendo em vista a eliminação da referida desvantagem, é possível recorrer a uma função ligeiramente diferente da (3.23), que

possui a característica de penalizar as soluções que se aproximam do limite da região admissível. A pesquisa unidimensional (*line search*) deve ser adaptada de modo a serem ignoradas eventuais soluções inadmissíveis. Deste modo o algoritmo SUMT produz uma sucessão de soluções admissíveis que inclui a solução final. Estas funções alternativas são designadas funções de penalidade interior ou funções barreira [Lue84].

O algoritmo SUMT associado a funções penalidade ou barreira apresenta na prática sérios inconvenientes. De todos o mais grave é a dificuldade de definição do ritmo de crescimento de r , de modo a não tornar a minimização sem restrições prematuramente mal condicionada [Bel85]. A obtenção de soluções precisas só é possível para valores muito elevados de r . Uma vez que nestas circunstâncias o problema se torna mal condicionado, o algoritmo SUMT apenas consegue fornecer soluções com um erro consideravelmente elevado. Devido à ausência de precisão e de eficiência [Bel82], os métodos de transformação baseados apenas em funções penalidade ou barreira são considerados obsoletos quando comparados com outros métodos mais recentes [Pow81].

Método de Lagrange

Os métodos baseados em multiplicadores de Lagrange procuram a solução do problema (3.20) após sua transformação num problema equivalente sem restrições. Esta transformação pode ser efectuada recorrendo ao Lagrangeano (ver Capítulo 2), porque um dos seus pontos estacionários corresponde à solução do problema (3.20). Este ponto é também um ponto de sela do Lagrangeano. Um método que procede à sua determinação para resolver (3.20) recebe a designação de primal-dual [Aro91]. Se no problema de optimização apenas estiverem presentes restrições igualdade o respectivo Lagrangeano é o seguinte

$$L(\underline{x}, \underline{\lambda}^h) = f(\underline{x}) + \sum_{k=1}^l [\lambda_k^h h_k(\underline{x})] \quad (3.24)$$

O cálculo do ponto estacionário de L corresponde à resolução do seguinte sistema de $n + l$ equações não lineares em que figuram como incógnitas as n componentes de \underline{x} e as l componentes de $\underline{\lambda}^h$

$$\nabla_{\underline{x}} f + \sum_{k=1}^l (\lambda_k^h \nabla_{\underline{x}} h_k) = \underline{0} \quad (3.25a)$$

$$\underline{h} = \underline{0} \quad (3.25b)$$

O cálculo da solução de um problema de optimização por intermédio da resolução do sistema de equações (3.25) recebe a designação de método de Lagrange [Aro91]. Se na

resolução do sistema de equações (3.25) for utilizado o método de Newton, o método de optimização daí resultante recebe a designação de Lagrange-Newton [Fle88].

A inclusão dos multiplicadores de Lagrange no conjunto de variáveis do problema faz com que o seu número passe de n a $n + l$. Este facto constitui uma evidente desvantagem em relação a outros métodos em que apenas são consideradas como variáveis as n componentes de x (e.g., algoritmos SUMT). No entanto, a presença dos multiplicadores de Lagrange permite o desenvolvimento de algoritmos mais robustos, mais precisos e que disponibilizam mais informação relativa à solução óptima.

As restrições desigualdade podem ser incluídas num algoritmo que apenas contempla a presença de restrições igualdade recorrendo a variáveis de desvio (ver Capítulo 2) ou ao conjunto das restrições activas (*active set*). A utilização de m variáveis de desvio conjuntamente com o método de Lagrange-Newton faz com que o número de variáveis do problema passe a ser $n + l + 2m$. Pelo facto de ter sido o método utilizado no presente trabalho, é apresentado no Capítulo 4 um desenvolvimento pormenorizado do método de Lagrange-Newton com o recurso a variáveis de desvio nas restrições desigualdade. Nos Capítulos seguintes é apresentado o respectivo algoritmo e alguns resultados da sua aplicação à optimização de estruturas. Apesar das desvantagens referidas na bibliografia, foi possível desenvolver um algoritmo robusto, eficiente e preciso com o qual se solucionaram problemas de optimização com um elevado número de variáveis e restrições.

Método do Lagrangeano aumentado

Para evitar que a presença dos multiplicadores de Lagrange conduza a um aumento significativo do número de variáveis do problema, Hestenes e Powell desenvolveram em 1969 um método baseado na seguinte função auxiliar [Hes69] [Pow69]

$$\Phi(\tilde{x}) = f(\tilde{x}) + \sum_{k=1}^l [\lambda_k h_k(\tilde{x})] + r \sum_{k=1}^l [h_k(\tilde{x})]^2 \quad (3.26)$$

Esta função refere-se a um problema de minimização sujeito apenas a restrições igualdade. A expressão (3.26) foi obtida adicionando ao Lagrangeano (3.24) um termo de penalidade que figura em (3.23). Por este motivo a função Φ é designada Lagrangeano aumentado. A função (3.26) é utilizada num algoritmo que procura reunir as principais vantagens do método de Lagrange e dos algoritmos SUMT e que é designado método do Lagrangeano aumentado (*augmented Lagrangean method* - ALM). Este método é iniciado com a selecção dos valores iniciais dos multiplicadores de Lagrange λ_k e do factor de penalidade r . Os valores iniciais dos λ_k podem ser nulos e o valor inicial de r deve ser não

nulo e pouco elevado. Em seguida procede-se a uma minimização sem restrições da função Φ , considerando apenas como variáveis as n componentes de \tilde{x} . Para que o resultado desta minimização se encontre próximo da solução \tilde{x}^* , é necessário que os multiplicadores de Lagrange λ_k se encontrem também próximos dos correspondentes à solução óptima ou que r possua um valor muito elevado. Esta última opção deve ser evitada porque conduz a um mau condicionamento da função Φ . Recorrendo à seguinte expressão [Van84], é possível calcular um valor mais correcto dos multiplicadores de Lagrange, com base no resultado da anterior minimização de Φ

$$\lambda_k^q = \lambda_k^{q-1} + 2r h_k \left(\tilde{x}^{q-1} \right) \quad (k = 1, \dots, l) \quad (3.27)$$

A minimização da função Φ e a actualização dos multiplicadores de Lagrange devem ser efectuadas diversas vezes, constituindo assim um processo iterativo em que q representa o número da iteração. É aconselhável que o parâmetro r seja sucessivamente incrementado, não devendo contudo assumir valores muito elevados para evitar que a função Φ se torne mal condicionada. Neste processo iterativo os multiplicadores de Lagrange convergem linearmente para os valores correspondentes à solução óptima $\tilde{\lambda}^*$ [Gil81]. Este facto faz com que a solução do problema de optimização $\left(\tilde{x}^q \right)$ apresente também convergência linear para a solução óptima $\left(\tilde{x}^* \right)$.

Em [Aro91] são apresentadas algumas alternativas para as expressões (3.26) e (3.27), que permitem uma aplicação do método do Lagrangeano aumentado à resolução de problemas de minimização com restrições igualdade e desigualdade. São também sugeridas estratégias alternativas de modo a melhorar as características de convergência do método. De um modo geral verifica-se que a um aumento da eficiência está associada uma diminuição da robustez e da precisão e que a melhoria das características de convergência só é conseguida se forem consideradas aproximações que requerem o cálculo de derivadas de ordem superior.

Métodos duais

O conceito de dualidade é em seguida apresentado por intermédio da sua aplicação a um problema de minimização com duas restrições igualdade. O programa matemático aqui considerado coincide com o problema (2.44) apresentado no Capítulo 2, que possui a seguinte formulação.

$$\text{Min. } f(\underline{x}) = x_1^2 + x_2^2 + x_3^2 \quad (3.28a)$$

s.a.

$$h_1(\underline{x}) = -x_1 - x_2 - x_3 + 2 = 0 \quad (3.28b)$$

$$h_2(\underline{x}) = x_1 - x_2 - x_3 + 2 = 0 \quad (3.28c)$$

O Lagrangeano correspondente a este programa matemático é o seguinte

$$L(\underline{x}, \underline{\lambda}) = x_1^2 + x_2^2 + x_3^2 + \lambda_1(-x_1 - x_2 - x_3 + 2) + \lambda_2(x_1 - x_2 - x_3 + 2) \quad (3.29)$$

Uma vez que a solução óptima $(\underline{x}^*, \underline{\lambda}^*)$ é um ponto estacionário do Lagrangeano, as derivadas de L em ordem a x_1 , x_2 e x_3 têm de ser nulas.

$$\frac{\partial L}{\partial x_1} = 0 \Rightarrow 2x_1 - \lambda_1 + \lambda_2 = 0 \quad (3.30a)$$

$$\frac{\partial L}{\partial x_2} = 0 \Rightarrow 2x_2 - \lambda_1 - \lambda_2 = 0 \quad (3.30b)$$

$$\frac{\partial L}{\partial x_3} = 0 \Rightarrow 2x_3 - \lambda_1 - \lambda_2 = 0 \quad (3.30c)$$

As expressões (3.30) possuem uma característica muito importante e que consiste no facto de em cada equação apenas figurar uma das n variáveis x_i . Este facto ocorre sempre que a função objectivo e as restrições são constituídas por um somatório de funções que só dependem de uma das variáveis.

$$f(\underline{x}) = f_1(x_1) + \dots + f_n(x_n) \quad (3.31)$$

As funções que apresentam estas características são designadas separáveis. Quando a função objectivo e as restrições são funções separáveis, em cada uma das equações (3.30) figura uma só variável x_i , cujo valor pode ser facilmente calculado desde que sejam conhecidos os valores dos λ_k ($k = 1, \dots, l$). Neste exemplo simples é até possível explicitar cada uma das variáveis x_i em função das componentes de $\underline{\lambda}$

$$x_1 = 0.5\lambda_1 - 0.5\lambda_2 \quad (3.32a)$$

$$x_2 = 0.5\lambda_1 + 0.5\lambda_2 \quad (3.32b)$$

$$x_3 = 0.5\lambda_1 + 0.5\lambda_2 \quad (3.32c)$$

Substituindo os x_i ($i = 1, \dots, n$) fornecidos pelas expressões (3.32) no Lagrangeano (3.29), obtém-se uma função L' que é designada função objectivo dual e que depende apenas dos λ_k ($k = 1, \dots, l$)

$$L'(\lambda) = -0.75\lambda_1^2 - 0.75\lambda_2^2 - 0.5\lambda_1\lambda_2 + 2\lambda_1 + 2\lambda_2 \quad (3.33)$$

A função (3.33) encontra-se representada graficamente na Figura 3.10. De acordo com a teoria da dualidade [Lue84], o resultado da maximização de $L'(\lambda_1, \lambda_2)$ coincide com os multiplicadores de Lagrange λ^* correspondentes à solução óptima do programa matemático (3.28). Neste exemplo, os valores dos multiplicadores de Lagrange são $(\lambda_1^*, \lambda_2^*) = (1, 1)$. Substituindo estes valores em (3.32), obtém-se $(x_1^*, x_2^*, x_3^*) = (0, 1, 1)$. Estes resultados constituem a solução óptima do programa matemático (3.28) e coincidem com a solução apresentada na secção 2.2.3.

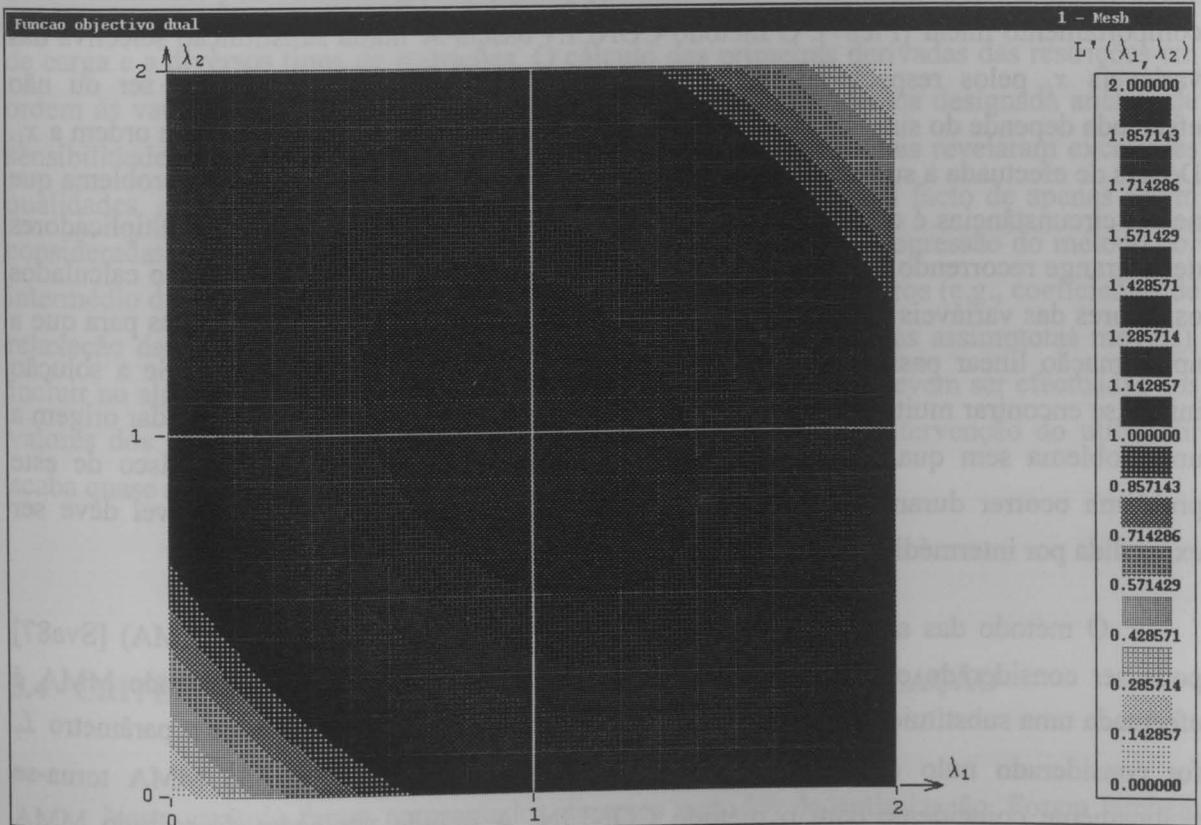


Figura 3.10 - Função objectivo dual (3.33).

A maximização de $L'(\lambda)$ pode ser efectuada com uma variante do método do máximo declive (*steepest descent*) designado método da máxima ascensão (*steepest ascent*). Este método recorre ao gradiente de L' para determinar a direcção à qual corresponde um máximo incremento da função objectivo dual. Segue-se uma pesquisa unidimensional para determinar a grandeza do passo a dar nessa direcção. Este procedimento é repetido diversas vezes até se obter uma solução à qual corresponde um valor suficientemente baixo da norma do gradiente de L' . As derivadas de L' em ordem aos multiplicadores de Lagrange podem ser facilmente calculadas por intermédio da substituição de (3.32) em (3.28b) e (3.28c) [Lue84]. Se no programa matemático estiverem presentes restrições desigualdade, a maximização da função objectivo dual $L'(\lambda)$ deve ser efectuada com a restrição de os multiplicadores de Lagrange associados a essas restrições serem positivos.

Apresentam-se em seguida dois métodos duais, que têm sido muito utilizados na resolução de problemas de optimização de estruturas. O primeiro é designado método de linearização convexa (*convex linearization method* - CONLIN) e foi desenvolvido tendo em vista a resolução de problemas de optimização de secções transversais e de espessuras [Fle86]. Posteriormente foi também aplicado à optimização da forma de estruturas com comportamento linear [Fle87]. O método CONLIN baseia-se numa substituição selectiva das variáveis x_i pelos respectivos recíprocos $1/x_i$. O facto de esta substituição ser ou não efectuada depende do sinal da derivada da função objectivo e de cada restrição em ordem a x_i . Depois de efectuada a substituição de variáveis é efectuada uma linearização do problema que nestas circunstâncias é convexa e separável. Este facto permite o cálculo dos multiplicadores de Lagrange recorrendo à maximização da função objectivo dual. Em seguida são calculados os valores das variáveis primais. Este procedimento deve ser repetido várias vezes para que a aproximação linear passe a reproduzir o problema original com maior rigor. Se a solução inicial se encontrar muito distanciada da solução final a aproximação linear pode dar origem a um problema sem qualquer solução. De um modo geral existe também o risco de este problema ocorrer durante o processo iterativo. Nestes casos a região admissível deve ser expandida por intermédio de uma relaxação das restrições.

O método das assimptotas móveis (*method of moving asymptotes* - MMA) [Sva87] pode ser considerado como uma generalização do método CONLIN. No método MMA é efectuada uma substituição das variáveis x_i por $1/(x_i - L_i)$ ou $1/(U_i - x_i)$. Se o parâmetro L_i for considerado nulo e o parâmetro U_i tender para infinito o método MMA torna-se praticamente coincidente com o método CONLIN. A grande vantagem do método MMA reside na possibilidade de controlar o processo iterativo por intermédio de uma modificação dos valores de L_i e U_i . Svanberg sugere que sempre que o processo iterativo apresente oscilações os valores L_i e U_i devem ser aproximados de x_i . Se a convergência se tornar lenta,

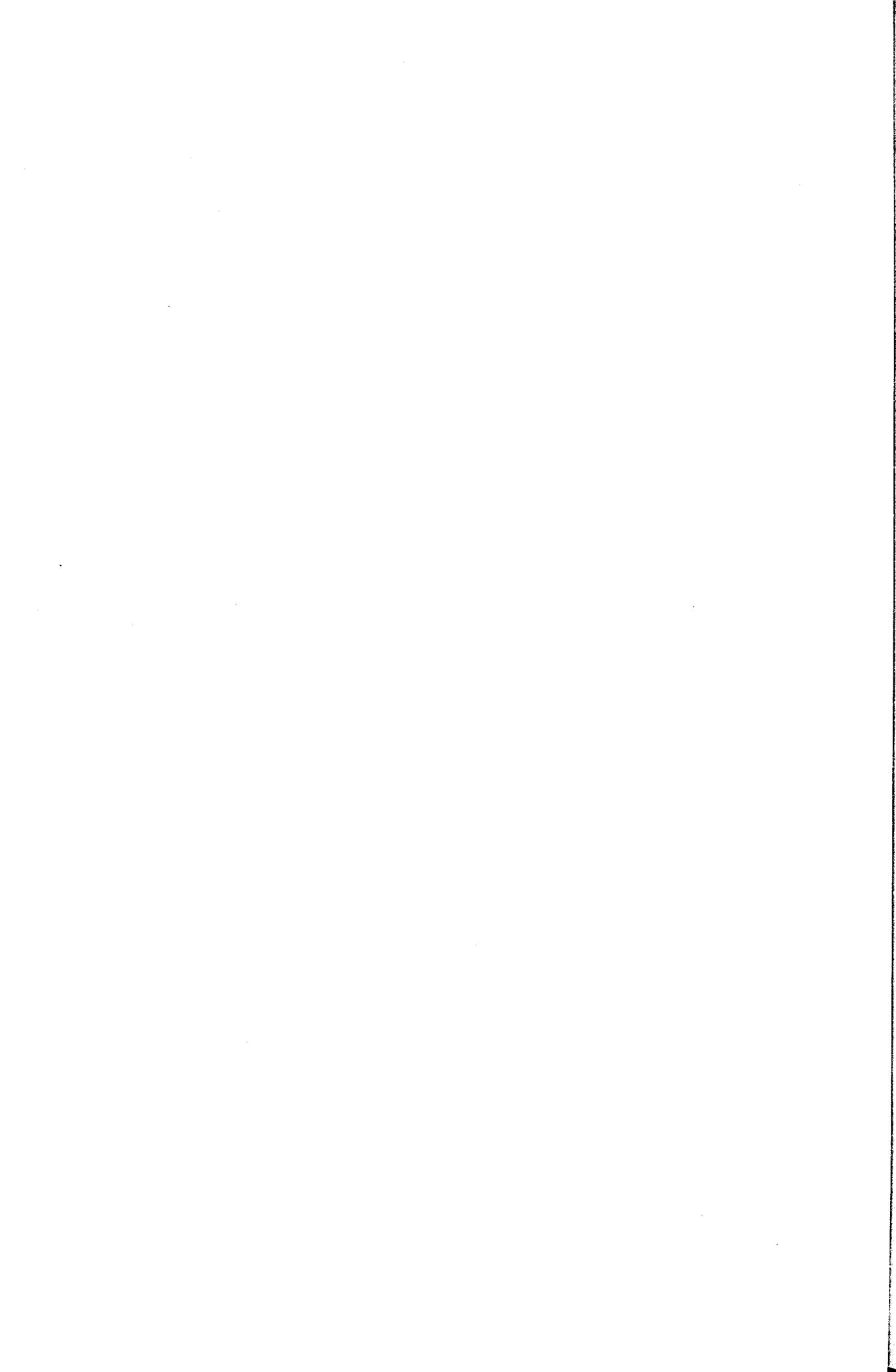
L_i e U_i devem assumir valores mais distanciados de x_i . O nome deste método é devido ao facto de L_i e U_i representarem a posição de assíptotas cuja posição pode ser alterada durante o processo iterativo. Uma eventual dificuldade na definição e modificação dos valores de L_i e U_i ao abordar novos tipos de problemas pode ser considerada como uma desvantagem do método MMA.

Em [Sva87] são apresentados alguns exemplos simples que procuram mostrar que o método MMA é mais robusto e eficiente que o CONLIN. Com ambos estes métodos foi possível resolver problemas de optimização de estruturas em regime linear relativamente complexos [Fle86] [Esp87]. Em [Boc92] o método MMA é generalizado a problemas que além de restrições desigualdade possuem também restrições igualdade. A importância da disponibilidade de informação relativa à curvatura das funções é realçada em [Fle89]. O cálculo de segundas derivadas revela-se essencial para alcançar esse objectivo.

Os resultados apresentados em [Fle86] e [Esp87] mostram que, quer com o método CONLIN, quer com o MMA, é possível resolver problemas de optimização de estruturas com um elevado número de graus de liberdade. No entanto, em ambos os casos o número de variáveis de projecto dos problemas abordados não excedeu poucas dezenas. O comportamento das estruturas foi considerado linear e estas estavam sujeitas a diversos casos de carga e a diversos tipos de restrições. O cálculo das primeiras derivadas das restrições em ordem às variáveis de projecto foi efectuado recorrendo a uma técnica designada análise de sensibilidades (ver Capítulo 6). Nestas circunstâncias os métodos duais revelaram excelentes qualidades. As principais desvantagens destes métodos consistem no facto de apenas serem consideradas aproximações lineares e de ser necessário controlar a progressão do método por intermédio da selecção dos valores de um elevado número de parâmetros (e.g., coeficientes de relaxação das restrições, limites de variação da solução e posição das assíptotas móveis). Incluir no algoritmo a capacidade de decidir quais as alterações que devem ser efectuadas nos valores dos parâmetros é em geral uma tarefa árdua e fálvel. A intervenção do utilizador acaba quase sempre por ser necessária.

3.4 - CRITÉRIOS DE SELECÇÃO DE MÉTODOS DE OPTIMIZAÇÃO

Neste capítulo foram apresentados diversos métodos de optimização. Foram também referidas as principais vantagens e limitações de cada um deles. A selecção dos métodos apresentados baseou-se essencialmente na intenção de exemplificar cada uma das classes de métodos passíveis de serem utilizados na optimização de estruturas. Foram mais



CAPÍTULO 4

MÉTODO DE NEWTON

Os problemas de otimização abordados no âmbito do presente trabalho foram resolvidos pelo método de Lagrange-Newton (ver Capítulo 3). Este método baseia-se na pesquisa de uma solução que verifique determinadas condições necessárias de optimalidade. No caso da minimização de uma função com restrições é procurada uma solução que seja um ponto estacionário do Lagrangeano. A verificação desta condição corresponde a um sistema de equações não lineares que traduz o anulamento do gradiente do Lagrangeano. O método de Lagrange-Newton baseia-se no cálculo da solução deste sistema de equações pelo método de Newton [Fle88].

Neste capítulo é apresentado um algoritmo destinado à resolução de problemas de otimização pelo método de Lagrange-Newton. A versão base deste algoritmo revela-se pouco eficiente e pouco robusta. Com essa versão base apenas se conseguem resolver problemas de otimização com um número de variáveis e restrições relativamente baixo. Tendo em vista a eliminação destas desvantagens, foram acrescentadas ao algoritmo diversas técnicas auxiliares. O algoritmo foi programado tendo em consideração a necessidade de evitar um exagerado consumo de memória. Foi também considerada prioritária a necessidade de programar o algoritmo de tal forma que viesse a ser possível a resolução de problemas reais num intervalo de tempo razoável.

4.1 - ALGORITMO DE OPTIMIZAÇÃO

A minimização pelo método de Lagrange-Newton de uma função sujeita a restrições igualdade e desigualdade apresenta algumas características particulares que devem ser

devidamente exploradas. A formulação genérica do problema de optimização é neste caso a seguinte

$$\text{Min. } f(\underline{x}) \quad \underline{x} = (x_1, \dots, x_n) \quad (4.1a)$$

s.a.

$$\underline{g}(\underline{x}) \leq 0 \quad \underline{g} = (g_1, \dots, g_m) \quad (4.1b)$$

$$\underline{h}(\underline{x}) \leq 0 \quad \underline{h} = (h_1, \dots, h_l) \quad (4.1c)$$

Considera-se que as variáveis \underline{x} são reais e que as funções f , \underline{g} e \underline{h} são reais, contínuas e apresentam primeiras e segundas derivadas também contínuas. Recorrendo a m variáveis de desvio é possível transformar (4.1) num programa matemático em que apenas estão presentes restrições igualdade.

$$\text{Min. } f(\underline{x}) \quad (4.2a)$$

s.a.

$$g_i(\underline{x}) + s_i^2 = 0 \quad (i = 1, \dots, m) \quad (4.2b)$$

$$h_i(\underline{x}) = 0 \quad (i = 1, \dots, l) \quad (4.2c)$$

O Lagrangeano do programa matemático (4.2) é o seguinte (ver Capítulo 2)

$$L(\underline{s}, \underline{\lambda}^g, \underline{x}, \underline{\lambda}^h) = f(\underline{x}) + \sum_{k=1}^m \left[\lambda_k^g (g_k(\underline{x}) + s_k^2) \right] + \sum_{k=1}^l \left[\lambda_k^h h_k(\underline{x}) \right] \quad (4.3)$$

O Lagrangeano (4.3) depende de quatro tipos de variáveis, que se consideram ordenadas da maneira indicada por motivos que serão adiante esclarecidos. A resolução do problema de optimização (4.1) baseia-se no cálculo de uma solução $(\underline{s}^*, \underline{\lambda}^{g*}, \underline{x}^*, \underline{\lambda}^{h*})$ que satisfaça a seguinte condição necessária de optimalidade

$$\nabla L = 0 \quad (4.4)$$

A condição de optimalidade (4.4) corresponde a um sistema de equações não lineares com $2m + n + l$ equações e igual número de incógnitas. Efectuando a derivação do Lagrangeano (4.3) em ordem às variáveis s_i , λ_i^g , x_i e λ_i^h obtém-se

$$2s_i\lambda_i^g = 0 \quad (i = 1, \dots, m) \quad (4.5a)$$

$$g_i + s_i^2 = 0 \quad (i = 1, \dots, m) \quad (4.5b)$$

$$\frac{\partial f}{\partial x_i} + \sum_{k=1}^m \left(\lambda_k^g \frac{\partial g_k}{\partial x_i} \right) + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial h_k}{\partial x_i} \right) = 0 \quad (i = 1, \dots, n) \quad (4.5c)$$

$$h_i = 0 \quad (i = 1, \dots, l) \quad (4.5d)$$

Este sistema de equações não lineares pode ser resolvido pelo método de Newton (ver Capítulo 3), sendo adoptado o algoritmo genérico da Figura 3.1. Nas considerações que se seguem será designado por \tilde{X} o vector que engloba os quatro tipos de variáveis presentes em (4.3)

$$\tilde{X} = \left(\tilde{s}, \tilde{\lambda}^g, \tilde{x}, \tilde{\lambda}^h \right) \quad (4.6)$$

Na fase E) do algoritmo da Figura 3.1 é calculada a variação da solução $\Delta \tilde{X}^q$. De acordo com (3.19) (ver secção 3.2.3), o cálculo de $\Delta \tilde{X}^q$ requer a resolução do seguinte sistema de equações lineares

$$\tilde{H} \left(\tilde{X}^{q-1} \right) \Delta \tilde{X}^q + \nabla L \left(\tilde{X}^{q-1} \right) = 0 \quad (4.7)$$

No sistema de equações lineares (4.7) a matriz dos coeficientes é a matriz Hessiana do Lagrangeano e o vector dos termos independentes é o gradiente do Lagrangeano. Esta matriz e este vector são calculados no ponto correspondente à solução corrente \tilde{X}^{q-1} .

A matriz Hessiana do Lagrangeano é obtida por derivação de cada uma das equações (4.5) em ordem a todas as variáveis presentes em \tilde{X} . Uma vez que em (4.5) figuram quatro tipos de equações e em \tilde{X} figuram quatro tipos de variáveis, a matriz \tilde{H} é constituída por dezasseis submatrizes. Efectuando a derivação das equações (4.5) em ordem às variáveis s_j , λ_j^g , x_j e λ_j^h obtém-se a seguinte matriz simétrica

	(m)	(m)	(n)	(l)	
(m)	\tilde{H}^λ	\tilde{H}^s	0	0	
(m)		0	\tilde{H}^g	0	(4.8a)
(n)	SIMÉTRICA		\tilde{H}^f	\tilde{H}^h	
(l)				0	

$$H_{ij}^\lambda = \begin{cases} 2\lambda_i^g, & \text{se } i = j \\ 0, & \text{se } i \neq j \end{cases} \quad (i = 1, \dots, m; j = 1, \dots, m) \quad (4.8b)$$

$$H_{ij}^s = \begin{cases} 2s_i, & \text{se } i = j \\ 0, & \text{se } i \neq j \end{cases} \quad (i = 1, \dots, m; j = 1, \dots, m) \quad (4.8c)$$

$$H_{ij}^g = \frac{\partial g_i}{\partial x_j} \quad (i = 1, \dots, m; j = 1, \dots, n) \quad (4.8d)$$

$$H_{ij}^f = \frac{\partial^2 f}{\partial x_i \partial x_j} + \sum_{k=1}^m \left(\lambda_k^g \frac{\partial^2 g_k}{\partial x_i \partial x_j} \right) + \sum_{k=1}^l \left(\lambda_k^h \frac{\partial^2 h_k}{\partial x_i \partial x_j} \right) \quad (i = 1, \dots, n; j = 1, \dots, n) \quad (4.8e)$$

$$H_{ij}^h = \frac{\partial h_j}{\partial x_i} \quad (i = 1, \dots, n; j = 1, \dots, l) \quad (4.8f)$$

As submatrizes \tilde{H}^λ e \tilde{H}^s apenas possuem termos não nulos na diagonal. Por este motivo é suficiente dispor de dois vectores com m componentes cada para efectuar o armazenamento destas duas submatrizes. Por motivos que serão adiante apresentados, da matriz \tilde{H}^g apenas é necessário armazenar os termos não nulos. A matriz de $n + l$ linhas e $n + l$ colunas que inclui \tilde{H}^f e \tilde{H}^h é armazenada como uma matriz triangular superior cheia. Efectuando o armazenamento das cinco submatrizes (4.8b-f) da forma indicada, consegue-se uma grande economia de memória em relação ao armazenamento completo da matriz Hessiana.

Os sistema de equações lineares (4.7) possui como vector de termos independentes o gradiente do Lagrangeano no ponto correspondente à solução corrente. O cálculo deste vector em cada iteração é efectuada recorrendo aos primeiros membros das equações (4.5). Da

resolução do sistema de equações lineares (4.7) resulta o vector $\Delta \tilde{X}^q$. Este vector corresponde à variação do valor de todas as variáveis que estão presentes em (4.6). De acordo com o algoritmo da Figura 3.1, é em seguida efectuado o cálculo do parâmetro de pesquisa unidimensional α^q . O valor de α^q deve ser calculado com o objectivo de minimizar a norma do gradiente do Lagrangeano na direcção $\Delta \tilde{X}^q$. Em seguida é actualizado o valor de todas as variáveis \tilde{X} recorrendo à seguinte expressão

$$\tilde{X}^q = \tilde{X}^{q-1} + \alpha^q \Delta \tilde{X}^q \quad (4.9)$$

O critério de convergência correspondente à fase C) do algoritmo da Figura 3.1 baseia-se na verificação da seguinte condição

$$\left\| \nabla L(\tilde{X}^q) \right\| < \varepsilon \quad (4.10)$$

Uma vez que o método de Newton possui convergência quadrática, o parâmetro ε pode adoptar um valor muito pequeno. Na generalidade dos problemas de optimização resolvidos no âmbito do presente trabalho foi atribuído a ε o valor 10^{-5} . Com este critério de convergência e com este valor de ε obtêm-se em geral soluções óptimas com mais de quatro algarismos significativos correctos.

Na solução óptima \tilde{X}^* estão presentes os valores das variáveis de projecto $\begin{pmatrix} x^* \\ \tilde{} \end{pmatrix}$, das variáveis de desvio $\begin{pmatrix} s^* \\ \tilde{} \end{pmatrix}$ e dos multiplicadores de Lagrange $\begin{pmatrix} \lambda^g \\ \lambda^h \\ \tilde{} \end{pmatrix}$. Para facilitar a análise da solução óptima, os valores das variáveis de desvio devem ser apresentados elevados ao quadrado. De acordo com (4.2b), o valor de s_i^2 representa a folga associada a cada restrição desigualdade. As restrições que apresentarem um valor de s_i^2 próximo de zero pertencem ao conjunto das restrições activas (*active set*). Se algum componente do vector λ^g apresentar um valor negativo, o processo iterativo deve ser repetido com outra solução inicial. Na prática só se deve considerar negativo um valor inferior a $-\varepsilon$. A presença de valores negativos nas componentes de λ^g indica que existem soluções admissíveis com um valor inferior da função objectivo (ver Capítulo 2).

4.2 - TÉCNICAS DE ARMAZENAMENTO E DERIVAÇÃO DE FUNÇÕES

No programa matemático (4.1) a função objectivo e cada uma das restrições são funções que dependem do problema que se pretende resolver. Uma vez que os algoritmos de optimização necessitam de avaliar e eventualmente derivar estas funções, é necessário introduzi-las de alguma forma no código computacional. O processo mais simples é o da descrição das funções com a sintaxe da linguagem de programação utilizada. Em seguida é necessário compilar e ligar o código que contém a descrição das funções ao programa de optimização. Se o algoritmo necessitar das derivadas das funções, estas podem ser calculadas numericamente. O cálculo de derivadas por este processo pode apresentar graves problemas numéricos se o valor da perturbação das funções não for convenientemente escolhido. Em alternativa, é possível derivar previamente as funções e adicionar as expressões das derivadas à descrição efectuada com a sintaxe da linguagem de programação. Este procedimento é pouco prático e apresenta o grave inconveniente de o volume de informação que tem de ser fornecido poder facilmente exceder a capacidade do compilador. Quando se recorre a esta técnica, a utilização do programa de optimização só é possível se estiver disponível um compilador compatível com o que foi utilizado na compilação dos restantes módulos. As subrotinas de optimização da biblioteca NAG [NAG91] apresentam estas desvantagens, ficando assim restringidas à resolução de problemas com um número moderado de variáveis e restrições. Se um programa de optimização apenas se destinar à resolução de uma determinada classe de problemas com características bem definidas, a inserção das diversas funções no código computacional pode não constituir um inconveniente. Neste caso a única desvantagem consiste na impossibilidade de acrescentar posteriormente restrições que não tenham sido previstas pelo programador.

A introdução da função objectivo e restrições no programa de optimização pode ser efectuada por um processo alternativo designado programação simbólica. Neste caso é necessário que seja o programa de computador a interpretar, derivar e avaliar as diversas funções que constituem o programa matemático. Estas funções são incluídas no ficheiro de dados, de acordo com uma sintaxe compatível com o programa que vai proceder à sua interpretação. Deste modo, com um único programa executável é possível abordar problemas de optimização com distintas formulações [Van89b]. Esta alternativa apresenta grande versatilidade e facilidade de utilização. A principal desvantagem é a elevada complexidade associada ao desenvolvimento do código computacional.

No âmbito do presente trabalho foi desenvolvida uma versão limitada da alternativa designada programação simbólica. A função objectivo f e as restrições g_j e h_k apresentam neste caso a forma genérica correspondente à seguinte função $p(\underline{x})$

$$p(\underline{x}) = \sum_{t=1}^T \left(c_t \prod_{i=1}^n x_i^{a_{ti}} \right) \quad (4.11)$$

T - número de termos;

n - número de variáveis de projecto;

c_t - coeficiente do termo t ;

a_{ti} - expoente da variável x_i no termo t .

Em (4.11) os coeficientes c_t e as variáveis x_i são números reais sem restrição de sinal. Na versão actual do código computacional os expoentes a_{ti} são números inteiros sem restrição de sinal. A passagem dos expoentes a_{ti} a números reais não apresenta qualquer dificuldade, implicando apenas um aumento do consumo de recursos informáticos. No presente trabalho apenas são abordados problemas com expoentes a_{ti} inteiros e sem restrição de sinal.

Durante a execução do programa de optimização os valores dos coeficientes c_t e dos expoentes a_{ti} correspondentes às diversas funções que constituem o programa matemático são lidos no ficheiro de dados. Quer neste ficheiro, quer no tipo de armazenamento efectuado pelo código computacional, apenas são considerados os termos que apresentam um valor de c_t não nulo e em cada termo apenas figuram as variáveis com expoente a_{ti} não nulo. Apresenta-se em seguida como exemplo uma função que se enquadra na expressão genérica (4.11)

$$p(\underline{x}) = 5.9x_1^2x_4^{-3} - 3.1x_2 + 2.7x_1^{-1}x_3x_5^2 - 1.8 \quad (4.12)$$

A derivação da função (4.11) em ordem a uma variável x_j pode ser facilmente efectuada, resultando a seguinte expressão genérica

$$\frac{\partial p}{\partial x_j} = \sum_{t=1}^T \left(c_t a_{tj} x_j^{a_{tj}-1} \prod_{\substack{i=1 \\ i \neq j}}^n x_i^{a_{ti}} \right) \quad (4.13)$$

Devido às características das expressões (4.11) e (4.13), é essencial que durante a execução do programa as variáveis com expoente negativo nunca se anulem. Uma vez que da derivação de (4.11) resulta uma função do mesmo tipo, é igualmente simples e eficiente calcular as segundas derivadas de $p(\underline{x})$.

Com o objectivo de calcular o gradiente do Lagrangeano (4.5) e a Hessiana do Lagrangeano (4.8), é necessário efectuar a avaliação das funções $p(\underline{x})$ e calcular as respectivas derivadas de ordem um e dois no ponto correspondente à solução corrente. Para evitar a necessidade de armazenar um elevado volume de informação temporária associada às sucessivas derivações, foi adoptado o seguinte algoritmo.

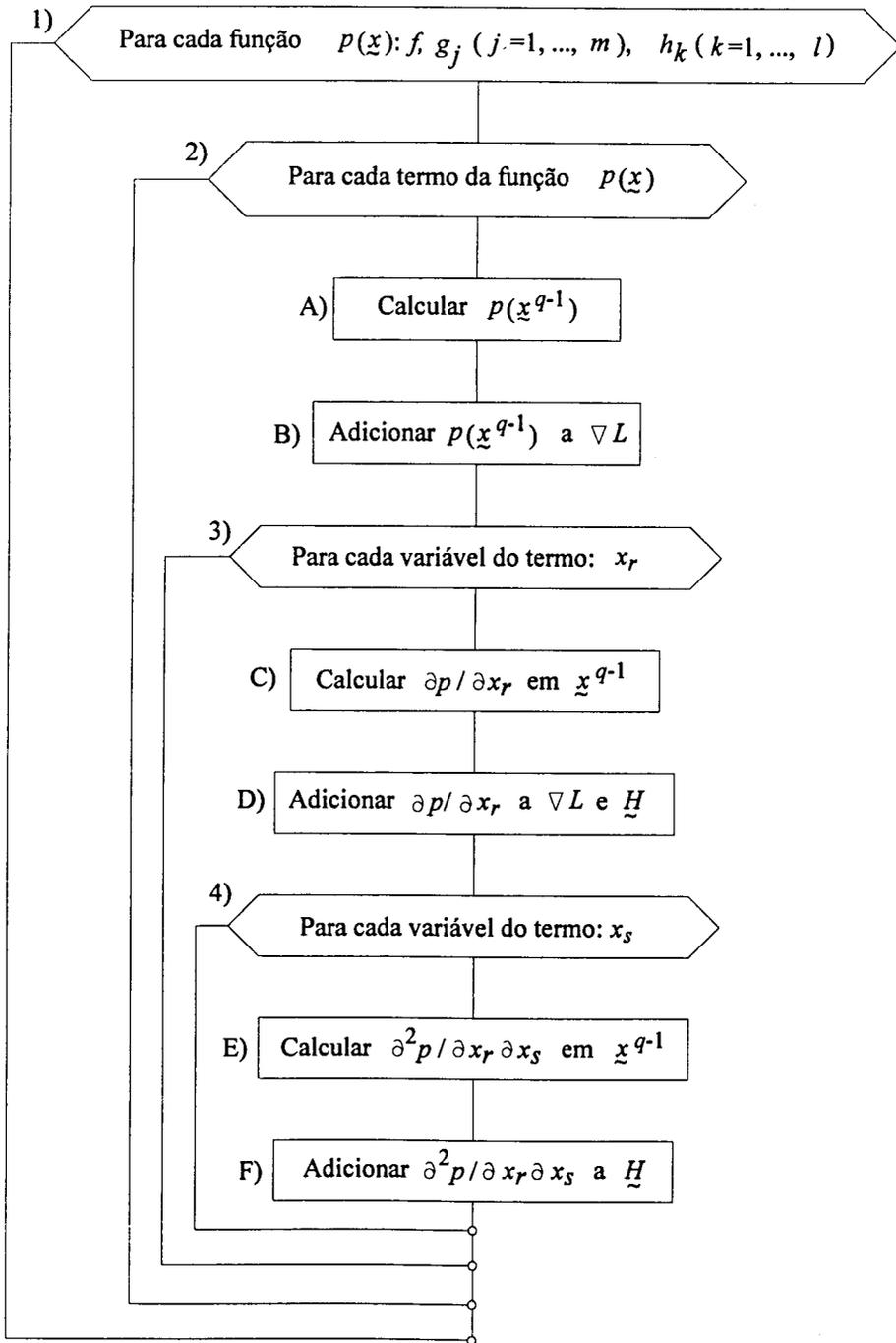


Figura 4.1 - Algoritmo utilizado na avaliação e derivação de funções.

Notas relativas ao algoritmo da Figura 4.1:

- o ciclo 1) percorre as $1 + m + l$ funções do programa matemático (4.1);
- o ciclo 2) percorre um número de termos T que depende da função;
- os ciclos 3) e 4) apenas percorrem as variáveis que possuem um expoente não nulo;
- uma vez que a matriz Hessiana é simétrica, no ciclo 4) é suficiente percorrer as variáveis que apresentam $s \geq r$;
- nas fases B), D) e F) a adição dos valores calculados a ∇L e H é efectuada de acordo com as expressões (4.5) e (4.8). É necessário atender às seguintes excepções: na fase B) o valor da função objectivo não é adicionado a ∇L e na fase D) $\partial f / \partial x_r$ não é adicionado a H .

A utilização do algoritmo da Figura 4.1 e o facto de em cada termo figurar habitualmente um reduzido número de variáveis permite que o armazenamento e a avaliação das diversas derivadas seja efectuado com um reduzido consumo de memória. Verifica-se também que o tempo gasto nestas operações, no cálculo do gradiente do Lagrangeano e no cálculo da matriz Hessiana é desprezável quando comparado com o correspondente à resolução do sistema de equações lineares (4.7). Com a utilização do algoritmo da Figura 4.1 é eliminada uma das principais desvantagens que é atribuída ao método de Newton e que consiste na dificuldade de calcular a matriz Hessiana [Aro89].

4.3 - TÉCNICAS DE SCALING

As inúmeras técnicas destinadas a melhorar o condicionamento de um programa matemático são genericamente designadas *scaling*. A transformação de um programa matemático mal condicionado num outro bem condicionado conduz em geral a um aumento da probabilidade de se obter uma solução precisa e a uma diminuição do tempo de resolução. A solução do problema transformado tem de coincidir com a do problema original ou, em alternativa, tem de existir um processo de relacionar as duas soluções. As técnicas de *scaling* podem ser muito ou pouco elaboradas e podem afectar diferentes grandezas associadas à resolução do problema de optimização. Por este motivo, o tipo de *scaling* que deve ser efectuado depende em parte do método de optimização que se pretende vir a utilizar. Deve em

geral ser evitada a utilização de técnicas de *scaling* demasiado complexas, porque o esforço despendido na sua aplicação pode não ser compensador.

Existem fundamentalmente duas técnicas de *scaling*, que devem ser sempre associadas à resolução de um problema de minimização com restrições. A primeira baseia-se numa transformação de variáveis e a segunda tem como objectivo a normalização das restrições. De acordo com o respectivo significado físico, as diversas variáveis de um problema de optimização podem assumir valores com ordens de grandeza muito distintas. É também possível que uma determinada variável só possa assumir valores num intervalo de pequena amplitude (e.g. $1000.1 \leq x_i \leq 1000.2$). Quer num caso quer noutra, para evitar eventuais problemas numéricos seria vantajoso que todas as variáveis adoptassem valores por exemplo no intervalo $[-1,1]$. Este objectivo pode ser alcançado recorrendo à seguinte transformação de variáveis [Gil81]

$$x_i = Z_i \bar{x}_i + W_i \quad (i = 1, \dots, n) \quad (4.14)$$

Os coeficientes Z_i e W_i têm de ser determinados de tal forma que a transformação inversa de (4.14) converta o intervalo correspondente à variável x_i num intervalo com extremos próximos da unidade e com amplitude aproximadamente unitária. Depois de obtida a solução \bar{x}^* , é possível calcular a solução do problema original recorrendo a (4.14). No caso da minimização sem restrições, um problema é bem condicionado se a respectiva matriz Hessiana pouco diferir da matriz identidade. De modo a transformar o problema inicial num outro que apresente esta característica, é proposta em [Van84] uma transformação de variáveis do tipo (4.14) mas com W_i nulo.

A normalização das restrições deve ser sempre efectuada para que todas elas influenciem de um modo semelhante a convergência para a solução óptima. Em [Van84] é aconselhada a seguinte técnica de normalização das restrições

$$g_j(\tilde{x}) + c_j \leq 0 \quad \rightarrow \quad \frac{g_j(\tilde{x}) + c_j}{|c_j|} \leq 0 \quad (4.15)$$

Esta transformação deve ser aplicada a todas as restrições igualdade e desigualdade que apresentem um termo constante c_j não nulo. A normalização pode também ser efectuada por intermédio da multiplicação de todas as restrições por uma constante calculada de tal forma que as normas dos gradientes das restrições igualem a norma do gradiente da função objectivo [Van84]. Estes gradientes podem ser calculados no ponto correspondente à solução inicial.

No presente trabalho foram utilizadas técnicas de *scaling* que, além de melhorar o condicionamento do programa matemático, se destinam a permitir a definição prévia de diversas constantes que influenciam o comportamento do processo iterativo. Se não forem aplicadas técnicas de *scaling*, o valor da tolerância ϵ associada à verificação do critério de convergência (4.10) não é independente das unidades utilizadas na definição do programa matemático. Este facto dificulta em cada caso a selecção do valor de ϵ . Durante o processo iterativo e na análise da solução óptima, é necessário verificar se determinadas grandezas são nulas. Devido à acumulação de erros de arredondamento, considera-se que uma grandeza é nula quando apresenta um valor absoluto inferior a uma tolerância. Se não forem utilizadas técnicas de *scaling*, a selecção em cada caso de um valor para esta tolerância pode ser problemática.

Transformação das variáveis

No algoritmo desenvolvido no âmbito do presente trabalho foi incluída a possibilidade de efectuar uma transformação das variáveis de projecto. Esta transformação é efectuada de acordo com (4.14), sendo W_i considerado nulo

$$x_i = Z_i \bar{x}_i \quad (i = 1, \dots, n) \quad (4.16)$$

Para que a um valor de \bar{x}_i unitário esteja associada a manutenção do valor correspondente à solução inicial x_i^0 , atribuem-se às componentes de Z os seguintes valores

$$Z_i = x_i^0 \quad (i = 1, \dots, n) \quad (4.17)$$

Substituindo (4.16) em cada termo das diversas funções (4.11) que constituem o programa matemático (4.1), resulta

$$c_t \prod_{i=1}^n x_i^{a_{it}} \rightarrow \bar{c}_t \prod_{i=1}^n \bar{x}_i^{a_{it}} \quad (4.18)$$

Os coeficientes \bar{c}_t podem ser facilmente calculados com base na solução inicial $Z = x^0$

$$\bar{c}_t = c_t \prod_{i=1}^n Z_i^{a_{it}} \quad (4.19)$$

Depois de obtida a solução óptima \bar{x}^* é possível convertê-la na solução do problema original recorrendo a (4.16).

Para que a transformação de variáveis não conduza a uma solução errada, é necessário que todas as componentes de \tilde{x}^0 sejam não nulas. Para evitar problemas numéricos é aconselhável que todas as componentes de \tilde{x}^0 apresentem um valor absoluto superior a uma tolerância predefinida.

A solução inicial do problema transformado é a seguinte

$$\bar{x}_i^0 = 1 \quad (i = 1, \dots, n) \quad (4.20)$$

Após a conversão das restrições desigualdade (4.1b) em restrições igualdade é também efectuada uma transformação linear das variáveis de desvio s_j .

$$g_j(\tilde{x}) + s_j^2 = 0 \quad (j = 1, \dots, m) \quad (4.21)$$

$$s_j = Z_j \bar{s}_j \quad (j = 1, \dots, m) \quad (4.22)$$

$$g_j(\tilde{x}) + Z_j^2 \bar{s}_j^2 = 0 \quad (j = 1, \dots, m) \quad (4.23)$$

Uma vez que é aconselhável que a restrição (4.21) seja respeitada quando $\tilde{x} = \tilde{x}^0$, o valor inicial de s_j deve ser o seguinte

$$s_j^0 = \sqrt{|g_j(\tilde{x}^0)|} \quad (j = 1, \dots, m) \quad (4.24)$$

Em (4.24) $g_j(\tilde{x}^0)$ figura em módulo para que sejam também contempladas as situações em que a solução inicial não respeita a restrição $g_j(\tilde{x}) \leq 0$. Por analogia com o tipo de transformação que é efectuada no caso das variáveis de projecto, é atribuído a Z_j o seguinte valor

$$Z_j = s_j^0 \quad (j = 1, \dots, m) \quad (4.25)$$

Nestas circunstâncias o valor inicial das variáveis de desvio no problema transformado é o seguinte

$$\bar{s}_j^0 = 1 \quad (j = 1, \dots, m) \quad (4.26)$$

Depois de obtida a solução óptima do problema transformado, devem ser calculadas as folgas correspondentes às restrições desigualdade do problema original. O valor dessas folgas é fornecido pela seguinte expressão

$$Z_j^2 \bar{s}_j^{*2} \quad (j = 1, \dots, m) \quad (4.27)$$

Se algum valor de Z_j obtido com base em (4.24) e (4.25) for nulo, a respectiva restrição (4.23) figura no programa matemático transformado como uma restrição sempre activa. Se este facto ocorrer e se na solução óptima do problema original essa restrição for inactiva, a solução do problema transformado não corresponde à do problema original. Por este motivo nenhum Z_j pode ser nulo, sendo desaconselhável por questões numéricas a adopção de valores muito pequenos. Este problema deve ser evitado por intermédio da selecção de uma solução inicial que implique a existência de uma folga significativa em todas as restrições desigualdade. Se não for possível evitar que o valor de Z_j obtido com base em (4.24) e (4.25) seja demasiado pequeno, deve ser utilizado um valor predefinido.

Normalização das restrições

No algoritmo de optimização desenvolvido no âmbito do presente trabalho foi incluída uma técnica de normalização automática das restrições. O objectivo desta normalização foi o de fazer com que os gradientes da função objectivo e de todas as restrições apresentem norma unitária para $\tilde{x} = \tilde{x}^0$. Nas considerações que se seguem as restrições (4.23) são incluídas no conjunto das restrições igualdade. Na normalização dos gradientes das funções que constituem o programa matemático original são utilizados os seguintes coeficientes e as seguintes transformações

$$Z_0 = \left\| \nabla f \left(\tilde{x}^0 \right) \right\| \quad (4.28)$$

$$Z_k = \left\| \nabla h_k \left(\tilde{x}^0 \right) \right\| \quad (k = 1, \dots, l) \quad (4.29)$$

$$\nabla \bar{f} = \frac{1}{Z_0} \nabla f \quad (4.30)$$

$$\nabla \bar{h}_k = \frac{1}{Z_k} \nabla h_k \quad (k = 1, \dots, l) \quad (4.31)$$

$$\bar{f} = \frac{1}{Z_0} f \quad (4.32)$$

$$\bar{h}_k = \frac{1}{Z_k} h_k \quad (k = 1, \dots, l) \quad (4.33)$$

Uma vez que as funções f e h_k são do tipo (4.11), as transformações (4.32) e (4.33) são efectuadas recorrendo à multiplicação dos coeficientes c_i por $1/Z_0$ ou $1/Z_k$. O anulamento do gradiente do Lagrangeano do programa matemático transformado corresponde ao seguinte sistema de equações não lineares (ver secção 2.2.3)

$$\nabla \bar{f} + \sum_{k=1}^l (\bar{\lambda}_k \nabla \bar{h}_k) = \underline{0} \quad (4.34a)$$

$$\bar{h}_k = 0 \quad (k = 1, \dots, l) \quad (4.34b)$$

Substituindo (4.30), (4.31) e (4.33) neste sistema de equações obtém-se

$$\frac{1}{Z_0} \nabla f + \sum_{k=1}^l \left(\bar{\lambda}_k \frac{1}{Z_k} \nabla h_k \right) = \underline{0} \quad (4.35a)$$

$$\frac{1}{Z_k} h_k = 0 \quad (k = 1, \dots, l) \quad (4.35b)$$

Multiplicando a equação (4.35a) por Z_0 e cada equação (4.35b) pelo respectivo Z_k obtém-se

$$\nabla f + \sum_{k=1}^l \left(\bar{\lambda}_k \frac{Z_0}{Z_k} \nabla h_k \right) = \underline{0} \quad (4.36a)$$

$$h_k = 0 \quad (k = 1, \dots, l) \quad (4.36b)$$

Se a solução óptima for um ponto regular, os vectores $\nabla h_k \left(\underline{x}^* \right)$ são linearmente independentes (ver secção 2.2.6). Nestas circunstâncias, comparando (2.36) com (4.36) conclui-se que os multiplicadores de Lagrange do problema original podem ser obtidos a partir dos correspondentes ao problema transformado utilizando a seguinte expressão

$$\lambda_k = \bar{\lambda}_k \frac{Z_0}{Z_k} \quad (k = 1, \dots, l) \quad (4.37)$$

Da comparação de (2.36) com (4.36) conclui-se também que a normalização da função objectivo e restrições não afecta a solução do programa matemático.

Para evitar divisões por zero ou por valores muito pequenos, os coeficientes Z_0 e Z_k devem ser maiores do que uma tolerância. Nas raras situações em que as expressões (4.28) e (4.29) conduzirem a valores de Z_0 ou Z_k demasiado pequenos deve ser utilizada outra solução inicial \underline{x}^0 .

Atendendo ao modo como a normalização da função objectivo e restrições é efectuada, sugere-se que no problema transformado os valores iniciais dos multiplicadores de Lagrange sejam unitários

$$\bar{\lambda}_k^0 = 1 \quad (k = 1, \dots, l) \quad (4.38)$$

Se forem aplicadas ao programa matemático todas as técnicas de *scaling* aqui sugeridas, a solução inicial do problema transformado é a seguinte

$$\bar{X}^0 = \left(\bar{s}, \bar{\lambda}^g, \bar{x}, \bar{\lambda}^h \right)^0 = (1, \dots, 1) \quad (4.39)$$

Exemplo de aplicação das técnicas de *scaling*

Com o objectivo de exemplificar o efeito da aplicação das diversas técnicas de *scaling* atrás referidas, apresenta-se o seguinte programa matemático

$$\text{Min. } 2000x_1 \quad (4.40a)$$

s.a.

$$-x_1 + 200 + s_1^2 = 0 \quad (4.40b)$$

$$x_2 - 0.2 + s_2^2 = 0 \quad (4.40c)$$

$$-10x_1x_2 + 500 = 0 \quad (4.40d)$$

As restrições igualdade (4.40b) e (4.40c) resultam da adição de um termo de desvio a duas restrições desigualdade que estavam presentes no problema original. Como solução inicial de (4.40) é adoptado o vector $\bar{x}^0 = (500, 0.1)$. Para que (4.40b) e (4.40c) sejam respeitadas, as variáveis de desvio devem adoptar os valores $\bar{s}^0 = (\sqrt{300}, \sqrt{0.1})$. Efectuando uma transformação de variáveis de acordo com (4.16) e (4.22), seguida de uma normalização da função objectivo e das restrições de acordo com (4.32) e (4.33) resulta o seguinte programa matemático

$$\text{Min. } \bar{x}_1 \quad (4.41a)$$

s.a.

$$-0.640 \bar{x}_1 + 0.256 + 0.384 \bar{s}_1^2 = 0 \quad (4.41b)$$

$$0.447 \bar{x}_2 - 0.894 + 0.447 \bar{s}_2^2 = 0 \quad (4.41c)$$

$$-0.707 \bar{x}_1 \bar{x}_2 + 0.707 = 0 \quad (4.41d)$$

No Quadro 4.1 encontra-se a solução inicial e a solução óptima relativa aos programas matemáticos (4.40) e (4.41). Pode-se facilmente verificar que os valores absolutos dos coeficientes dos termos do programa matemático (4.40), bem como os valores das respectivas soluções inicial e óptima se situam no intervalo $[0.1, 2\ 500\ 000]$. Depois de aplicadas as técnicas de *scaling*, este intervalo passa a ser $[0.256, 2]$. A diminuição da amplitude do intervalo e o facto de todos os valores passarem a situar-se próximo da unidade são aspectos benéficos do ponto de vista numérico. A confirmação deste facto pode ser obtida por intermédio da resolução de ambos os programas matemáticos pelo método de Newton. Na versão (4.40), i.e., sem a aplicação de qualquer técnica de *scaling*, o número de iterações necessário à resolução do problema é muito superior ao correspondente à versão (4.41). A resolução do problema (4.40) sem *scaling* requer a alteração de diversos parâmetros e tolerâncias numéricas para se conseguir a convergência para uma solução precisa. Na versão (4.41) obtém-se uma solução precisa com os valores habituais dos parâmetros e tolerâncias. Quando é aplicado à versão transformada do programa matemático, o método de Newton apresenta-se muito mais robusto.

Quadro 4.1 - Solução inicial e solução óptima relativa aos programas matemáticos (4.40) e (4.41).

	Programa matemático (4.40)		Programa matemático (4.41)		
	Solução inicial	Solução óptima	Solução inicial	Solução óptima	
x_1	500	250	1	0.500	\bar{x}_1
x_2	0.1	0.2	1	2.000	\bar{x}_2
s_1	$\sqrt{300}$	$\sqrt{50}$	1	0.408	\bar{s}_1
s_2	$\sqrt{0.1}$	0	1	0.000	\bar{s}_2
λ_1^g	10 000	0	1	0.000	$\bar{\lambda}_1^g$
λ_2^g	10 000	2 500 000	1	0.559	$\bar{\lambda}_2^g$
λ_1^h	10 000	1 000	1	0.707	$\bar{\lambda}_1^h$

4.4 - RESOLUÇÃO DO SISTEMA DE EQUAÇÕES LINEARES

A resolução de um problema de optimização pelo método de Newton requer que em cada iteração seja resolvido o seguinte sistema de equações lineares (ver secção 4.1)

$$H\left(\underset{\sim}{X}^{q-1}\right)\Delta\underset{\sim}{X}^q = -\nabla L\left(\underset{\sim}{X}^{q-1}\right) \quad (4.42)$$

Este sistema de equações pode ser simplifadamente representado do seguinte modo

$$\underset{\sim}{A} \underset{\sim}{x} = \underset{\sim}{b} \quad (4.43)$$

Neste sistema de equações a matriz dos coeficientes é a matriz Hessiana do Lagrangeano calculada no ponto correspondente à solução corrente $\underset{\sim}{X}^{q-1}$. Os elementos desta matriz são calculados com as expressões (4.8). O vector dos termos independentes é o simétrico do gradiente do Lagrangeano também calculado no ponto $\underset{\sim}{X}^{q-1}$. O cálculo dos elementos de ∇L é efectuado com os primeiros membros das equações (4.5).

No algoritmo descrito no presente capítulo, a fase que requer maior quantidade de memória e demora mais tempo é a correspondente à resolução do sistema de equações (4.42). Uma vez que é esta fase que condiciona a *performance* global do algoritmo, deve ser utilizado um método de resolução de sistemas de equações que tire partido da simetria e da esparsidade da matriz Hessiana. O algoritmo desenvolvido no âmbito do presente trabalho apresenta nesta fase a possibilidade de ser utilizado em alternativa um método directo ou um método iterativo. O método directo considerado é o de eliminação de Gauss adaptado às características da matriz Hessiana e o iterativo é o dos gradientes conjugados com pré-condicionamento. Este último apresenta características que permitem a sua fácil adaptação à resolução de sistemas de equações com a matriz dos coeficientes esparsa. A adaptação destes dois métodos à resolução de (4.42) é em seguida apresentada.

4.4.1 - Método directo

A matriz dos coeficientes correspondente ao sistema de equações (4.42) é simétrica e é constituída por dezasseis submatrizes definidas pelas expressões (4.8). Se não forem consideradas as características particulares da matriz Hessiana, torna-se necessário armazenar os seus $(2m + n + l) \times (2m + n + l)$ elementos. A quantidade de memória necessária para

armazenar a matriz nestas circunstâncias alcança valores muito elevados mesmo com um número moderado de variáveis e restrições. Com o objectivo de economizar memória e diminuir o tempo de resolução, devem ser utilizados algoritmos que explorem a estrutura e a esparsidade da matriz Hessiana.

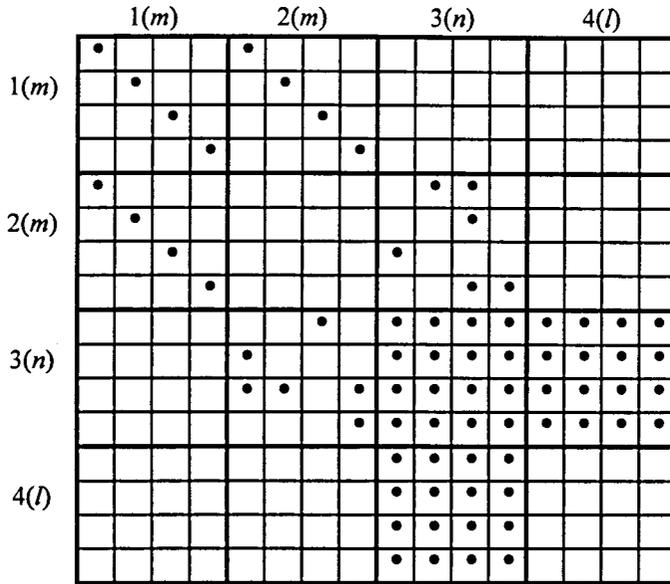


Figura 4.2 - Termos não nulos da matriz dos coeficientes antes de ser iniciada a fase de eliminação.

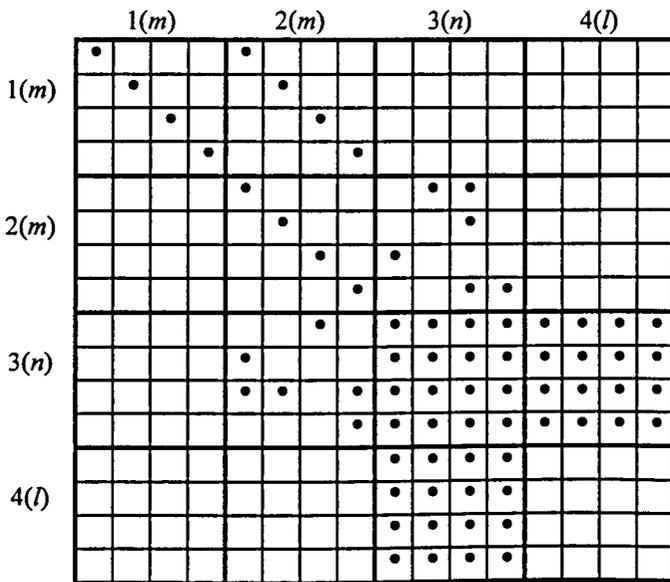


Figura 4.3 - Termos não nulos da matriz dos coeficientes após a eliminação da coluna 1.

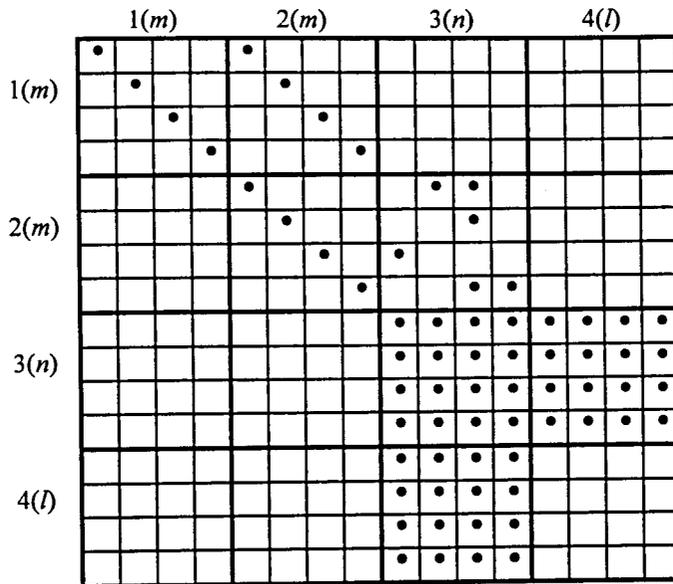


Figura 4.4 - Termos não nulos da matriz dos coeficientes após a eliminação das colunas 1 e 2.

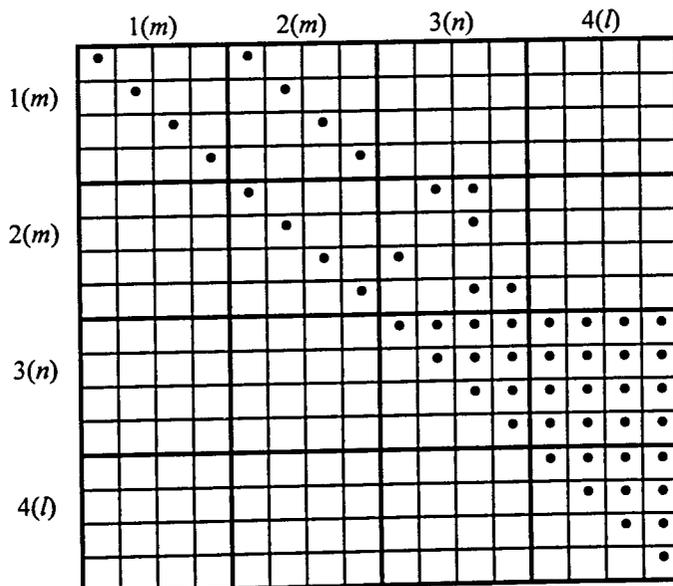


Figura 4.5 - Termos não nulos da matriz dos coeficientes após a eliminação das colunas 1, 2, 3 e 4.

O método de eliminação de Gauss possui uma característica designada *fill-in*, que consiste no facto de alguns elementos da matriz dos coeficientes que inicialmente eram nulos deixarem de o ser, devido às operações que são efectuadas na fase de eliminação. Este facto faz com que seja necessário prever espaço de armazenamento para os elementos que

inicialmente são diferentes de zero e para os elementos que durante a fase de eliminação deixam de ser nulos. Quando a matriz dos coeficientes não apresenta um padrão de esparsidade adequado, revela-se fundamental efectuar trocas de linhas e colunas com o objectivo de conseguir uma disposição dos seus elementos que não conduza a demasiado *fill-in*. No caso da matriz Hessiana, foi possível obter uma conveniente disposição dos termos não nulos por intermédio da selecção da ordem pela qual os diversos tipos de variáveis figuram no Lagrangeano (4.3). Nestas circunstâncias e de acordo com (4.8), obtém-se uma matriz Hessiana, cujo padrão de esparsidade se encontra exemplificado na Figura 4.2. Nas Figuras 4.3 a 4.5 encontra-se a evolução que este padrão de esparsidade apresenta nas sucessivas fases correspondentes à eliminação de Gauss da matriz Hessiana. Em relação a este conjunto de figuras são em seguida apresentadas algumas observações.

A matriz Hessiana inicial é simétrica, bem como a submatriz que durante a eliminação de Gauss ainda possui termos não nulos abaixo da diagonal. Por este motivo nunca é necessário armazenar elementos que não pertençam ao triângulo superior. Atendendo ao padrão de esparsidade final apresentado na Figura 4.5, a matriz Hessiana é armazenada do seguinte modo:

- as submatrizes H_{11} ($m \times m$), H_{12} ($m \times m$) e H_{22} ($m \times m$) são armazenadas em três vectores, possuindo cada um deles m componentes;
- da submatriz H_{23} ($m \times n$) são armazenados apenas os termos que na matriz Hessiana inicial são não nulos. Esta submatriz apresenta na quase totalidade dos casos uma elevada esparsidade, que se mantém durante a eliminação de Gauss devido ao facto de H_{23} não apresentar *fill-in*. O padrão de esparsidade de H_{23} nas Figuras 4.2 a 4.5 constitui apenas um exemplo;
- da submatriz $(n + l) \times (n + l)$ que inclui H_{33} , H_{34} , H_{43} e H_{44} apenas é armazenado o triângulo superior, cujo número de elementos é $(n + l) \times (n + l + 1)/2$. Na generalidade dos casos, a quantidade de memória necessária para armazenar esta matriz triangular é de ordem de grandeza superior à correspondente às restantes submatrizes da Hessiana.

O tempo de CPU correspondente a cada uma das fases indicadas nas Figuras 4.2 a 4.5 depende essencialmente das características das submatrizes e do tipo de armazenamento associado a cada uma delas. O número de termos não nulos, o padrão de esparsidade e o *fill-in* têm uma influência nos tempos de resolução que é muito difícil de quantificar. Por esse motivo, nas considerações que se seguem é apenas efectuada uma breve referência às diferentes ordens de grandeza dos tempos de CPU relativos a cada uma das fases:

- a eliminação dos elementos da submatriz H_{21} (ver Figura 4.3) é efectuada recorrendo a operações simples sobre os vectores em que se encontram armazenadas as submatrizes H_{11} , H_{12} e H_{22} . Esta fase apenas apresenta *fill-in* nos elementos da diagonal de H_{22} . O tempo de CPU associado à eliminação da coluna 1 não é significativo quando comparado com o correspondente à eliminação das colunas 3 e 4 (ver Figura 4.5);
- a eliminação dos elementos da submatriz H_{32} (ver Figura 4.4) apenas apresenta *fill-in* na submatriz H_{33} . Por esse motivo, as operações efectuadas apenas envolvem os termos não nulos de H_{22} e H_{23} . Como nesta fase H_{22} é uma matriz diagonal e na generalidade dos casos H_{23} é muito esparsa, o tempo de CPU associado à eliminação da coluna 2 é também pouco significativo;
- a eliminação dos elementos que se situam abaixo da diagonal nas colunas 3 e 4 (ver Figura 4.5) é efectuada sem ter em consideração a eventual presença de elementos nulos na matriz triangular superior. O padrão de esparsidade apresentado pelas submatrizes H_{33} e H_{34} varia muito com o tipo de problema. Muitos dos elementos de H_{33} que inicialmente eram nulos deixam de o ser devido ao *fill-in* que ocorre durante a eliminação de H_{32} . Por todos estes motivos a eliminação das colunas 3 e 4 é efectuada sem atender a uma eventual esparsidade, mas considerando a simetria. O tempo de CPU associado a esta fase revela-se assim claramente predominante na quase totalidade dos problemas.

Quando o método de eliminação de Gauss é aplicado à resolução de um sistema de equações cuja matriz dos coeficientes não é positiva definida, podem surgir elementos nulos na respectiva diagonal. Se a ordenação das linhas da matriz não for alterada, a presença destes elementos nulos conduz a uma divisão por zero, ficando assim comprometida a obtenção da solução do sistema de equações. Se na diagonal aparecerem elementos com um valor absoluto demasiado pequeno (e.g. $|a_{ii}| < 10^{-8}$), o sistema de equações pode ser resolvido mas apresenta uma solução pouco precisa. No caso da matriz Hessiana (4.8) é possível evitar o aparecimento de elementos nulos em H_{11} e H_{22} , substituindo por ε (e.g. 10^{-4}) as componentes de \tilde{s} e $\tilde{\lambda}^g$ cujo valor absoluto for inferior a ε . Nestas circunstâncias, a eliminação da coluna 1 coloca valores não nulos na diagonal de H_{22} . Na submatriz H_{33} podem aparecer elementos nulos na diagonal, apesar de se verificar que durante a eliminação de H_{32} são adicionados alguns valores não nulos à diagonal de H_{33} . No caso da submatriz H_{44} , são adicionados alguns valores não nulos à respectiva diagonal durante a eliminação de H_{43} . Os motivos apresentados justificam o facto de raramente aparecerem elementos nulos na diagonal da matriz Hessiana em circunstâncias que originem uma divisão por zero. Nos restantes casos a divisão por zero

pode ser evitada recorrendo a uma troca de linhas na matriz dos coeficientes. Esta troca de linhas deve ser acompanhada de uma troca de colunas para evitar a destruição da simetria da matriz Hessiana. Nestas circunstâncias deve ser procurado entre os elementos situados na diagonal abaixo do elemento nulo aquele que apresenta um maior valor absoluto. A referida troca de linhas e colunas faz com que este elemento ocupe o lugar do elemento nulo. Esta operação é designada pivotagem diagonal e deve considerar como nulos os elementos que apresentarem um valor absoluto demasiado pequeno.

Em raras situações pode acontecer que todos os elementos da diagonal abaixo do elemento nulo sejam também nulos. Nestes casos, a matriz Hessiana pode ser não singular e a pivotagem diagonal não resolve o problema da divisão por zero. A seguinte matriz constitui um exemplo relativo a esta situação particular

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad (4.44)$$

Estas raras situações podem ser ultrapassadas passando a considerar a matriz dos coeficientes como não simétrica e efectuando a clássica pivotagem parcial [Bur93]. É também possível contornar o problema substituindo o método de eliminação de Gauss pelo método iterativo que será adiante apresentado.

Nesta secção foram apresentadas algumas estratégias destinadas a aumentar a eficiência do algoritmo e a economizar memória durante a resolução do sistema de equações lineares. Convém contudo realçar que o algoritmo desenvolvido continua a apresentar uma generalidade que permite abordar a resolução de qualquer programa matemático do tipo (4.2).

4.4.2 - Método iterativo

No algoritmo de optimização apresentado neste capítulo, a resolução do sistema de equações lineares (4.42) pode ser efectuada recorrendo ao método dos gradientes conjugados. Este método está presente no algoritmo como uma alternativa à eliminação de Gauss referida na secção anterior. A selecção do método directo ou do iterativo constitui uma opção que é da responsabilidade do utilizador.

O método dos gradientes conjugados foi desenvolvido em 1952 por Hestenes e Stiefel [Hes52]. Os fundamentos do método e do algoritmo que foi utilizado na resolução do sistema de equações (4.42) encontram-se em [Lue84]. Este sistema de equações será de aqui em diante referido na versão (4.43), correspondendo a matriz \underline{A} à Hessiana (4.8) e o vector \underline{b} ao simétrico do gradiente do Lagrangeano (4.3). O método dos gradientes conjugados requer que a matriz \underline{A} seja simétrica, sendo desejável que ela seja também positiva definida. Quando a matriz \underline{A} é indefinida, o método dos gradientes conjugados continua a poder ser aplicado, mas apresenta o inconveniente de estar sujeito à ocorrência de divisões por zero ou por valores demasiado pequenos [Lue69]. Nestes casos, a eficiência ou mesmo o sucesso da resolução do sistema de equações podem ser gravemente comprometidos. Uma vez que a matriz Hessiana é na generalidade dos casos indefinida, é conveniente recorrer à seguinte transformação do sistema de equações (4.43)

$$\underline{A}^T \underline{A} \underline{x} = \underline{A}^T \underline{b} \quad (4.45)$$

A matriz dos coeficientes deste sistema de equações $\left(\underline{A}^T \underline{A} \right)$ é simétrica e positiva definida, possibilitando assim uma utilização mais robusta do método dos gradientes conjugados. A transformação de (4.43) em (4.45) apresenta no entanto o inconveniente de o coeficiente de estabilidade (*condition number*) da matriz $\underline{A}^T \underline{A}$ ser o quadrado do correspondente à matriz \underline{A} [Aro89]. Este facto faz com que o número de iterações que é necessário efectuar para obter a solução de (4.45) seja significativamente superior ao correspondente a um sistema de equações cuja matriz dos coeficientes apresente um coeficiente de estabilidade da ordem de grandeza do de \underline{A} .

O método dos gradientes conjugados torna-se mais eficiente se for efectuada uma transformação designada pré-condicionamento, que tem como objectivo uma diminuição do coeficiente de estabilidade da matriz dos coeficientes. No caso do sistema de equações (4.45) e pelos motivos atrás referidos, a realização do pré-condicionamento revela-se particularmente importante. Multiplicando ambos os membros de (4.45) por \underline{C}^T e inserindo $\underline{C} \underline{C}^{-1}$, obtém-se o seguinte sistema de equações

$$\overline{\underline{A}} \overline{\underline{x}} = \overline{\underline{b}} \quad (4.46)$$

em que

$$\overline{\underline{A}} = \underline{C}^T \underline{A}^T \underline{A} \underline{C} \quad (4.47)$$

$$\bar{\tilde{x}} = \tilde{C}^{-1} \tilde{x} \quad (4.48)$$

$$\bar{\tilde{b}} = \tilde{C}^T \tilde{A}^T \tilde{b} \quad (4.49)$$

A matriz de pré-condicionamento \tilde{C} é quadrada, invertível e destina-se a tornar o sistema de equações bem condicionado. No presente trabalho, o pré-condicionamento é efectuado com a seguinte matriz diagonal [Fri70].

$$C_{ii} = \left[D_{ii} \left(\tilde{A}^T \tilde{A} \right) \right]^{-1/2} \quad (4.50)$$

Em (4.50), $D \left(\tilde{A}^T \tilde{A} \right)$ é uma matriz diagonal, cujos elementos D_{ii} são os elementos da diagonal de $\tilde{A}^T \tilde{A}$. Considerando um vector auxiliar \tilde{S} definido do seguinte modo

$$S_i = \text{norma da coluna } i \text{ da matriz } \tilde{A} \quad (4.51)$$

verifica-se facilmente que

$$D_{ii} \left(\tilde{A}^T \tilde{A} \right) = S_i^2 \quad (4.52)$$

Uma vez que a matriz \tilde{A} é simétrica, as componentes de \tilde{S} também correspondem às normas das linhas de \tilde{A} . Substituindo (4.52) em (4.50) obtém-se uma expressão para C_{ii} que permite efectuar facilmente o seu cálculo a partir dos elementos de \tilde{A} .

$$C_{ii} = S_i^{-1} \quad (4.53)$$

Deste modo o pré-condicionamento pode ser realizado sem ser necessário efectuar o cálculo de $\tilde{A}^T \tilde{A}$. O cálculo da matriz \tilde{C} recorrendo a (4.51) e (4.53) revela-se particularmente eficiente quando a matriz \tilde{A} é esparsa e o armazenamento dos respectivos termos não nulos é efectuado por linhas ou por colunas. É óbvio que este tipo de pré-condicionamento não pode ser efectuado se em \tilde{A} existir uma linha/coluna com todos os elementos nulos. Uma vez que a matriz \tilde{C} só apresenta termos não nulos na diagonal, o seu armazenamento pode ser efectuado num vector, com um número de componentes igual ao número de linhas de \tilde{C} .

Na Figura 4.6 encontra-se o algoritmo correspondente à resolução de (4.46) pelo método dos gradientes conjugados. Uma vez que é considerada uma solução inicial \bar{x}_0 nula, o cálculo do resíduo inicial na fase C) é efectuado com a seguinte expressão

$$\underline{r}_0 = \underline{\bar{A}} \underline{\bar{x}}_0 - \underline{\bar{b}} = -\underline{\bar{b}} = -\underline{C}^T \underline{A}^T \underline{b} \quad (4.54)$$

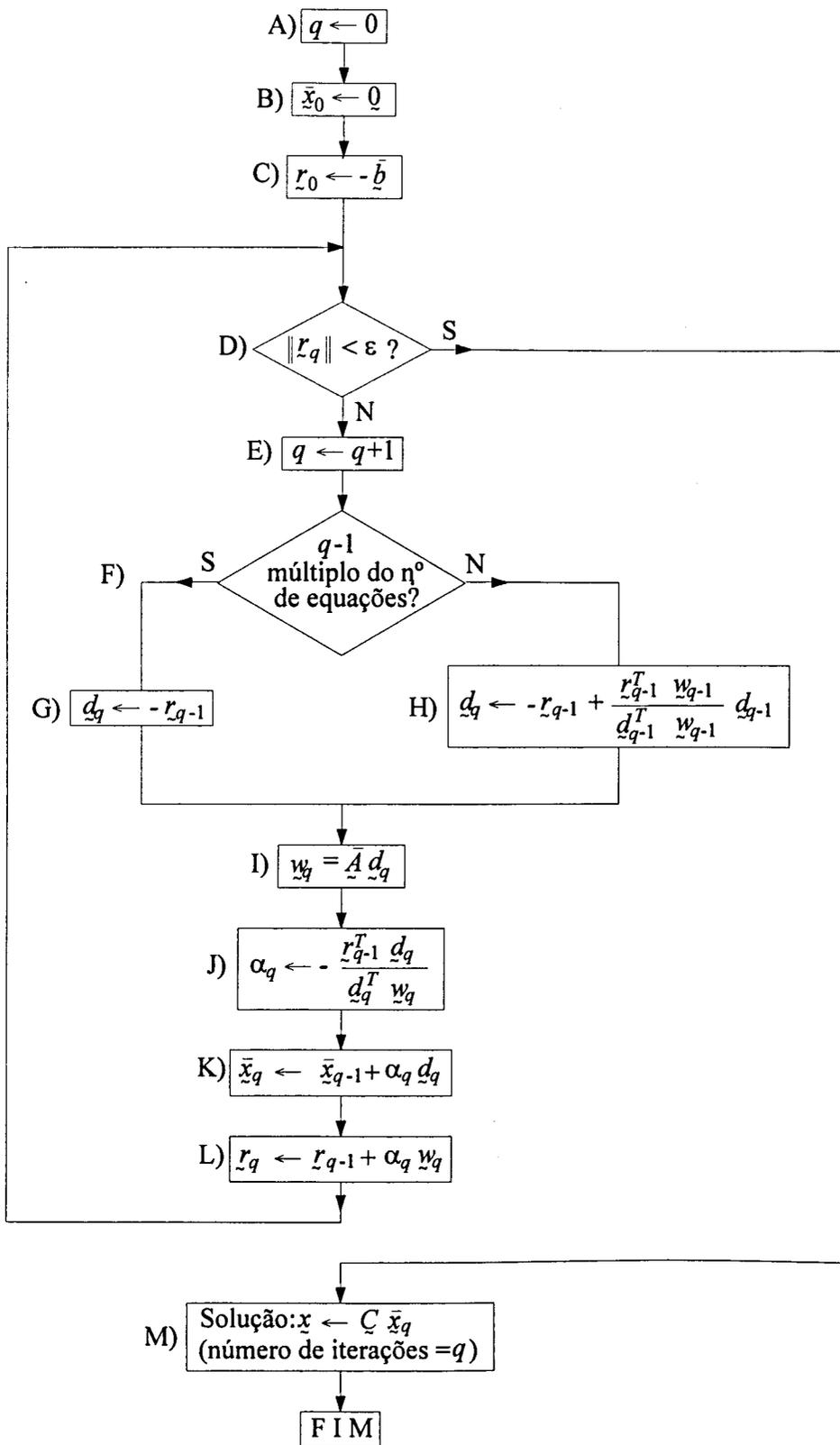
As operações correspondentes ao cálculo de \bar{b} com a expressão (4.49) devem ser efectuadas da direita para a esquerda, i.e., deve ser efectuado em primeiro lugar o produto matriz-vector $\underline{A}^T \underline{b}$. O vector resultante é em seguida multiplicado pelo vector em que está armazenada a diagonal da matriz \underline{C} . Na fase I) é calculado o vector auxiliar \underline{w}_q recorrendo à seguinte expressão em que $\underline{\bar{A}}$ se encontra substituído por (4.47).

$$\underline{w}_q = \underline{\bar{A}} \underline{d}_q = \underline{C}^T \underline{A}^T \underline{A} \underline{C} \underline{d}_q \quad (4.55)$$

Nesta expressão os quatro produtos devem também ser efectuados da direita para a esquerda, de modo a constituírem dois produtos matriz-vector e dois produtos entre vectores. Assim é de novo evitado o volumoso cálculo de $\underline{A}^T \underline{A}$. Na fase M) é calculada a solução do sistema de equações original, recorrendo à transformação inversa de (4.48). As restantes fases do algoritmo da Figura 4.6 apenas envolvem operações triviais com vectores e com escalares. Na generalidade dos casos, estas operações são menos exigentes do ponto de vista computacional do que as correspondentes à fase I).

O algoritmo da Figura 4.6 apresenta uma estrutura semelhante à do algoritmo de optimização da Figura 3.1, correspondendo a direcção \underline{d} à variação da solução $\Delta \underline{x}$. Esta analogia é devida ao facto de a resolução de um sistema de equações lineares pelo método dos gradientes conjugados corresponder à minimização de uma função quadrática [Lue84].

Uma vez que no algoritmo da Figura 4.6 as únicas operações que envolvem a matriz simétrica \underline{A} são produtos matriz-vector, é suficiente armazenar os elementos não nulos do respectivo triângulo superior. Verifica-se ainda que a disposição destes elementos na matriz \underline{A} é irrelevante, porque quando é utilizado o método dos gradientes conjugados não existe *fill-in*. No sistema de equações lineares relativo a cada iteração do método de Newton, a matriz dos coeficientes \underline{A} corresponde à matriz Hessiana (4.8). Uma vez que na generalidade dos casos esta matriz é muito esparsa (ver Figura 4.2), o método dos gradientes conjugados apresenta em relação aos métodos directos significativas vantagens no consumo de memória.



Nota: na primeira iteração ($q = 1$) é executada a instrução G).

Figura 4.6 - Algoritmo correspondente ao método dos gradientes conjugados com pré-condicionamento.

O tempo de CPU relativo à resolução de um sistema de equações lineares pelo método dos gradientes conjugados depende essencialmente do número de iterações e do tempo de CPU associado a cada produto matriz-vector [Aze90]. O número de iterações é difícil de estimar antecipadamente porque depende do coeficiente de estabilidade e do número de valores próprios distintos da matriz dos coeficientes [Fri70]. Quando o número de equações é muito elevado, a acumulação de erros de arredondamento contribui também para um aumento do número de iterações. O tempo de CPU correspondente a um produto matriz-vector depende essencialmente do número de termos não nulos da matriz Hessiana. Uma vez que na generalidade dos casos a quantificação de todos estes parâmetros é difícil de concretizar, revela-se fundamental testar os vários métodos disponíveis com diferentes tipos de problemas. No Capítulo 6 são apresentadas algumas conclusões de um estudo deste tipo. Pode no entanto adiantar-se que na resolução de (4.7) o método dos gradientes conjugados necessita quase sempre de um número muito elevado de iterações e, conseqüentemente, de um tempo de CPU elevado, ficando assim em clara desvantagem relativamente ao método de eliminação de Gauss. O método dos gradientes conjugados apresenta uma característica, que em certas situações é muito útil e que consiste no facto de ser capaz de calcular uma das soluções de um sistema de equações indeterminado. Quando se utiliza um método directo, a ocorrência de uma matriz Hessiana singular impede a resolução do problema de optimização pelo método de Newton.

4.5 - PESQUISA UNIDIMENSIONAL

Na fase F) do algoritmo da Figura 3.1 é efectuado o cálculo de um parâmetro (α^q) que modifica a grandeza do vector $\Delta \tilde{x}^q$. Nas considerações que se seguem o parâmetro α^q é designado apenas α . Esta fase, que é genericamente designada pesquisa unidimensional (*line search*), tem como objectivo a minimização do erro na direcção $\Delta \tilde{x}^q$. Quando o método de Newton é utilizado na resolução de um problema de optimização com restrições, o erro E que se pretende minimizar na direcção $\Delta \tilde{x}^q$ corresponde à norma do gradiente do Lagrangeano (ver secção 4.1).

$$\text{Min. } E(\alpha) = \left\| \nabla L \left(\tilde{X}^{q-1} + \alpha \Delta \tilde{X}^q \right) \right\| \quad (4.56)$$

No problema de minimização (4.56), \tilde{X}^{q-1} e $\Delta \tilde{X}^q$ são considerados constantes. Nestas circunstâncias, a minimização de $E(\alpha)$ constitui um problema com uma só variável, sendo possível efectuar a sua resolução com um dos métodos que foram apresentados na secção 3.1. No presente trabalho é utilizado o método das bissecções sucessivas (ver secção 3.1.2), seguido de uma aproximação polinomial do segundo grau (ver secção 3.1.1). O cálculo das derivadas $dE/d\alpha$ que são utilizadas no método das bissecções sucessivas é efectuado por diferenças finitas. O intervalo a partir do qual são iniciadas as bissecções sucessivas é, na generalidade dos casos, o seguinte

$$[\alpha_{\min}, \alpha_{\max}] = [10^{-3}, 1] \quad (4.57)$$

Uma vez que a adopção de um valor de α nulo tem como consequência a manutenção da solução corrente \tilde{X}^{q-1} , é sugerido um valor mínimo de 10^{-3} . Quando o método de Newton é aplicado à minimização de uma função quadrática o valor de α deve ser unitário [Van84]. Na generalidade dos problemas, verifica-se que durante o processo iterativo os valores de α que minimizam o erro (4.56) são inferiores à unidade. Numa vizinhança da solução óptima a aproximação quadrática considerada passa a ser muito rigorosa e nessas circunstâncias o valor de α deve ser unitário. Uma vez que se considera que a direcção $\Delta \tilde{X}^q$ está sempre orientada para a solução óptima, não são admitidos valores de α negativos. Com base na experiência adquirida na resolução de inúmeros problemas de optimização, é possível afirmar que o valor ideal de α raramente é superior à unidade e que nas primeiras 5 a 10 iterações o limite superior do intervalo (α_{\max}) deve ser inferior a 0.5. Esta diminuição de α_{\max} destina-se a impedir a divergência do método de Newton nas primeiras iterações.

Antes de iniciar as bissecções sucessivas é fundamental verificar se a função $E(\alpha)$ é convexa no intervalo $[\alpha_{\min}, \alpha_{\max}]$. Esta verificação é simplificada recorrendo à análise do sinal de $E'(\alpha_{\min})$ e $E'(\alpha_{\max})$. No Quadro 4.2 encontram-se as diferentes situações que são consideradas e, em cada caso, a decisão relativa à minimização de $E(\alpha)$.

Nas raras situações em que a função $E(\alpha)$ se apresenta muito irregular o diagnóstico indicado no Quadro 4.2 pode conduzir a conclusões erradas. Nesses casos a amplitude do intervalo $[\alpha_{\min}, \alpha_{\max}]$ deve ser diminuída.

Quadro 4.2 - Diagnóstico simplificado do tipo de função $E(\alpha)$.

$E'(\alpha_{\min})$	$E'(\alpha_{\max})$	Tipo de função	Valor de α adoptado
0	0	Constante	α_{\max}
-	-	Linear decrescente	α_{\max}
+	+	Linear crescente	α_{\min}
+	-	Côncava	α_{\min} ou α_{\max}
-	+	Convexa	Resultado da minimização de $E(\alpha)$

1)

2)

1) Se $E(\alpha_{\min}) < E(\alpha_{\max})$ é adoptado α_{\min} senão é adoptado α_{\max} .

2) É efectuado um reduzido número de bissecções sucessivas seguidas de uma aproximação polinomial do segundo grau.

Na generalidade dos casos, o tempo de CPU relativo à pesquisa unidimensional é muito inferior ao que é despendido no cálculo de Δx^q . Por este motivo, na fase correspondente ao cálculo de α^q é considerada mais importante a robustez do método utilizado do que a sua eficiência. Foi também experimentalmente verificado que não há necessidade de resolver o problema (4.56) com uma precisão muito elevada, porque o método de Newton consegue corrigir eventuais erros associados a esta fase.

4.6 - INTERRUPÇÃO DO PROCESSO ITERATIVO

O processo iterativo correspondente à resolução de problemas de optimização pelo método de Newton (ver Figura 3.1) deve ser interrompido quando for alcançado um número máximo de iterações, quando deixar de haver uma progressão significativa em direcção à solução óptima ou quando for satisfeito o critério de convergência (4.10). Destas três situações, só a última é que indica que o processo iterativo foi bem sucedido e que a solução final pode ser a solução óptima pretendida. A verificação da condição (4.10) não é suficiente para garantir a qualidade da solução obtida, porque também é verificada em mínimos locais e em pontos que não satisfazem as condições de Karush-Kuhn-Tucker (ver Capítulo 2). Nestes casos o processo iterativo deve ser repetido com outra solução inicial.

Uma vez que o método de Newton apresenta convergência quadrática, o valor do parâmetro ε em (4.10) pode ser muito pequeno. No presente trabalho foi quase sempre atribuído a ε o valor 10^{-5} . Se forem utilizadas as técnicas de *scaling* referidas na secção 4.3, obtém-se uma solução final com cerca de cinco algarismos significativos correctos. Na resolução de problemas de optimização pelo método de Newton é muito frequente a obtenção de soluções às quais corresponde um valor de ε inferior a 10^{-7} . Nestes casos a solução apresenta cerca de sete algarismos significativos correctos. No Capítulo 6 é apresentada uma comparação entre a solução exacta e a solução numérica de um problema de optimização, tendo em vista o relacionamento entre o valor de ε e o número de algarismos significativos correctos na solução final.

Para obter soluções de elevada precisão nos casos em que é utilizado o método directo na resolução do sistema de equações lineares (4.7), é necessário acrescentar ao algoritmo de optimização a técnica que é em seguida apresentada. Atendendo às características iniciais da matriz Hessiana (4.8) e ao resultado da eliminação de Gauss da submatriz H_{21} (ver Figuras 4.2 e 4.3), verifica-se que na diagonal de H_{11} e H_{22} surgem valores nulos quando as correspondentes componentes de s_i e λ_i^s forem nulas. Uma vez que na eliminação de Gauss sem pivotagem não podem existir elementos nulos na diagonal, a presença de componentes de s e λ^s nulas pode impedir a resolução do sistema de equações lineares. Se em vez de valores nulos estiverem presentes na diagonal valores muito pequenos, a solução do sistema de equações lineares pode apresentar um erro inaceitável. Verifica-se também que para satisfazer as equações não lineares (4.5 a) é necessário que pelo menos metade das componentes de s e λ^s sejam nulas. O algoritmo relativo ao método de Newton apresenta logo nas primeiras iterações uma grande tendência para tornar algumas componentes de s e λ^s muito pequenas.

Uma vez que na eliminação de H_{21} e H_{32} não é efectuada pivotagem com o objectivo de evitar o *fill-in*, revela-se fundamental impedir que durante o processo iterativo as componentes de s e λ^s adoptem valores absolutos inferiores a um parâmetro ε_X . Com base na experiência acumulada na resolução de diversos problemas, sugere-se para este parâmetro um valor de 10^{-5} a 10^{-4} . Uma vez que ao impedir que as variáveis adoptem valores absolutos inferiores a ε_X se está a introduzir um erro na solução, é necessário que na última iteração esta restrição seja eliminada. Também com base na experiência, considera-se que a última iteração vai ser efectuada se em três iterações consecutivas o erro for inferior a um parâmetro ε_L . Para ε_L sugere-se um valor entre 10^{-3} e 10^{-2} . Depois de efectuadas as três referidas iterações, as variáveis que adoptam valores muito pequenos deixam de ser substituídas por ε_X e, na generalidade dos casos, ocorre um grande decréscimo do erro.

Na Figura 4.7 está representada a evolução do erro durante o processo iterativo correspondente à resolução de um problema de otimização pelo método de Newton. Após a ocorrência de três iterações consecutivas com um erro inferior a 10^{-3} , as variáveis com valores pequenos deixam de ser substituídas por 10^{-5} e na última iteração o erro decresce cerca de 100 000 vezes. Neste problema foi possível obter uma solução final à qual corresponde uma norma do gradiente do Lagrangeano (4.10) próxima de 10^{-9} . Nos raros casos em que o processo iterativo não apresenta este tipo de comportamento, nas últimas iterações a resolução do sistema de equações lineares (4.7) deve ser efectuada recorrendo ao método dos gradientes conjugados. Deste modo, mesmo que a matriz Hessiana se torne singular, é possível continuar a avançar para a solução óptima.

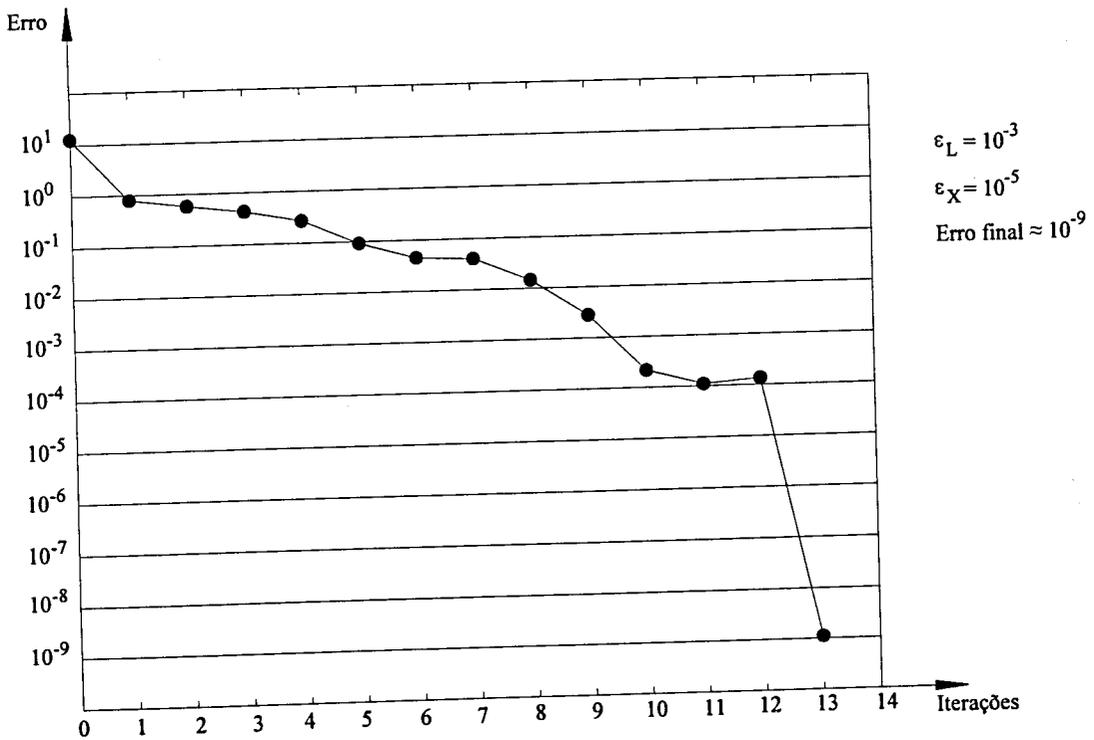


Fig. 4.7 - Evolução do erro durante o processo iterativo

4.7 - OBSERVAÇÕES FINAIS

A utilização do método de Newton na resolução de problemas de optimização é desaconselhada por alguns autores, devido ao facto de apresentar significativas desvantagens relativamente a outros métodos. Em [Aro89] são apresentados como graves inconvenientes a necessidade de calcular e armazenar segundas derivadas, a resolução em cada iteração de um sistema com um elevado número de equações lineares e o facto de não ser possível resolver o problema se numa das iterações a matriz Hessiana for singular. Em [Gil81] é referido que a aproximação quadrática só é rigorosa numa vizinhança da solução corrente, ficando assim o sucesso da aplicação do método de Newton dependente da selecção de uma solução inicial suficientemente próxima da solução final. Numa fase preliminar do presente trabalho foi confirmada a existência destas desvantagens. No entanto, com a utilização das diversas técnicas que foram apresentadas no presente capítulo, foi possível atenuar ou mesmo ultrapassar os inconvenientes que habitualmente são associados ao método de Newton. Torna-se assim possível beneficiar da sua principal qualidade que consiste na convergência quadrática. No Capítulo 6 são apresentados diversos problemas de optimização com um elevado número de variáveis e restrições. A resolução destes problemas permitiu constatar que o método de Newton pode ser utilizado com eficiência e versatilidade.

CAPÍTULO 5

PROGRAMA DE CÁLCULO AUTOMÁTICO

No âmbito do presente trabalho foi desenvolvido um programa de computador destinado à resolução de problemas de optimização pelo método de Newton. Este programa baseia-se no algoritmo descrito no Capítulo 4 e é genericamente designado NEWTOP. Neste capítulo são apresentadas algumas características e funcionalidades associadas ao programa de cálculo automático, que nele foram incluídas com o objectivo de facilitar a resolução de diferentes tipos de problemas de optimização.

No desenvolvimento de programas de optimização, a linguagem que é habitualmente utilizada é o FORTRAN 77 [Aro89] [Van89a] [NAG91]. As primeiras versões do programa NEWTOP foram também escritas em FORTRAN 77, tendo-se nessa altura constatado que as limitações da linguagem iriam dificultar o desenvolvimento do programa no sentido de ele se tornar versátil, eficiente e fácil de utilizar. Com o objectivo de ultrapassar as dificuldades encontradas, foi efectuada a reconversão de todo o *software* para a linguagem ANSI C [Ker88]. Esta linguagem de programação apresenta uma grande versatilidade, encontra-se standardizada e pode ser utilizada na quase totalidade das plataformas informáticas actualmente disponíveis. No desenvolvimento do programa NEWTOP, foi possível tirar partido das seguintes vantagens que a linguagem C apresenta em relação ao FORTRAN 77:

- *Gestão da memória* - é possível requisitar memória para os vectores/matrizes ao nível de cada subrotina e em seguida libertá-la para que possa vir a ser utilizada por outros vectores/matrizes. A quantidade de memória que é requisitada pode ser a estritamente necessária. Com este tipo de gestão da memória, num programa que disponha de diversos algoritmos alternativos, é possível em fase de execução só requisitar memória para os vectores/matrizes correspondentes ao algoritmo seleccionado. Existe ainda a vantagem de um só programa executável poder ser utilizado em computadores com diferentes quantidades de memória, tirando sempre partido de toda a memória existente.

- *Manipulação de ficheiros* - a linguagem C apresenta uma grande versatilidade na leitura e escrita de ficheiros. Revela-se particularmente importante a possibilidade de acesso directo a ficheiros com um conteúdo heterogéneo.
- *Ponteiros* - a utilização de ponteiros na manipulação de vectores e matrizes [Pre92] apresenta uma grande versatilidade. Esta vantagem revela-se particularmente importante na manipulação de cadeias de caracteres (*strings*).

Com o objectivo de tornar o programa NEWTOP o mais portátil possível, foram evitadas eventuais dependências do *hardware* e do sistema operativo, bem como o recurso a funções não *standard*. Nos módulos de visualização gráfica foi utilizada a biblioteca *Xlib* do *X Window System* [Nye90]. Uma vez que esta biblioteca raramente se encontra disponível em sistemas não UNIX, foram desenvolvidas funções que simulam o seu comportamento e que podem ser facilmente adaptadas a qualquer plataforma informática. Presentemente o programa NEWTOP e os respectivos módulos de visualização gráfica podem ser utilizados em plataformas UNIX e MSDOS.

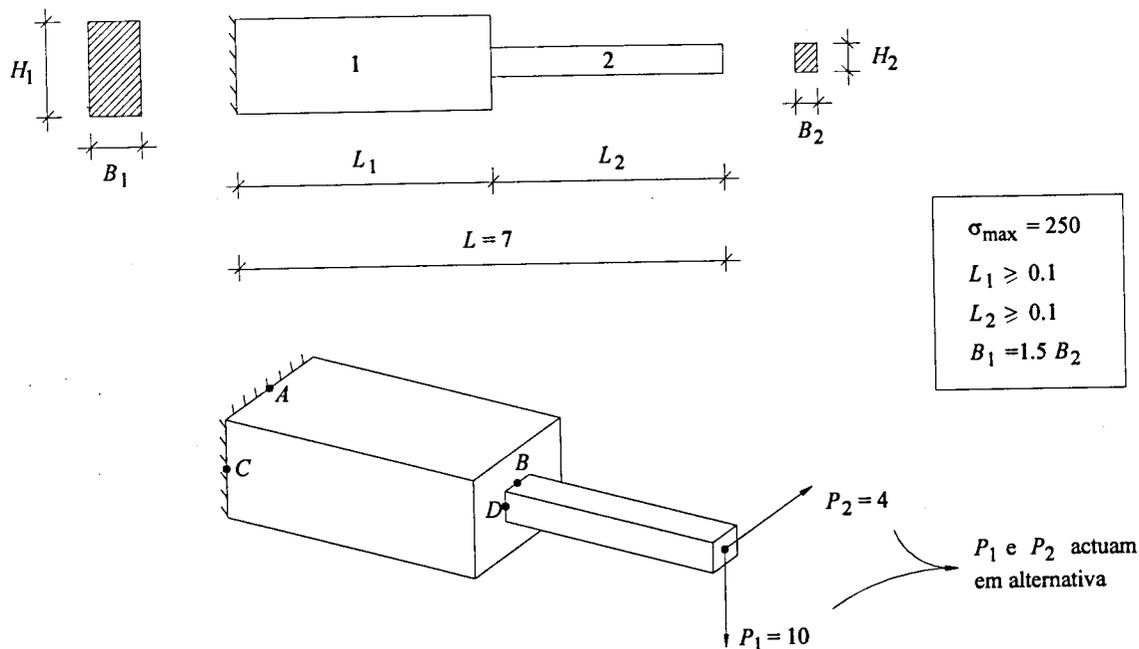
São em seguida apresentados alguns aspectos associados à utilização do programa NEWTOP. A respectiva descrição é, sempre que possível, exemplificada recorrendo a um problema de optimização estrutural.

5.1 - DESCRIÇÃO DO PROGRAMA MATEMÁTICO

O programa matemático correspondente ao problema de optimização que se pretende resolver tem de ser de alguma forma fornecido ao programa de computador. Na secção 4.2 foram indicadas algumas técnicas destinadas à realização desta tarefa, sendo em seguida apresentada com mais pormenor a que é utilizada no programa NEWTOP.

A apresentação do modo como é descrito um programa matemático e alguns aspectos com ele relacionados é exemplificada com o seguinte problema de optimização estrutural:

- Minimizar o volume de uma viga em consola com as características indicadas na Figura 5.1.



Nota: com o objectivo de simplificar a formulação do problema não é considerado o peso próprio.

Figura 5.1 - Problema de minimização do volume de uma viga em consola.

O programa matemático correspondente ao problema da Figura 5.1 encontra-se na Listagem 5.1. Uma vez que se trata de um problema destinado apenas a exemplificar algumas características do programa NEWTOP, é considerada uma formulação em que figuram como variáveis as seguintes grandezas

$$\tilde{x} = (P_1, P_2, L_1, L_2, L, B_1, H_1, B_2, H_2) \tag{5.1}$$

Listagem 5.1 - Programa matemático correspondente ao problema indicado na Figura 5.1.

```

### Main title of the nonlinear program
Optimizacao de uma consola com duas seccoos

Min.
    L1 * B1 * H1 + L2 * B2 * H2 ;

s.t.i.c.
    Tensao em A:  6 * P1 * L * B1 ^ -1 * H1 ^ -2 - 250 < 0 ;
    Tensao em B:  6 * P1 * L2 * B2 ^ -1 * H2 ^ -2 - 250 < 0 ;
    Tensao em C:  6 * P2 * L * B1 ^ -2 * H1 ^ -1 - 250 < 0 ;
    Tensao em D:  6 * P2 * L2 * B2 ^ -2 * H2 ^ -1 - 250 < 0 ;
# Side constraints
    L1 minimo:   - L1 + 0.1 < 0 ;
    L2 minimo:   - L2 + 0.1 < 0 ;

s.t.e.c.
    Comprimento total:  L - L1 - L2 = 0 ;
    Relacao entre B1 e B2:  B1 - 1.5 * B2 = 0 ;
    Valor de L:  L - 7 = 0 ;
    Valor de P1:  P1 - 10 = 0 ;
    Valor de P2:  P2 - 4 = 0 ;

END_OF_FILE

```

De acordo com a Listagem 5.1, são consideradas restrições de tensão nos pontos A e B quando actua P_1 e nos pontos C e D quando actua P_2 . As duas restrições que impõe um valor mínimo de 0.1 para L_1 e L_2 destinam-se a impedir a convergência do processo iterativo para soluções com L_1 ou L_2 nulo. Verificou-se que este tipo de soluções correspondem a mínimos locais com um volume superior ao mínimo global.

Na listagem 5.1 estão também presentes diversas restrições igualdade. A primeira relaciona os comprimentos dos dois tramos L_1 e L_2 com o comprimento total L . A segunda restrição constitui uma exigência de proporcionalidade entre as variáveis B_1 e B_2 . Esta restrição igualdade é a única que pode ser omitida sem comprometer o significado físico do problema. As restantes restrições igualdade apenas se destinam a fixar os valores de L , P_1 e P_2 . Neste problema de optimização seria muito simples substituir as cinco restrições igualdade nas restantes funções do programa matemático, porque é possível transformá-las de modo a explicitar cinco variáveis distintas. Esta operação não foi efectuada, porque neste capítulo apenas se pretende exemplificar alguns aspectos relacionados com a utilização do programa NEWTOP na resolução de problemas de optimização genéricos.

No ficheiro de dados cujo conteúdo se encontra na Listagem 5.1 é possível identificar os seguintes componentes:

- *Comentários opcionais* - todos os caracteres que se encontram após "#";
- *Título geral do problema*
- *Palavras chave*
 - Min. - *minimize* - minimizar - precede a função objectivo
 - s.t.i.c. - *subject to inequality constraints* - sujeito a restrições desigualdade - precede as restrições desigualdade
 - s.t.e.c. - *subject to equality constraints* - sujeito a restrições igualdade - precede as restrições igualdade
 - END_OF_FILE - fim do ficheiro;
- *Títulos das restrições* - precedem cada restrição e terminam com o carácter ":";
- *Funções* (cada função pode ocupar mais do que uma linha)
 - função objectivo - termina apenas com ";"
 - restrições desigualdade - terminam com "< 0 ;"
 - restrições igualdade - terminam com "= 0 ;"

As diversas funções que figuram no programa matemático têm de ser do tipo (4.11) (ver secção 4.2). No ficheiro de dados do programa NEWTOP, cada termo tem de estar de acordo com a seguinte sintaxe

$$[+|-][c_i][*][x_1][^ a_{i1}][*x_2][^ a_{i2}] \dots \quad (5.2)$$

Em (5.2) o coeficiente c_i pode ser um número real qualquer e os expoentes a_{ii} têm de ser números inteiros sem restrição de sinal. Às variáveis devem ser atribuídos nomes sugestivos iniciados por uma letra. O sinal que precede o termo só é opcional se se tratar do primeiro termo da função. Se algum c_i ou a_{ii} for omitido admite-se que o seu valor é unitário. Os termos que apresentarem um valor de c_i nulo são ignorados, bem como as variáveis com um valor de a_{ii} nulo. Apresentam-se em seguida alguns exemplos de termos de funções destinados a esclarecer a sintaxe expressa em (5.2) e as observações efectuadas. É também indicado o modo como o termo é interpretado pelo programa NEWTOP.

$$- - 5.7 \quad \longrightarrow \quad 5.7 \quad (5.3a)$$

$$x1 \quad \longrightarrow \quad x_1 \quad (5.3b)$$

$$-x_1 \longrightarrow -x_1 \quad (5.3c)$$

$$5.7 * x_1 \longrightarrow 5.7 x_1 \quad (5.3d)$$

$$- 5.7 * x_1^{-3} \longrightarrow -5.7 x_1^{-3} \quad (5.3e)$$

$$+ - 5.7 * x_1^{-3} * x_2 \longrightarrow - 5.7 x_1^{-3} x_2 \quad (5.3f)$$

$$+ + 5.7 * B_{26}^{-3} * BETA^2 \longrightarrow 5.7 B_{26}^{-3} \beta^2 \quad (5.3g)$$

$$- 5.7 * d_{5,9} * Tmax^0 \longrightarrow - 5.7 d_{5,9} \quad (5.3h)$$

$$- + 0.0 * a * b \longrightarrow 0.0 \quad (5.3i)$$

O módulo do programa NEWTOP que procede à leitura e interpretação de um programa matemático do tipo indicado na Listagem 5.1 efectua uma validação exaustiva de toda a informação fornecida. Se for encontrado algum erro é assinalada a linha em que ele se encontra e é apresentada uma breve descrição da anomalia.

5.2 - CÁLCULO DO VALOR INICIAL DAS VARIÁVEIS DEPENDENTES

Na generalidade dos casos, verifica-se que o método de Newton apresenta uma maior robustez quando a solução inicial é admissível. É também desejável que nenhuma restrição desigualdade se encontre activa no ponto correspondente à solução inicial. Atendendo ao significado físico do problema, é em geral simples seleccionar uma solução inicial que verifique com alguma folga a totalidade das restrições desigualdade. Nos problemas de optimização estrutural é habitual iniciar o processo iterativo com uma solução suficientemente sobredimensionada, de modo que todas as restrições de tensão e de deslocamento sejam verificadas. Para que a solução inicial seja admissível é ainda necessário que todas as restrições igualdade sejam rigorosamente verificadas. Em muitas situações, para se conseguir uma solução inicial com estas características é necessário efectuar alguns cálculos preliminares recorrendo a outros programas de computador, que podem eventualmente não estar disponíveis. Para evitar estes inconvenientes, foram incluídos no programa NEWTOP um conjunto de procedimentos que efectuam alterações na solução inicial fornecida de modo que as restrições igualdade passem a ser respeitadas.

Num problema de optimização com n variáveis de projecto, m restrições desigualdade e l restrições igualdade, o conjunto das variáveis de projecto deve ser considerado dividido em

dois subconjuntos. Esta separação baseia-se no facto de ser em geral possível recorrer às l restrições igualdade para calcular o valor de igual número de variáveis, depois de serem fixados os valores das restantes $n - l$ variáveis. O conjunto das restrições igualdade constitui assim um sistema de l equações não lineares com l incógnitas. Por este motivo, as $n - l$ variáveis cujo valor inicial é arbitrariamente fixado são designadas variáveis independentes. Às restantes l variáveis deve ser atribuído um valor inicial que resulta da resolução do referido sistemas de equações, sendo por esse motivo designadas variáveis dependentes. A selecção das $n - l$ variáveis que são consideradas independentes depende do tipo de problema e do significado físico das variáveis. No caso da optimização de estruturas com comportamento linear, sugere-se que sejam consideradas como variáveis dependentes as reacções de apoio e os deslocamentos dos nós, porque é possível formular igual número de equações de equilíbrio, que figuram no programa matemático como restrições igualdade (ver Capítulo 6). Em problemas mais genéricos a selecção das variáveis dependentes deve ser efectuada de modo que seja possível obter a solução do sistema de l equações não lineares e l incógnitas constituído pelas restrições igualdade. No programa NEWTOP este sistema de equações é resolvido pelo método de Newton, após a substituição das variáveis independentes pelos valores que figuram na solução inicial. Esta aplicação do método de Newton para calcular os valores iniciais das variáveis dependentes é muito semelhante à resolução do sistema de equações que resulta do anulamento do gradiente do Lagrangeano (ver Capítulo 4).

No problema de optimização apresentado na secção 5.1 (ver Listagem 5.1) figuram nove variáveis ($n = 9$) e cinco restrições igualdade ($l = 5$). Foram consideradas independentes as seguintes $n - l$ variáveis às quais foram atribuídos os seguintes valores iniciais

$$L_1 = 3.5 \quad (5.4a)$$

$$H_1 = 1.5 \quad (5.4b)$$

$$B_2 = 0.6 \quad (5.4c)$$

$$H_2 = 1.2 \quad (5.4d)$$

Substituindo estes valores nas restrições igualdade, obtém-se o seguinte sistema de cinco equações, que neste exemplo simples são lineares

$$L - 3.5 - L_2 = 0 \quad (5.5a)$$

$$B_1 - 0.9 = 0 \quad (5.5b)$$

$$L - 7 = 0 \quad (5.5c)$$

$$P_1 - 10 = 0 \quad (5.5d)$$

$$P_2 - 4 = 0 \quad (5.5e)$$

Da resolução deste sistema de equações resultam os seguintes valores iniciais das variáveis dependentes

$$P_1 = 10.0 \quad (5.6a)$$

$$P_2 = 4.0 \quad (5.6b)$$

$$L_2 = 3.5 \quad (5.6c)$$

$$L = 7.0 \quad (5.6d)$$

$$B_1 = 0.9 \quad (5.6e)$$

Os valores indicados em (5.4) e (5.6) constituem assim uma solução inicial admissível, porque se verifica que, além das restrições igualdade, as restrições desigualdade indicadas na Listagem 5.1 são também respeitadas.

A resolução do sistema de l equações (5.5) pelo método de Newton requer uma solução inicial para as l variáveis presentes. Se as equações forem todas lineares, a solução inicial pode ser nula, sendo o sistema resolvido com uma única iteração. Se estiverem presentes equações não lineares, deve ser fornecida uma solução inicial não nula e com valores de ordem de grandeza compatível com o significado físico das variáveis. Nestes casos torna-se em geral necessário efectuar mais do que uma iteração para se obter uma solução inicial que respeite as restrições igualdade. Se este processo iterativo preliminar não for bem sucedido, é sempre possível tentar resolver o problema de optimização com uma solução inicial não admissível.

5.3 - PRÉ-SUBSTITUIÇÃO DE VARIÁVEIS

Quando a uma restrição igualdade corresponde uma expressão simples em que uma das variáveis pode ser explicitada, é possível e vantajoso efectuar a substituição dessa variável nas restantes funções do programa matemático. Após esta operação, a restrição igualdade pode ser suprimida e o problema de optimização passa a ter menos uma variável. Esta redução do número de restrições igualdade e do número de variáveis permite na generalidade dos

casos obter mais rapidamente a solução do programa matemático e conduz a uma economia de memória.

Atendendo ao tipo de funções que são manipuladas pelo programa NEWTOP (ver secção 4.2), é possível efectuar a pré-substituição dos seguintes tipos de expressões sem correr o risco de aumentar significativamente o volume de informação associado ao programa matemático

$$x_i = c \quad (5.7)$$

$$x_j = c x_k \quad (5.8)$$

Em (5.7) e (5.8) c é uma constante real com um valor fixo e predefinido. As restrições igualdade deste tipo podem ser suprimidas após a substituição de x_i e x_j em todas as funções que constituem o programa matemático. Apresenta-se em seguida um exemplo destinado a esclarecer as diversas operações associadas à pré-substituição de variáveis.

Substituir

$$x_1 = 2x_3 \quad (5.9)$$

no seguinte termo de uma das funções que constituem o programa matemático

$$\dots + 4x_1^3 x_2^{-2} x_3^{-1} + \dots \quad (5.10)$$

Após a substituição o termo (5.10) passa a ser o seguinte

$$\dots + 32x_3^3 x_2^{-2} x_3^{-1} + \dots \quad (5.11)$$

Para que o cálculo de derivadas venha a ser correctamente efectuado, é necessário proceder à simplificação do termo (5.11), resultando

$$\dots + 32x_2^{-2} x_3^2 + \dots \quad (5.12)$$

As operações relativas à pré-substituição de (5.9) em (5.10) e as simplificações que conduzem a (5.12) são efectuadas pelo programa NEWTOP de um modo muito eficiente. É no entanto necessário assinalar quais são as restrições que se pretende suprimir e em cada uma delas identificar a variável que é explicitada e em seguida substituída.

O algoritmo que no programa NEWTOP procede à pré-substituição de variáveis é capaz de resolver correctamente a seguinte situação

$$x_1 = 2x_3 \quad (5.13a)$$

$$x_2 = 4x_4 \quad (5.13b)$$

$$x_3 = 5x_2 \quad (5.13c)$$

Antes de proceder à substituição de x_1 , x_2 e x_3 no programa matemático é necessário modificar as restrições (5.13) para que as variáveis x_2 e x_3 sejam integralmente substituídas. O programa NEWTOP efectua a transformação de (5.13a-c) no seguinte conjunto de restrições equivalentes

$$x_1 = 40x_4 \quad (5.14a)$$

$$x_2 = 4x_4 \quad (5.14b)$$

$$x_3 = 20x_4 \quad (5.14c)$$

Deste modo as variáveis situadas no primeiro membro (x_1 , x_2 e x_3) deixam de figurar no programa matemático. A variável que se encontra no segundo membro (x_4) permanece na formulação como uma variável independente ou dependente (ver secção 5.2).

O seguinte exemplo corresponde a uma situação em que a substituição atrás referida não pode ser efectuada

$$x_1 = x_3 \quad (5.15a)$$

$$x_2 = x_1 \quad (5.15b)$$

$$x_3 = x_2 \quad (5.15c)$$

Neste caso não é possível transformar (5.15) num conjunto de restrições que possua a particularidade de nenhuma variável figurar simultaneamente num primeiro e num segundo membro. Esta situação é detectada pelo programa NEWTOP, sendo designada substituição circular. Quando ocorre, a execução é interrompida e é assinalada a existência de um erro nos dados.

No programa matemático corresponde ao problema de optimização apresentado na secção 5.1 (ver Listagem 5.1) as quatro últimas restrições igualdade são do tipo (5.7) ou (5.8)

$$B_1 = 1.5B_2 \quad (5.16)$$

$$L = 7 \quad (5.17)$$

$$P_1 = 10 \quad (5.18)$$

$$P_2 = 4 \quad (5.19)$$

A substituição destas restrições na formulação indicada na Listagem 5.1 dá origem ao seguinte programa matemático

Listagem 5.2 - Programa matemático que resulta da substituição de algumas variáveis.

```

### Main title of the nonlinear program
Optimizacao de uma consola com duas seccoes
Min.
    1.5 * L1 * H1 * B2 + L2 * B2 * H2 ;
s.t.i.c.
    Tensao em A:  280 * H1 ^ -2 * B2 ^ -1 - 250 < 0 ;
    Tensao em B:  60 * L2 * B2 ^ -1 * H2 ^ -2 - 250 < 0 ;
    Tensao em C:  74.6667 * H1 ^ -1 * B2 ^ -2 - 250 < 0 ;
    Tensao em D:  24 * L2 * B2 ^ -2 * H2 ^ -1 - 250 < 0 ;
    L1 minimo:   - L1 + 0.1 < 0 ;
    L2 minimo:   - L2 + 0.1 < 0 ;
s.t.e.c.
    Comprimento total:  7 - L1 - L2 = 0 ;
END_OF_FILE

```

As variáveis correspondentes à versão inicial do programa matemático e que se encontram enumeradas em (5.1) podem ser classificadas do seguinte modo

P_1 - a substituir

P_2 - a substituir

L_1 - independente (valor inicial = 3.5)

L_2 - dependente (valor inicial = 3.5)

L - a substituir

B_1 - a substituir

H_1 - independente (valor inicial = 1.5)

B_2 - independente (valor inicial = 0.6)

H_2 - independente (valor inicial = 1.2)

A comparação das Listagens 5.1 e 5.2 permite verificar que é significativamente mais simples preparar um programa matemático com as características da versão original, devido ao facto de existir uma maior homogeneidade das expressões. A supressão de algumas variáveis por intermédio da substituição de restrições igualdade é efectuada de um modo eficiente pelo programa NEWTOP, resultando desta operação um programa matemático que pode ser resolvido com uma maior economia de recursos informáticos. Em problemas que apresentem uma formulação mais complexa e um número mais elevado de variáveis e restrições, as vantagens das técnicas referidas tornam-se mais significativas.

5.4 - DESCRIÇÃO DO PROGRAMA NEWTOP

Nas secções anteriores deste capítulo e nos capítulos anteriores foram apresentadas diversas técnicas utilizadas no programa NEWTOP, os respectivos fundamentos e o núcleo do algoritmo correspondente método de Newton. Nesta secção é apresentada a ordem pela qual as diversas técnicas vão sendo aplicadas durante a resolução de um problema de optimização. São também referidos alguns aspectos relativos à utilização do programa.

A resolução de um problema de optimização com o programa NEWTOP envolve num caso geral a execução das seguintes tarefas:

- A) leitura, interpretação, validação e armazenamento das funções que constituem o programa matemático (ver Listagem 5.2);
- B) leitura, interpretação, validação e armazenamento das restrições simples do tipo (5.7) e (5.8);
- C) substituição das variáveis que figuram no primeiro membro das equações (5.7) e (5.8) na totalidade das funções que constituem o programa matemático;
- D) simplificação das funções que constituem o programa matemático (ver (5.9) a (5.12) na secção 5.3);
- E) leitura da solução inicial;
- F) leitura de parâmetros que influenciam o comportamento do processo iterativo;
- G) cálculo do valor inicial das variáveis dependentes (ver secção 5.2);
- H) modificação das funções que constituem o programa matemático como consequência da transformação das variáveis de projecto (ver secção 4.3);

- I) cálculo dos coeficientes associados às variáveis de desvio (ver secção 4.3);
- J) cálculo das normas dos gradientes das restrições no ponto correspondente à solução inicial (ver secção 4.3);
- K) normalização das restrições (ver secção 4.3);
- L) processo iterativo de acordo com o algoritmo da Figura 3.1 (ver também o Capítulo 4);
- M) apresentação dos resultados.

São em seguida apresentados com mais pormenor alguns aspectos relacionados com as diversas fases correspondentes à resolução de um problema de optimização com o programa NEWTOP.

Se no programa matemático não existirem restrições do tipo (5.7) e (5.8), as fases B), C) e D) não são executadas. Se na fase F) o utilizador não indicar valores para os referidos parâmetros, o programa prossegue com opções que na generalidade dos casos permitem a obtenção de bons resultados. Se não existirem restrições igualdade, também não existem variáveis dependentes e, conseqüentemente, na fase G) não é realizada qualquer operação. Quando existem variáveis dependentes, o utilizador pode optar por não executar a fase G), sendo nesse caso utilizados como solução inicial os valores que foram lidos na fase E). A aplicação das diversas técnicas de *scaling* correspondentes às fases H), I), J) e K) é opcional, sendo no entanto fortemente recomendada a sua utilização.

A informação correspondente à fase F) engloba um elevado número de parâmetros, sendo em seguida apresentados apenas os mais importantes. É também indicado o valor ou opção que deve ser adoptado na generalidade dos casos:

- transformação das variáveis de projecto	SIM	(5.20)
- transformação das variáveis de desvio	SIM	(5.21)
- normalização das restrições	SIM	(5.22)
- valor inicial dos multiplicadores de Lagrange	1.0	(5.23)
- erro tolerado na verificação do critério de convergência (4.10)	10^{-5}	(5.24)
- número máximo de iterações relativas ao método de Newton	50	(5.25)
- parâmetro ε_L (ver secção 4.6)	10^{-2}	(5.26)

- parâmetro ϵ_x (ver secção 4.6)	10^{-4}	(5.27)
- cálculo do erro correspondente à resolução do sistema de equações lineares (4.7)	NÃO	(5.28)
- valor mínimo do parâmetro de pesquisa unidimensional (α_{\min}) (ver secção 4.5)	10^{-3}	(5.29)
- valor máximo do parâmetro de pesquisa unidimensional (α_{\max}) (ver secção 4.5)	1.0	(5.30)
- cálculo do valor inicial das variáveis dependentes	SIM	(5.31)
- método de resolução do sistema de equações lineares (4.7) (directo/iterativo)	DIRECTO	(5.32)

A primeira tentativa de resolução de um problema de optimização deve ser efectuada com as opções indicadas de (5.20) a (5.32). Estas opções só devem ser alteradas se o processo iterativo não for bem sucedido ou se a precisão dos resultados não for satisfatória. Durante a resolução de problemas de optimização com o programa NEWTOP podem ocorrer situações que impedem ou retardam a convergência para a solução óptima. Algumas destas situações podem ser ultrapassadas com simples alterações das opções atrás referidas, sendo assim evitada a necessidade de modificar a solução inicial ou a formulação do problema. Devido à impossibilidade de enumerar todos os casos que requerem uma alteração das opções, apresentam-se apenas alguns exemplos.

Se nas primeiras iterações a solução apresentar grandes oscilações, o parâmetro (5.30) deve ser diminuído. Um aumento significativo do parâmetro (5.23) também implica uma convergência mais lenta, sendo assim diminuída a probabilidade de ocorrerem oscilações. Se o parâmetro de pesquisa unidimensional adoptar valores muito pequenos, deve ser alterada a opção (5.28) para que o erro correspondente à resolução do sistema de equações lineares passe a ser calculado. A eventual ocorrência de valores desse erro demasiado elevados é provocada por um mau condicionamento da matriz Hessiana. Este problema pode ser ultrapassado aumentando o valor dos parâmetros (5.26) e (5.27) (ver secção 4.6) ou modificando o parâmetro (5.32) para que a resolução do sistema de equações (4.7) passe a ser efectuada com o método iterativo (ver secção 4.4.2).

O ficheiro que contém as opções (5.20) a (5.32) é lido em todas as iterações relativas ao método de Newton. Se o programa NEWTOP estiver a ser executado num computador com sistema operativo multi-tarefa (e.g., UNIX), o utilizador pode alterar o ficheiro que contém as opções durante o processo iterativo sem proceder à sua interrupção. Torna-se assim

possível conduzir o processo iterativo com base nas informações que vão sendo disponibilizadas pelo programa NEWTOP. Esta técnica permite que durante o processo iterativo sejam rectificadas opções cujos valores iniciais não tenham sido bem escolhidos e permite uma pronta reacção do utilizador a comportamentos inesperados do algoritmo. Por estes motivos, é fundamental que durante o processo iterativo seja disponibilizada a maior quantidade possível de informação relativa às grandezas que em cada iteração são modificadas pelo algoritmo. A rápida percepção das características desta elevada quantidade de informação só é possível com recurso à visualização gráfica em seguida apresentada.

Visualização do comportamento do processo iterativo

A informação que é possível visualizar durante a execução do programa NEWTOP encontra-se exemplificada na Figura 5.2, que corresponde à resolução do problema de optimização de uma consola com duas secções (ver Figura 5.1). Na metade superior da figura e da esquerda para a direita encontram-se os gráficos com a evolução ao longo do processo iterativo do valor do parâmetro de pesquisa unidimensional (ver secção 4.5), do valor da norma do gradiente do Lagrangeano (ver secção 4.1) e do valor da função objectivo. Para facilitar a visualização dos valores mais pequenos, é utilizada no segundo gráfico uma escala linear-logarítmica. Na metade inferior da Figura 5.2 encontra-se um histograma com os valores das variáveis que estão presentes na fase iterativa, não sendo portanto representados os valores correspondentes às variáveis pré-substituídas (ver secção 5.3). De acordo com a formulação indicada na Listagem 5.2, encontram-se representados da esquerda para a direita os valores de L_1 , L_2 , H_1 , B_2 e H_2 , seguindo-se as seis variáveis de desvio, os seis multiplicadores de Lagrange associados às restrições desigualdade e por último o multiplicador de Lagrange correspondente à única restrição igualdade. Uma vez que foram utilizadas as técnicas de *scaling* referidas na secção 4.3 e os valores representados correspondem às variáveis transformadas, a sua grandeza indica a variação em relação à solução inicial.

Quando o processo iterativo é bem sucedido e não ocorrem oscilações da solução, o parâmetro de pesquisa unidimensional adopta valores próximos da unidade nas primeiras e últimas iterações, sendo os valores correspondentes às restantes iterações significativamente inferiores. A evolução da norma do gradiente do Lagrangeano apresenta na generalidade dos casos características semelhantes às indicadas na Figura 5.2, pelos motivos referidos na secção 4.6. Nas iterações intermédias é frequente a função objectivo apresentar valores inferiores ao correspondente à solução óptima. Verifica-se no entanto que essas soluções não são admissíveis e, conseqüentemente, não podem ser consideradas como solução do problema de optimização.

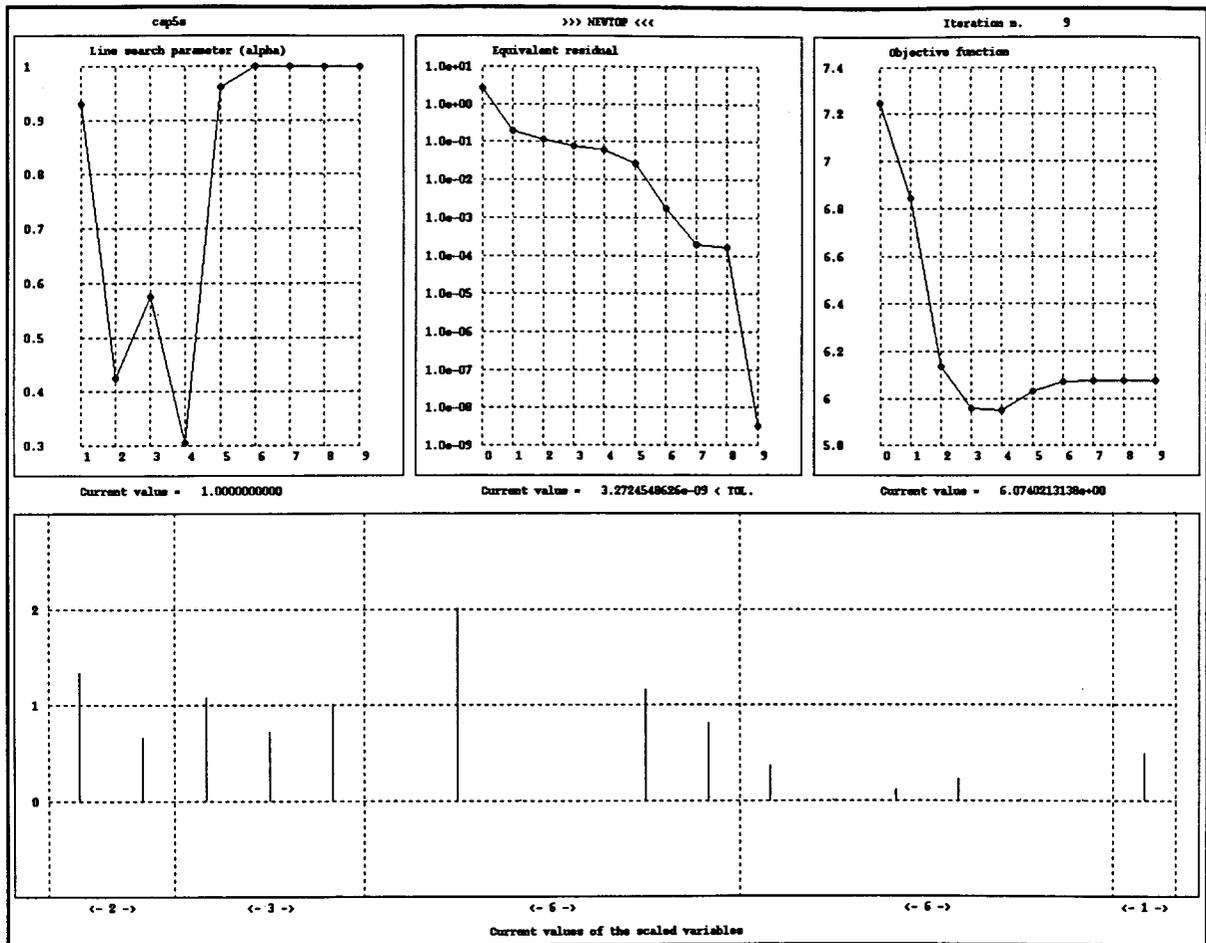


Figura 5.2 - Consola com duas secções - visualização da evolução de alguns parâmetros e histograma com a solução final.

A representação gráfica indicada na Figura 5.2 pode ser efectuada durante o processo iterativo ou após a sua conclusão. No primeiro caso os três gráficos apresentam a evolução dos parâmetros até à situação corrente e no histograma é representada a solução disponível após a última iteração efectuada. Se o processo iterativo estiver já concluído, é possível visualizar qualquer solução intermédia, bem como a evolução dos três parâmetros até à iteração seleccionada.

Apresentação dos resultados

O programa NEWTOP foi desenvolvido com o objectivo de poder ser utilizado na resolução de qualquer problema de optimização, que possa ser formulado como um programa matemático constituído por funções reais e contínuas. Por este motivo, os resultados

fornechos pelo programa NEWTOP devem ser apresentados com um formato independente de qualquer problema específico, sem contudo ser comprometida a sua fácil interpretação. Este compromisso foi conseguido com o formato que se encontra exemplificado na Listagem 5.3. Os resultados indicados correspondem à solução do problema de optimização de uma consola com duas secções (ver Figura 5.1). Na generalidade dos casos os resultados são constituídos por alguns parâmetros gerais, seguidos dos valores das variáveis de projecto, das folgas correspondentes às restrições desigualdade e dos valores dos multiplicadores de Lagrange. Todos estes resultados correspondem ao programa matemático original, i.e., os valores que foram transformados devido à aplicação das técnicas de *scaling* são sujeitos a uma transformação inversa de acordo com o que foi indicado na secção 4.3. Deste modo, as variáveis de projecto encontram-se expressas nas unidades correspondentes ao respectivo significado físico, as folgas encontram-se expressas nas unidades correspondentes à restrição desigualdade original (ver Listagem 5.2) e os multiplicadores de Lagrange encontram-se expressos numas unidades que correspondem ao quociente entre as unidades da função objectivo e as unidades da respectiva restrição (ver secção 2.2.5). Na Listagem 5.3 encontram-se também os valores das variáveis P_1 , P_2 , L e B_1 , que tinham sido substituídas de acordo com (5.16) a (5.19). Para facilitar a interpretação dos resultados, os nomes das variáveis são associados aos respectivos valores e os títulos das restrições são associados aos valores das respectivas folgas e multiplicadores de Lagrange. A análise dos resultados apresentados na Listagem 5.3 permite concluir que a solução óptima do problema indicado na Figura 5.1 é fundamentalmente caracterizada pelo facto de as restrições de tensão nos pontos A , C e D se encontrarem activas, enquanto que no ponto B existe uma folga significativa.

Após a resolução do problema de optimização é possível representar num único gráfico a evolução dos valores das variáveis durante o processo iterativo. Na Figura 5.3 encontra-se representada a evolução das variáveis independentes e dependentes relativas ao problema de optimização da consola com duas secções (ver Figura 5.1). Neste exemplo simples pode ser facilmente verificado que em todas as iterações a solução corrente verifica a única restrição igualdade ($L_1 + L_2 = 7$).

Listagem 5.3 - Consola com duas secções - solução óptima obtida com o programa NEWTOP.

```

*** Parametros gerais:
Numero de iteracoes -----> 9
Valor da funcao objectivo -----> 6.07402131e+00
Erro -----> 3.27245486e-09

*** Variaveis de projecto:
1 1 10.000000000000000000000000 -> P1
2 2 4.000000000000000000000000 -> P2
3 3 4.66666667187260220000 -> L1
4 4 2.33333332812739780000 -> L2
5 5 7.000000000000000000000000 -> L
6 6 0.64537145685348940000 -> B1
7 7 1.61342864659555190000 -> H1
8 8 0.43024763790232623000 -> B2
9 9 1.21007148496578170000 -> H2

*** Folgas nas restricoes desigualdade:
10 1 0.000000000000000000043 -> Tensao em A
11 2 27.77777790200220000000 -> Tensao em B
12 3 0.000000000000000000112 -> Tensao em C
13 4 0.000000000000000000056 -> Tensao em D
14 5 4.56666667187259900000 -> L1 minimo
15 6 2.23333332812739860000 -> L2 minimo

*** Multiplicadores de Lagrange das restricoes desigualdade:
16 1 0.00809869507110470500 -> Tensao em A
17 2 0.00000000000175988805 -> Tensao em B
18 3 0.00323947803280744090 -> Tensao em C
19 4 0.00485921702699293730 -> Tensao em D
20 5 0.00000000013703880206 -> L1 minimo
21 6 0.00000000013703880206 -> L2 minimo

*** Multiplicadores de Lagrange das restricoes igualdade:
22 1 1.04126079618194360000 -> Comprimento total
    
```

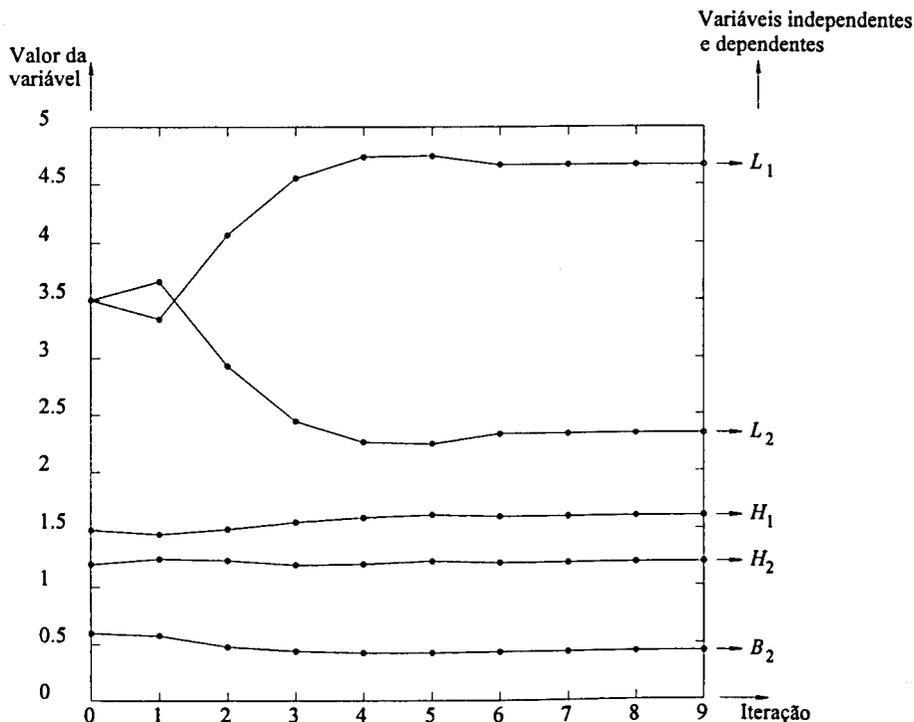


Figura 5.3 - Consola com duas secções - evolução dos valores das variáveis.

5.5 - SOLUÇÃO INICIAL

O sucesso da resolução de um problema de optimização com o programa NEWTOP depende significativamente da solução inicial (ver secção 3.2.3). Para que a probabilidade de sucesso seja elevada, a solução inicial deve situar-se tão próximo quanto possível da solução final. Na generalidade dos casos é vantajoso que a solução inicial seja admissível e que as restrições desigualdade apresentem uma folga significativa. Para se obter uma solução inicial com estas características, é fundamental efectuar um pré-dimensionamento de todas as grandezas de acordo com o respectivo significado físico. Sempre que possível, deve-se recorrer à utilização prévia de um método de optimização mais rudimentar, que não seja tão dependente da qualidade da solução inicial (e.g., método das pesquisas aleatórias). Em problemas de optimização estrutural é habitual considerar como solução inicial uma estrutura ligeiramente sobredimensionada.

Para exemplificar uma situação de não convergência que ocorre em alguns problemas, apresenta-se o seguinte programa matemático

Min.

$$2x_1 + x_2 \quad (5.33a)$$

s.a

$$0.75x_1^4 - 6x_1^3 + 16x_1^2 - 17x_1 - x_2 + 9 \leq 0 \quad (5.33b)$$

$$x_1 + 3x_2 - 9 \leq 0 \quad (5.33c)$$

Uma vez que (5.33) só depende de duas variáveis, é possível efectuar a representação gráfica da função objectivo e da região admissível (ver Figura 5.4). Quando o programa NEWTOP é utilizado com o ponto A (3.2.1.2) como solução inicial, o processo iterativo converge para o ponto B (3.00, 0.75), que corresponde à solução óptima x^* . Quando a solução inicial é o ponto C (1,2), as sucessivas soluções vão-se aproximando do ponto D . Como nas proximidades deste ponto não existe nenhuma solução admissível, o processo iterativo deixa de apresentar progressões significativas, sem que tenha sido alcançada uma solução que apresente um erro suficientemente pequeno. Esta situação pode ser visualizada por intermédio dos gráficos representados na Figura 5.5 (ver também a secção 5.4), sendo caracterizada pelo facto de o parâmetro de pesquisa unidimensional apresentar valores cada vez mais pequenos e de o erro estabilizar com um valor significativo. Como nestas circunstâncias a solução quase não varia, o mesmo ocorre obviamente com a função objectivo.

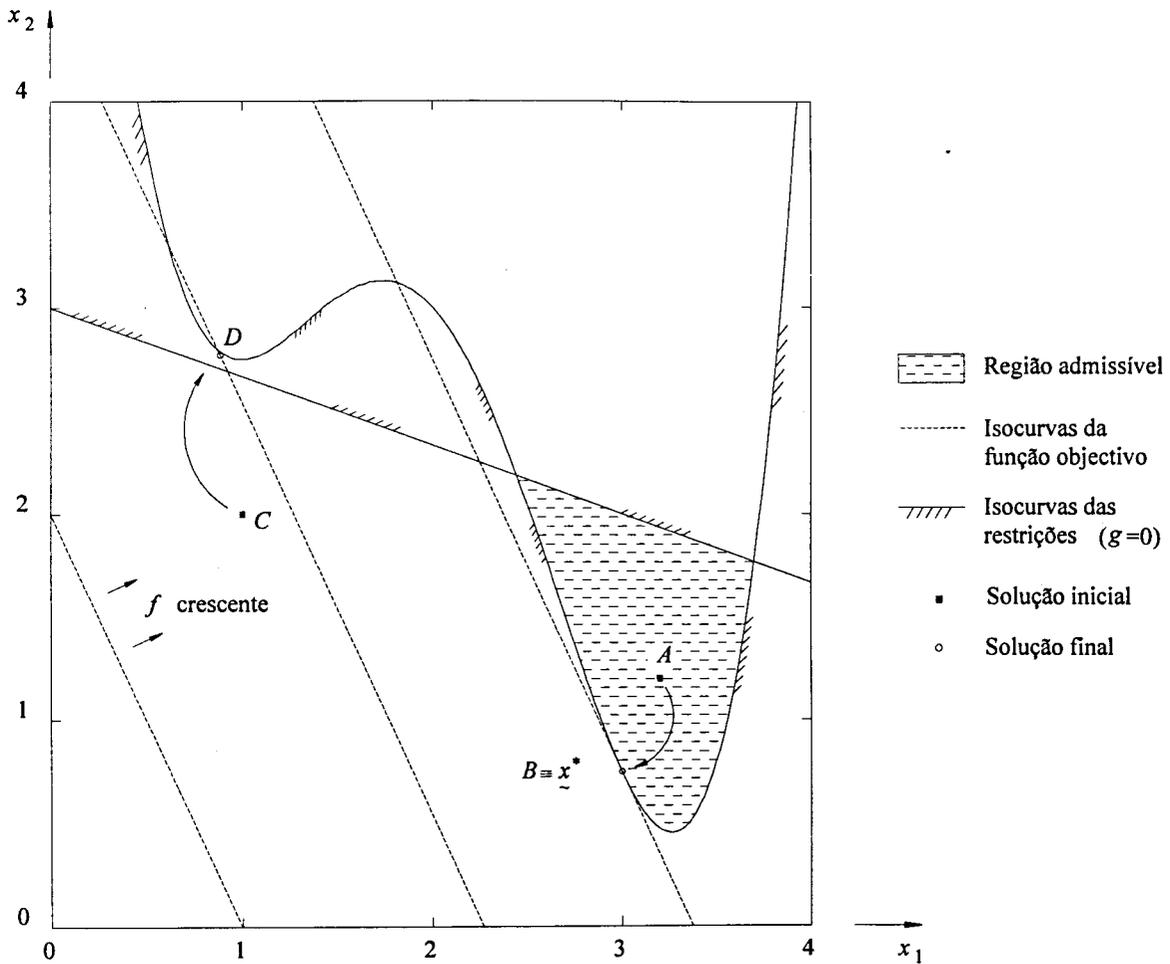


Figura 5.4 - Representação gráfica do programa matemático (5.33).

Com este caso simples pretende-se exemplificar o comportamento do processo iterativo, quando a solução é atraída para uma região em que o erro é pequeno mas não existe nenhuma solução admissível. Nestes casos verifica-se que o algoritmo não consegue fornecer nenhuma solução satisfatória, sendo necessário recorrer a uma outra solução inicial. O comportamento exemplificado na Figura 5.5 ocorre por vezes em problemas com um número mais elevado de variáveis e restrições, sendo nesses casos impossível visualizar graficamente as características da função objectivo e das restrições. Para ultrapassar esta situação é também necessário recorrer a uma solução inicial alternativa.

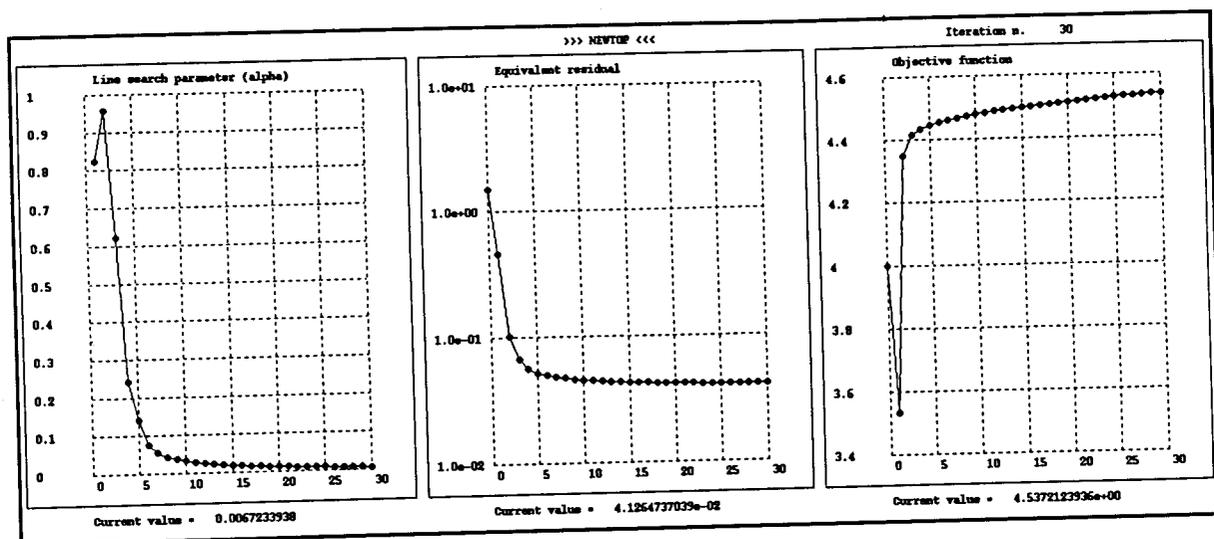


Figura 5.5 - Programa matemático (5.33) - evolução de alguns parâmetros quando não ocorre convergência.

Existem outras situações de insucesso na utilização do programa NEWTOP que podem ser facilmente diagnosticadas com o auxílio dos gráficos e histogramas exemplificados na Figura 5.2. Referem-se os casos de divergência (i.e., erro crescente), os casos em que na solução final existem restrições com multiplicador de Lagrange (λ_j^g) negativo e os casos em que a convergência se torna lenta devido ao facto de algumas variáveis apresentarem valores fora do respectivo significado físico (e.g., áreas negativas). Quando se obtém uma solução final que não é um mínimo local (e.g., máximo local), é em geral possível diagnosticar a situação, porque na solução é muito provável existirem algumas componentes de λ_j^g negativas. Nestes casos é também frequente a solução final apresentar um valor da função objectivo superior ao de outras soluções admissíveis conhecidas. Por último refere-se o caso em que a solução final verifica as condições necessárias de optimalidade de Karush-Kuhn-Tucker (ver Capítulo 2), mas não corresponde ao mínimo global, sendo portanto um mínimo local. Uma vez que a generalidade dos problemas de optimização não são convexos, não é conhecido nenhum processo de diagnosticar ou evitar a convergência para um mínimo local.

Para que os diversos tipos de insucesso atrás referidos sejam evitados, é fundamental que o problema de optimização se encontre correctamente formulado, que a solução inicial tenha sido devidamente pré-dimensionada e que os parâmetros (5.20) a (5.32) apresentem valores adequados. Nas situações em que se verificar ser difícil obter uma solução satisfatória devem ser efectuadas ligeiras alterações nos parâmetros (5.20) a (5.32), devem ser ensaiadas diversas soluções iniciais com características distintas e como último recurso devem ser

utilizados *move limits* (ver secção 3.3.1) e *side constraints* adicionais (ver secção 2.1.4). Quando é necessário preparar uma nova solução inicial, é em geral vantajoso recorrer a uma das soluções intermédias correspondentes a anteriores execuções do programa, mesmo que tenham sido mal sucedidas. Antes de utilizar esta nova solução inicial é conveniente verificar se todas as variáveis apresentam valores compatíveis com o respectivo significado físico. As soluções iniciais demasiado afastadas da região admissível devem ser sempre evitadas.

Nesta secção foram apresentadas diversas situações de insucesso na aplicação do programa NEWTOP, foi referido o modo de as diagnosticar e foram sugeridas algumas técnicas para as ultrapassar. A resolução de inúmeros problemas de optimização com características distintas permitiu, no entanto, concluir que, sempre que são seguidas as indicações aqui sugeridas, a solução óptima pode ser facilmente obtida.

5.6 - COMPARAÇÃO COM OS RESULTADOS DE OUTROS AUTORES

Para validar os resultados fornecidos pelo programa NEWTOP, foi efectuada a sua aplicação a problemas cuja resolução com outros programas se encontra publicada. Com esse objectivo foi seleccionado um problema de optimização de uma mola, cuja formulação pormenorizada se encontra em [Aro89]. O programa matemático correspondente a este problema é o seguinte

Min.

$$f = D d^2 N + 2 D d^2 \quad (5.34a)$$

s.a

$$g_1 = 71875d^4 - D^3N \leq 0 \quad (5.34b)$$

$$g_2 = 4D^2 + 1.46Dd - 2.46d^2 - 12566Dd^3 + 12566d^4 \leq 0 \quad (5.34c)$$

$$g_3 = D^2N - 140.45d \leq 0 \quad (5.34d)$$

$$g_4 = D + d - 1.5 \leq 0 \quad (5.34e)$$

$$g_5 = -d + 0.05 \leq 0 \quad (5.34f)$$

$$g_6 = d - 0.20 \leq 0 \quad (5.34g)$$

$$g_7 = -D + 0.25 \leq 0 \quad (5.34h)$$

$$g_8 = D - 1.30 \leq 0 \quad (5.34i)$$

$$g_9 = -N + 2 \leq 0 \quad (5.34j)$$

$$g_{10} = N - 15 \leq 0 \quad (5.34k)$$

Em (5.34) as três variáveis (D, d, N) estão sujeitas a dez restrições desigualdade, cujas expressões são equivalentes às indicadas em [Aro89]. Para que o programa NEWTOP pudesse ser aplicado, foi necessário converter as restrições originais em expressões do tipo (4.11). Na resolução com o programa NEWTOP foi considerada a solução inicial $(D, d, N)^0 = (1.0, 0.1, 14)$. Esta solução é admissível e implica a existência de uma folga significativa em todas as restrições. No Quadro 5.1 é apresentada a comparação entre os resultados obtidos com o programa NEWTOP e com um programa baseado na programação quadrática sequencial [Aro89].

Quadro 5.1 - Comparação entre os resultados obtidos com o programa NEWTOP e os que se encontram publicados em [Aro89].

	NEWTOP	[Aro89]
f	0.012 678 676	0.012 678 68
D	0.356 883 330	0.356 950
d	0.051 695 909	0.051 698 7
N	11.293 376	11.289 5
Erro*	9×10^{-10}	10^{-3}

* O erro não tem o mesmo significado nos dois métodos.

A comparação dos resultados apresentados no Quadro 5.1 permite concluir que os dois programas fornecem valores praticamente coincidentes para as principais grandezas que caracterizam a solução. Devido ao reduzido valor do erro e ao número de algarismos significativos correctos que nessas circunstâncias se obtêm, presume-se que os resultados correspondentes ao programa NEWTOP são mais rigorosos do que os publicados em [Aro89].

Em [Bel82] são apresentados os resultados de um estudo comparativo de diversos programas de optimização. Um dos exemplos utilizados consiste também na optimização de uma mola, sendo o respectivo programa matemático equivalente a (5.34). De acordo com os resultados publicados em [Bel82], os programas baseados no método das direcções admissíveis e no método do gradiente projectado não forneceram qualquer solução. Os programas baseados na minimização sem restrições sequencial (SUMT) e na programação quadrática recursiva apresentaram soluções com um custo cerca de 20% superior ao indicado no Quadro 5.1. Os resultados que mais se aproximam da melhor solução conhecida foram fornecidos por três programas baseados no método do Lagrangeano aumentado. Apesar de serem os mais rigorosos, esses resultados correspondem a um mínimo local e apresentam diferenças no segundo algarismo significativo, quando comparados com a solução indicada no Quadro 5.1.

Em [Van84] são apresentados os resultados de um estudo comparativo da utilização de diversos métodos na optimização de uma consola com cinco secções. Os diversos programas utilizados baseiam-se na minimização sem restrições sequencial (SUMT), programação linear sequencial, programação quadrática sequencial, direcções admissíveis, gradiente reduzido generalizado e no Lagrangeano aumentado. Em [Aze89] são apresentados resultados que mostram que todos estes métodos convergem para um mínimo local, enquanto que o programa NEWTOP fornece uma solução mais lógica e com um custo cerca de 4% inferior.

5.7 - OBSERVAÇÕES FINAIS

Neste capítulo foi apresentado um programa de computador destinado a resolver uma grande diversidade de problemas de optimização. Com este objectivo, foram nele incluídas diversas técnicas destinadas a permitir uma rápida e simples abordagem de novos tipos de problemas. Recorrendo à formulação e resolução de alguns problemas elementares, foi possível verificar que além de apresentar uma grande versatilidade e facilidade de utilização o programa NEWTOP é capaz de calcular soluções muito rigorosas.

No Capítulo 6 serão apresentadas algumas aplicações do programa NEWTOP à resolução de problemas de optimização de estruturas com um elevado número de variáveis e restrições. Com base nestes exemplos será quantificada uma outra característica importante de um programa de optimização, que consiste na sua eficiência.

CAPÍTULO 6

OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO LINEAR

O conceito de projecto de estruturas está habitualmente associado a uma pesquisa da solução mais económica que satisfaz um conjunto de requisitos predefinidos. A solução final a adoptar deve apresentar níveis adequados de segurança, funcionalidade, qualidade e facilidade de execução. De um modo geral estas exigências encontram-se regulamentadas ou são definidas caso a caso. A pesquisa de uma solução satisfatória pode ser efectuada de diversas formas, desde a realização de um reduzido número de tentativas orientadas pela intuição do projectista, até à utilização de um método que forneça a solução óptima de um modo quase integralmente automatizado.

Na Figura 6.1 é apresentado o algoritmo que é utilizado na quase totalidade dos casos de projecto de estruturas [Her93]. Este algoritmo é comum aos casos em que são utilizados métodos puramente heurísticos e à generalidade dos casos em que se recorre a técnicas de optimização.

Na fase inicial de formulação do problema está geralmente incluída a selecção do tipo de estrutura, dos materiais a utilizar e das restrições que condicionam a solução final. É também nesta fase que são escolhidos os parâmetros que poderão ser modificados no caso de a solução não ser satisfatória. A solução inicial pode ser arbitrada, sendo no entanto mais aconselhável partir de uma solução que respeite as restrições e que se encontre tão próximo quanto possível da solução final. Nesta fase, é fundamental a experiência e intuição do projectista. Em certos casos pode-se justificar o recurso à utilização de um método iterativo rudimentar apenas com o objectivo de se obter uma solução inicial com as características atrás referidas.

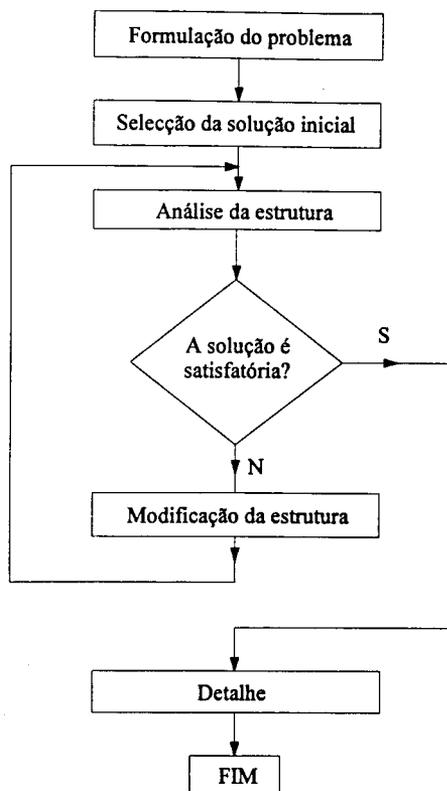


Figura 6.1 - Algoritmo tradicionalmente utilizado no projecto de estruturas.

No interior do ciclo iterativo indicado na Figura 6.1 encontra-se a análise da estrutura corrente. Uma vez que este cálculo é efectuado diversas vezes, pode-se justificar o recurso a técnicas de reanálise de estruturas [Aze84] [Kir81]. Os resultados obtidos com o método de análise/reanálise utilizado devem ser suficientemente rigorosos para que a apreciação da solução e o conjunto de modificações introduzidas na estrutura não sejam afectadas por essa falta de precisão. O critério em que se baseia a decisão de terminar ou não o ciclo iterativo pode reflectir vários níveis de exigência de precisão. Na generalidade dos casos é suficiente que a estrutura respeite os requisitos pré-estabelecidos e não esteja excessivamente sobredimensionada. Se forem utilizadas técnicas de optimização, a solução só é aceite se for admissível e satisfizer determinadas condições de optimalidade, que impliquem ser essa a mais vantajosa de todas as soluções admissíveis. Em certos casos, o ciclo iterativo é interrompido quando em duas iterações consecutivas não existe uma alteração significativa da solução.

No algoritmo de projecto de estruturas apresentado na Figura 6.1 a fase em que se procede à modificação da estrutura implica a tomada de decisões baseadas em critérios mais ou menos elaborados. Os diferentes métodos associados a esta fase do projecto de uma estrutura podem ser classificados em três grupos [Kir81] [Aro89] [Her93]:

- regras heurísticas;
- critérios de optimalidade (métodos indirectos);
- programação matemática (métodos directos).

A qualidade das soluções obtidas a partir de sucessivas modificações da estrutura com base em regras heurísticas depende essencialmente da intuição e da experiência do projectista. Estas regras dependem do tipo de problema que está a ser estudado e do tipo de estrutura utilizada. Na generalidade das situações apenas se procura redimensionar por tentativas as zonas da estrutura que se encontrem subdimensionadas ou significativamente sobredimensionadas. A aplicação de regras heurísticas não pode portanto ser considerada como uma técnica de optimização.

Os métodos baseados em critérios de optimalidade fundamentam-se na pesquisa de uma solução que satisfaça um conjunto de condições dependentes do significado físico do problema. É pressuposto que a solução estrutural que satisfaz essas condições corresponde à solução óptima do problema. A pesquisa desta solução é efectuada recorrendo ao algoritmo da Figura 6.1, sendo na fase de modificação da estrutura utilizado um conjunto de relações que conduzem a uma solução mais satisfatória. Estas relações podem ser deduzidas a partir das condições de Karush-Kuhn-Tucker ou com base em regras heurísticas, que conduzem a uma solução em que todos os pontos críticos da estrutura se encontram no limiar do colapso (*Fully Stressed Design*) [Ber87]. Os métodos baseados em critérios de optimalidade designam-se métodos indirectos, porque não têm como finalidade prioritária a minimização da função objectivo.

Os métodos directos são aqueles que se baseiam na programação matemática e têm como objectivo a determinação da solução que respeita as restrições e torna mínimo o valor da função objectivo. No algoritmo da Figura 6.1 a fase de modificação da estrutura engloba os seguintes passos:

- A - cálculo das derivadas da função objectivo e restrições em ordem às variáveis de projecto (análise de sensibilidades);
- B - cálculo no espaço das variáveis de projecto de uma direcção que conduza a uma diminuição da função objectivo;
- C - cálculo da grandeza do passo a dar nessa direcção de modo que a diminuição da função objectivo seja o mais acentuada possível (pesquisa unidimensional).

Na generalidade dos métodos baseados na programação matemática é necessário, ou pelo menos desejável, que as sucessivas soluções sejam admissíveis. No cálculo da direcção referida no passo B pode ser utilizado um dos métodos apresentados no Capítulo 3, que seja aplicável à minimização de funções com diversas variáveis e com restrições. No passo C pode-se recorrer a um dos métodos de minimização de funções com uma variável também referidos no Capítulo 3. Nos casos em que não for possível obter uma solução satisfatória para o problema de optimização, deve-se repetir o processo com outra solução inicial.

O algoritmo da Figura 6.1 e os métodos associados a cada uma das suas fases pressupõem que as variáveis relativas ao comportamento da estrutura (e.g., deslocamentos, forças, tensões) não são variáveis de projecto, sendo portanto consideradas de um modo implícito. No presente trabalho foi utilizada uma abordagem distinta do problema de optimização estrutural, baseada na inclusão das variáveis relativas ao comportamento da estrutura no conjunto das variáveis de projecto [Haf85] [Bar89]. Por este motivo esta formulação é designada análise e dimensionamento simultâneos ou formulação integrada, encontrando-se o respectivo algoritmo esquematizado na Figura 6.2.

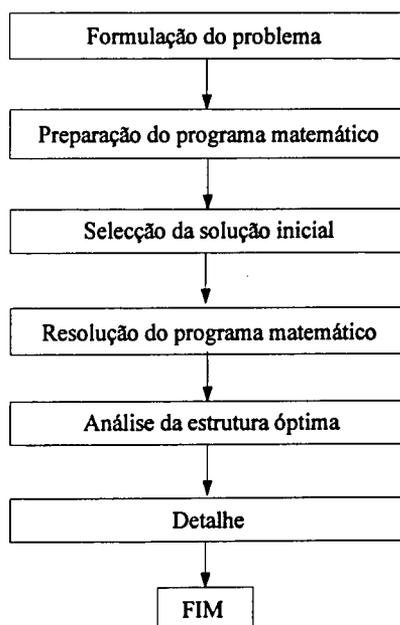


Figura 6.2 - Optimização de estruturas recorrendo à formulação integrada.

No algoritmo da Figura 6.2 cada fase é realizada uma só vez, exceptuando-se os casos em que não é possível resolver o programa matemático com a solução inicial especificada. Nestes casos é necessário seleccionar outra solução inicial e repetir a fase de resolução do

programa matemático. A análise da estrutura correspondente à solução óptima destina-se ao cálculo de certas grandezas que são necessárias na fase de detalhe e que não foram envolvidas no processo de optimização (e.g., reacções em apoios, esforços/tensões em pontos da estrutura em que essas grandezas não tenham sido restringidas). Na fase de análise da estrutura óptima, pode ser utilizado um programa convencional que disponha de pós-processamento gráfico dos resultados. Deste modo evita-se a necessidade de dotar com estas capacidades o programa de optimização de estruturas.

As vantagens e desvantagens de cada uma das formulações referidas serão mais adiante enumeradas.

6.1 - CRITÉRIOS DE OPTIMALIDADE

A optimização de estruturas baseada em critérios de optimalidade teve a sua origem na ideia intuitiva de que uma estrutura possui um peso mínimo quando nenhum dos seus elementos puder ser diminuído sem que a tensão admissível seja excedida em algum ponto da estrutura. Esta afirmação é sempre verdadeira no caso de estruturas isostáticas sujeitas a um único carregamento, tendo sido demonstrada por Mitchell no início do século XX [Kir81]. O projecto de estruturas baseado neste critério de optimalidade é designado *Fully Stressed Design* (FSD). Ao generalizar este método de optimização aos casos de estruturas hiperstáticas, sujeitas a vários carregamentos e com restrições de tensão máxima e dimensões mínimas, deixa de ser possível garantir a obtenção da solução óptima. Quando são consideradas restrições de deslocamento, de frequência própria e de outros tipos, o método de optimização baseado em critérios de optimalidade passa a ser deduzido a partir das condições de Karush-Kuhn-Tucker e deixa de ser designado *Fully Stressed Design* [Ber87].

6.1.1 - *Fully Stressed Design*

O método de optimização de estruturas baseado na procura do *Fully Stressed Design* (FSD) é particularmente adequado à minimização do peso de estruturas reticuladas, porque a tensão crítica instalada em cada uma das barras pode ser aproximadamente relacionada com a respectiva variável de projecto. No caso das estruturas articuladas, a referida tensão crítica é a tensão normal máxima instalada na barra e a variável de projecto é a área da respectiva secção

transversal. É possível generalizar o conceito de *Fully Stressed Design* às vigas contínuas desde que se considere como tensão crítica a tensão normal máxima na secção em que ocorre o momento flector máximo e como variável de projecto o módulo de flexão. Estas considerações são ainda extensíveis a alguns casos particulares relativos a meios contínuos. Refere-se como exemplo a minimização do peso de uma estrutura laminar sujeita a um estado plano de tensão e discretizada com elementos finitos de espessura constante. A cada elemento finito corresponde uma variável de projecto, que consiste na respectiva espessura. O *Fully Stressed Design* é uma solução tal que qualquer diminuição da espessura de um elemento implicaria a violação de pelo menos uma restrição.

Todos estes exemplos possuem como característica comum a seguinte relação de redimensionamento das barras/elementos, que se baseia no facto de a tensão crítica instalada ser inversamente proporcional ao valor da variável de projecto.

$$x^{q+1} = x^q \frac{\sigma_{crit}}{\sigma_{max}} \quad (6.1)$$

No algoritmo da Figura 6.1 a fase de modificação da estrutura pode basear-se na relação (6.1), sendo x^q o valor corrente de cada variável de projecto e x^{q+1} o valor que daria origem a uma tensão instalada igual a σ_{max} . Apesar de nas estruturas hiperstáticas esta relação não ser exacta, é de esperar que a sua utilização sucessiva conduza a uma convergência para o *Fully Stressed Design*.

De aqui em diante será apenas referido o caso da minimização do peso de treliças. Os outros exemplos atrás mencionados seriam tratados de um modo semelhante. A consideração de diversos casos de carga e de restrições de área mínima implica uma redefinição do conceito de *Fully Stressed Design*. Assim, passa a considerar-se que uma treliça apresenta um peso mínimo quando em cada barra está instalada uma tensão igual à tensão máxima em pelo menos um dos casos de carga, com excepção das barras que apresentam área mínima. No caso das treliças hiperstáticas esta afirmação pode não ser verdadeira, porque é ignorada a influência das variações das áreas das barras na distribuição das forças interiores [Ber87]. Mais adiante será apresentado um exemplo simples em que a solução FSD não é a solução óptima.

Se na optimização de uma treliça forem consideradas restrições de área mínima e se existir mais do que um caso de carga, o redimensionamento correspondente à aplicação da relação (6.1) pode ser efectuado com o algoritmo apresentado na Figura 6.3. Este algoritmo assegura que após a primeira iteração nenhuma barra possui uma área inferior à mínima. Em cada barra, a nova área é imposta pelo caso de carga que lhe provoca um maior incremento em

relação ao valor mínimo. Se em nenhum caso de carga for necessário que a barra possua uma área de valor superior ao mínimo, este valor é-lhe atribuído.

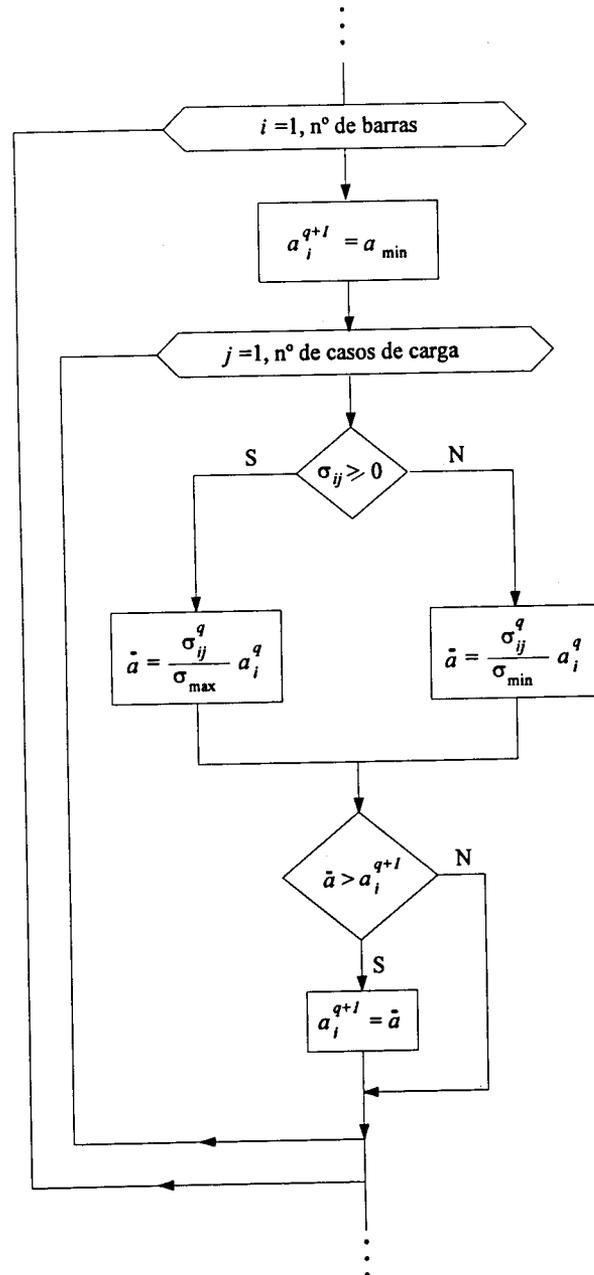
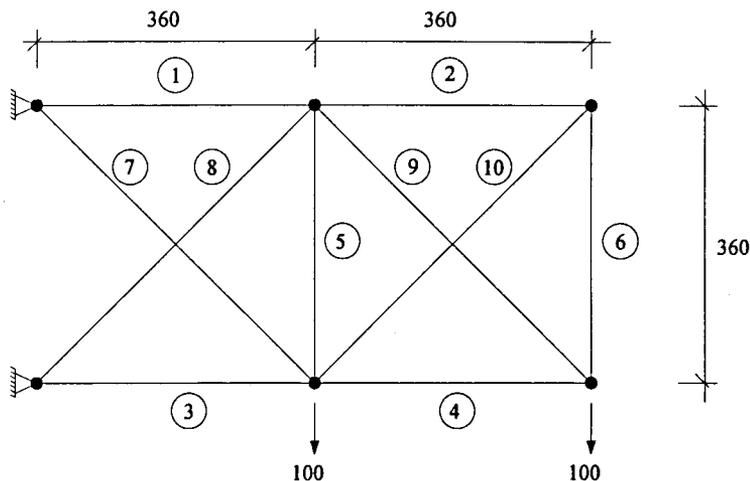


Figura 6.3 - Algoritmo de redimensionamento das áreas das barras de uma treliça com vista à obtenção do *Fully Stressed Design*.

6.1.2 - Optimização de uma treliça com 10 barras

O algoritmo da Figura 6.1 com o redimensionamento indicado na Figura 6.3 foi utilizado no cálculo do *Fully Stressed Design* relativo a uma treliça hiperstática com dez barras e um caso de carga (ver Figura 6.4) [Kir81].



Nota: não é considerado o peso próprio.

Figura 6.4 - Treliça com dez barras - geometria e acção exterior.

Pretende-se minimizar o volume da treliça indicada na Figura 6.4 supondo as áreas das barras como variáveis de projecto. Devem ser consideradas as seguintes restrições desigualdade

$$a_i \geq 0.1 \quad (i = 1, \dots, 10) \quad (6.2)$$

$$-25 \leq \sigma_i \leq 25 \quad (i = 1, \dots, 8, 10) \quad (6.3)$$

$$-50 \leq \sigma_i \leq 50 \quad (i = 9) \quad (6.4)$$

No Quadro 6.1 é efectuada a comparação entre a solução FSD e a solução óptima. Esta última foi obtida com o algoritmo indicado na Figura 6.2, que recorre à formulação integrada. A resolução do programa matemático foi efectuada com o programa NEWTOP (ver Capítulo 5). A optimização de estruturas recorrendo à formulação integrada e ao programa NEWTOP será apresentada mais adiante neste capítulo.

Quadro 6.1 - Treliça com dez barras - comparação da solução FSD com a solução óptima.

Barra	Solução FSD	Solução óptima
1	4.106066	7.900000
2	3.893934	0.100000
3	11.893934	8.100000
4	0.106066	3.900000
5	0.100000	0.100000
6	3.893934	0.100000
7	11.163708	5.798276
8	0.150000	5.515433
9	0.100000	3.676955
10	5.506854	0.141421
Função objectivo:	17252.360	14976.000

Neste exemplo simples a solução FSD apresenta características muito distintas da solução óptima. Se se repetir o cálculo destas soluções com um valor bastante inferior da área mínima, verifica-se que as barras que o algoritmo tenta eliminar são as que na Figura 6.5 estão representadas a tracejado.

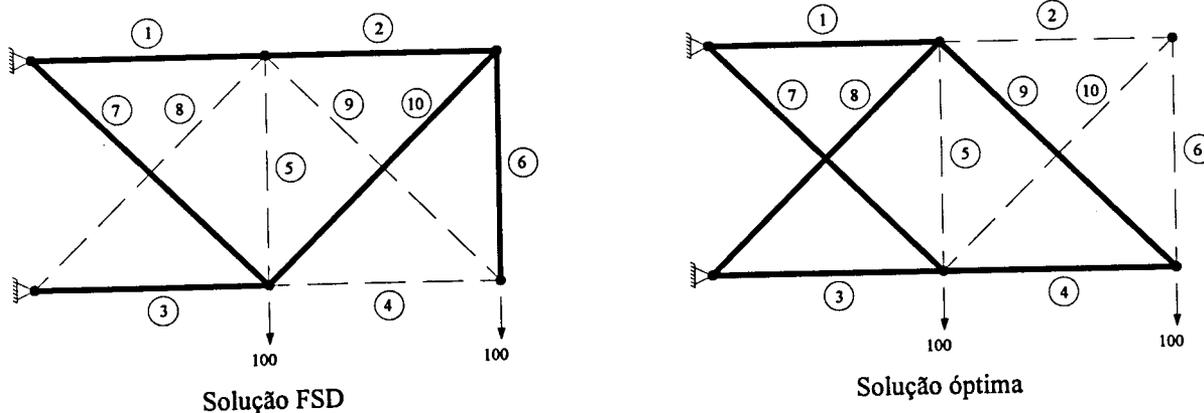


Figura 6.5 - Treliça com dez barras - as barras que cada um dos algoritmos tenta eliminar encontram-se representadas a tracejado.

A solução FSD apresenta, além de uma topologia distinta, um valor final da função objectivo 15% superior ao correspondente à solução óptima. Em exemplos mais complexos, que apresentem barras com módulo de elasticidade e peso específico distintos e múltiplos casos de carga, a probabilidade de o *Fully Stressed Design* ser significativamente distinto da solução óptima aumenta consideravelmente. Com o objectivo de avaliar a fiabilidade dos dois

métodos, foi fornecida como solução inicial do algoritmo FSD uma solução próxima da óptima e como solução inicial do algoritmo baseado no método de Newton uma solução parecida com o FSD. Verificou-se que o algoritmo FSD voltou a apresentar uma convergência para o mínimo local, apesar de partir da solução óptima. Com o programa NEWTOP foi possível chegar à solução óptima partindo da solução FSD, tendo sido no entanto necessário realizar um número de iterações superior ao habitual.

6.1.3 - Restrições de deslocamento

Os métodos baseados na pesquisa do *Fully Stressed Design* não são aplicáveis quando estão presentes restrições de deslocamento. Nestes casos o critério de optimalidade tem de ser deduzido a partir das condições de Karush-Kuhn-Tucker e apresenta-se particularmente simplificado quando existe uma única restrição de deslocamento do tipo

$$g = d - d_{\max} \leq 0 \quad (6.5)$$

ou

$$g = d_{\min} - d \leq 0 \quad (6.6)$$

A utilização de (6.5) ou (6.6) depende do sinal do deslocamento, quando a acção exterior actua sobre a estrutura óptima. Uma vez que o deslocamento restringido é na generalidade dos casos o deslocamento máximo em valor absoluto, o respectivo sinal é à partida conhecido, mesmo que a solução inicial se encontre muito distante da solução óptima. De aqui em diante apenas será referida a restrição (6.5), que é a que deve ser utilizada quando o deslocamento limitado é positivo. As considerações relativas à restrição (6.6) seriam semelhantes.

Devido ao facto de só existir uma restrição desigualdade, as condições de Karush-Kuhn-Tucker (ver Capítulo 2) apresentam a seguinte forma simplificada

$$\frac{\partial f}{\partial x_i} + \lambda \frac{\partial g}{\partial x_i} = 0 \quad (i = 1, \dots, n) \quad (6.7)$$

$$g \leq 0 \quad (6.8)$$

$$\lambda \geq 0 \quad (6.9)$$

$$g \lambda = 0 \quad (6.10)$$

Uma vez que a solução do problema de optimização é condicionada por uma única restrição, admite-se que ela se encontra activa. Nas condições de optimalidade (6.7) a (6.10) pode então considerar-se

$$g = 0 \quad (6.11)$$

$$\lambda > 0 \quad (6.12)$$

A expressão (6.7) pode tomar a seguinte forma

$$-\lambda \frac{\partial g / \partial x_i}{\partial f / \partial x_i} = 1 \quad (i = 1, \dots, n) \quad (6.13)$$

Designando o primeiro membro de (6.13) por D_i , o critério de optimalidade apresenta uma forma ainda mais simples

$$D_i = 1 \quad (i = 1, \dots, n) \quad (6.14)$$

Multiplicando ambos os membros de (6.14) por x_i^p e elevando-os a $1/p$ obtém-se uma expressão de redimensionamento que pode ser utilizada na fase de modificação da estrutura do algoritmo de optimização indicado na Figura 6.1 [Ber87].

$$x_i^{q+1} = x_i^q (D_i^{1/p})^q \quad (i = 1, \dots, n) \quad (6.15)$$

Com esta expressão são calculados os valores das variáveis na iteração $q+1$, recorrendo apenas a grandezas conhecidas na iteração q . O processo iterativo deve ser interrompido quando duas soluções consecutivas forem aproximadamente iguais. Este facto implica que o critério de optimalidade relativo à expressão (6.14) é respeitado para todos os valores de i .

Na expressão (6.15) o valor escolhido para o parâmetro p influencia a velocidade de convergência. Em cada caso deve ser adoptado um valor que corresponda a um conveniente compromisso entre velocidade e fiabilidade da convergência. Para que o processo iterativo seja convergente é fundamental que o factor $D_i^{1/p}$ da relação (6.15) possua um valor superior à unidade quando x_i apresentar um valor inferior ao da solução final.

Uma vez que o critério de optimalidade (6.14) se baseia na expressão (6.7), em que figuram apenas derivadas da função objectivo e da restrição, o facto de D_i ser unitário para todos os valores de i não implica que a expressão (6.11) seja respeitada. A solução tem portanto que ser modificada, de modo que a função g passe a apresentar um valor nulo. Esta modificação tem de ser efectuada de tal modo que o critério de optimalidade (6.14) continue a

ser verificado. Não foi apresentado ainda nenhum processo de cálculo do multiplicador de Lagrange que figura na expressão (6.13). Convém salientar que com um só valor de λ a expressão (6.7) tem de ser verificada para qualquer valor de i . O critério de modificação da solução para que a restrição g seja respeitada, bem como o processo de cálculo do multiplicador de Lagrange serão em seguida apresentados no âmbito da minimização do peso de uma treliça sujeita a uma restrição de deslocamento.

O peso de uma treliça pode ser simplificada com a seguinte expressão, em que o peso específico ρ_i e o comprimento de cada barra L_i são constantes e a respectiva secção $x_i = a_i$ é variável

$$f(x_1, \dots, x_n) = \sum_{i=1}^n \rho_i L_i x_i \quad (6.16)$$

A função f é portanto a função objectivo do problema de minimização do peso de uma treliça. É considerada uma única restrição de deslocamento segundo um dos graus de liberdade da treliça

$$g(x_1, \dots, x_n) = d(x_1, \dots, x_n) - d_{\max} \leq 0 \quad (6.17)$$

Supõe-se que quando o único caso de carga actua na estrutura optimizada, o deslocamento restringido é positivo e apresenta o seu valor máximo. A restrição (6.17) encontra-se portanto activa. O deslocamento d pode ser obtido com a seguinte expressão em que E_i representa o módulo de Young do material da barra, N_i é o esforço axial quando actua a solicitação real e \bar{N}_i o esforço axial quando actua uma solicitação constituída por uma carga unitária segundo o deslocamento d

$$d(x_1, \dots, x_n) = \sum_{i=1}^n \frac{N_i(x_1, \dots, x_n) \bar{N}_i(x_1, \dots, x_n) L_i}{E_i x_i} \quad (6.18)$$

Nas treliças hiperstáticas os esforços axiais dependem das áreas das barras, constituindo as treliças isostáticas um caso particular em que N_i e \bar{N}_i são funções constantes. Derivando a função objectivo em ordem a x_j obtém-se

$$\frac{\partial f}{\partial x_j} = \rho_j L_j \quad (6.19)$$

A derivada da função g é igual à derivada do deslocamento d em ordem a x_j e apresenta a seguinte expressão

$$\frac{\partial g}{\partial x_j} = \frac{\partial d}{\partial x_j} = -\frac{N_j \bar{N}_j L_j}{E_j x_j^2} + \sum_{i=1}^n \left[\frac{L_i}{E_i x_i} \frac{\partial(N_i \bar{N}_i)}{\partial x_j} \right] \quad (6.20)$$

É possível demonstrar que o somatório que figura na expressão (6.20) é sempre nulo [Kir81], resultando daí uma grande simplificação na formulação do critério de optimalidade.

$$\frac{\partial g}{\partial x_j} = -\frac{N_j \bar{N}_j L_j}{E_j x_j^2} \quad (6.21)$$

Substituindo (6.19) e (6.21) em (6.7) obtém-se

$$x_j = \sqrt{\lambda} \sqrt{\frac{N_j \bar{N}_j}{E_j \rho_j}} \quad (6.22)$$

Esta expressão pode ser utilizada na fase de modificação da estrutura do algoritmo da Figura 6.1, porque se encontra nas condições da expressão (6.15). O cálculo do valor do multiplicador de Lagrange e a modificação da solução de modo a tornar a restrição g activa constituem um problema que neste caso se resolve do seguinte modo. Considerando (6.17) como uma igualdade e substituindo d pela expressão (6.18) obtém-se

$$d_{\max} = \sum_{j=1}^n \frac{N_j \bar{N}_j L_j}{E_j x_j} \quad (6.23)$$

Se em (6.23) se substituir x_j pelo segundo membro de (6.22), obtém-se uma expressão que fornece $\sqrt{\lambda}$ em função de d_{\max}

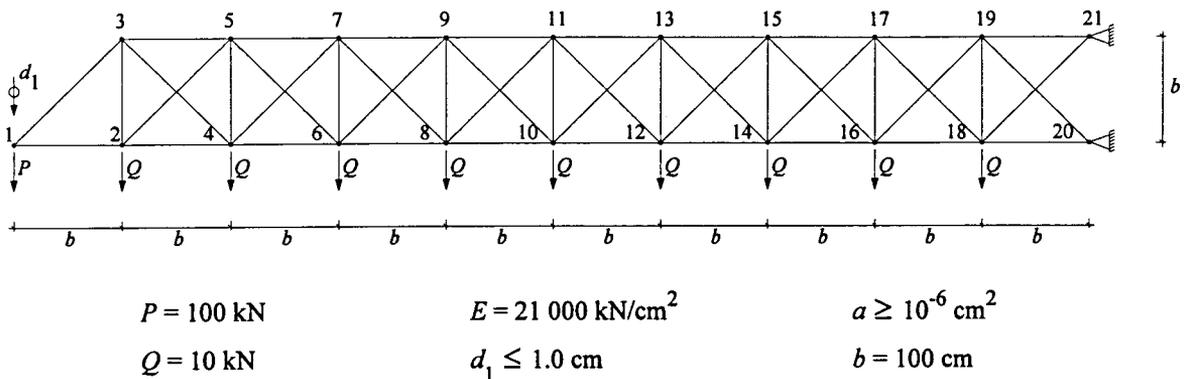
$$\sqrt{\lambda} = \frac{1}{d_{\max}} \sum_{j=1}^n \sqrt{\frac{N_j \bar{N}_j \rho_j L_j^2}{E_j}} \quad (6.24)$$

Utilizando este valor do multiplicador de Lagrange em (6.22), obtém-se uma expressão de redimensionamento que, ao ser utilizada no processo iterativo, conduz a uma convergência para uma solução em que a restrição g se encontra activa.

$$x_i = \frac{1}{d_{\max}} \sqrt{\frac{N_i \bar{N}_i}{E_i \rho_i}} \sum_{j=1}^n \sqrt{\frac{N_j \bar{N}_j \rho_j L_j^2}{E_j}} \quad (6.25)$$

6.1.4 - Optimização de uma treliça com 47 barras

Apresentam-se em seguida os resultados relativos à optimização das secções das barras de uma treliça sujeita a um caso de carga e a uma restrição de deslocamento. Na Figura 6.6 encontram-se todos os dados relativos a este problema.



Nota: não é considerado o peso próprio.

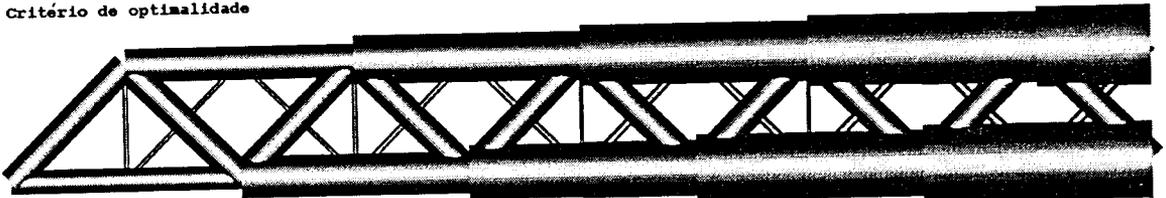
Figura 6.6 - Treliça com 47 barras - geometria, acção exterior e restrições.

Uma vez que todas as barras são constituídas por um único material, foi considerado um peso específico unitário, passando a função objectivo a ser o volume da estrutura. Este problema foi resolvido por dois métodos distintos. O primeiro baseou-se na aplicação do algoritmo da Figura 6.1, tendo a fase de modificação da estrutura sido concretizada com a relação (6.25). O processo iterativo foi interrompido quando duas soluções consecutivas se revelaram idênticas. Este facto indica que o critério de optimalidade (6.14) é respeitado. A solução assim obtida foi comparada com a que resulta da aplicação do programa NEWTOP (ver Capítulo 5). A resolução de problemas de optimização de estruturas com este programa será apresentada mais adiante neste capítulo.

Na aplicação de ambos os métodos foi exigida uma solução com um erro muito pequeno, tendo sido efectuado em ambos os casos um número de iterações muito superior ao habitual. No Quadro 6.2 encontra-se a melhor solução que foi obtida com cada um dos métodos. Estas duas soluções estão representadas graficamente na Figura 6.7, tornando-se assim mais perceptíveis as principais diferenças existentes entre elas. As barras estão representadas por cilindros cuja área da base é proporcional à área da secção transversal na solução óptima. Foram utilizados algoritmos de iluminação, de remoção de superfícies

escondidas e de *dithering*. Esta técnica de visualização de soluções óptimas é particularmente vantajosa no caso das treliças tridimensionais, sendo descrita mais adiante neste capítulo. Da comparação entre as duas soluções conclui-se que, apesar de algumas barras apresentarem áreas muito diferentes, os volumes finais são semelhantes, diferindo em menos de 1%. Esta pequena diferença é consequência do facto de nas barras de maior área as duas soluções serem idênticas. O aspecto mais significativo é o facto de a solução obtida com o programa NEWTOP corresponder a uma estrutura isostática, enquanto que na outra solução persiste um certo grau de hiperstaticidade. As deformadas representadas na Figura 6.8 realçam o facto de existirem diferenças significativas entre as duas soluções, apesar de os volumes e as deformadas na extremidade esquerda se revelarem praticamente coincidentes.

Critério de optimalidade



NEWTOP

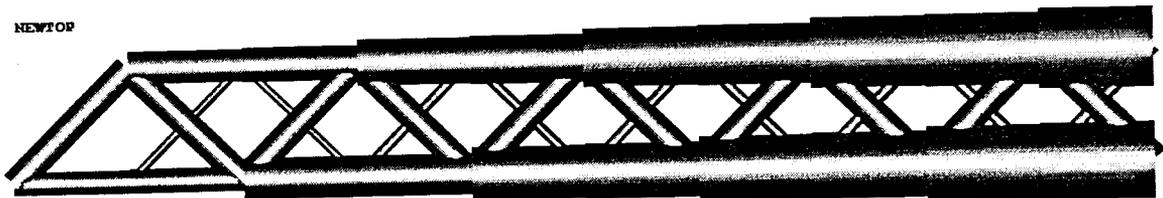


Figura 6.7 - Treliça com 47 barras - comparação entre a solução obtida com base num critério de optimalidade e a solução obtida com o programa NEWTOP.

Quadro 6.2 - Treliça com 47 barras - comparação entre as soluções obtidas com métodos de otimização distintos.

Barra	Área da barra (cm ²)	
	Critério de optimalidade	NEWTOP
1 - 2	65.259898	64.813875
2 - 4	67.346883	70.829277
4 - 6	203.253843	197.930401
6 - 8	209.615495	213.341352
8 - 10	353.179074	345.680225
10 - 12	363.978273	367.228430
12 - 14	514.347589	505.210094
14 - 16	529.881604	532.138421
16 - 18	685.988807	675.538365
18 - 20	706.649490	707.488716
3 - 5	131.659193	126.821135
5 - 7	135.826692	138.835934
7 - 9	275.662434	268.640190
9 - 11	284.204608	287.374239
11 - 13	431.256506	422.592513
13 - 15	444.295633	446.913863
15 - 17	597.605268	587.650176
17 - 19	615.590536	617.165656
19 - 21	774.184156	763.213243
1 - 3	92.291433	91.660661
2 - 5	2.942324	8.483828
4 - 7	98.157686	91.924513
6 - 9	6.016446	12.674980
8 - 11	103.183964	95.488653
10 - 13	9.159379	15.523625
12 - 15	107.580726	99.387743
14 - 17	12.399083	17.925145
16 - 19	111.376185	103.139535
18 - 21	15.241167	20.040928
3 - 4	93.852165	87.646582
5 - 6	2.942324	8.483825
7 - 8	99.209551	91.423424
9 - 10	6.016446	12.674980
11 - 12	103.871242	95.488655
13 - 14	9.159378	15.523625
15 - 16	107.849152	99.387745
17 - 18	12.399083	17.925146
19 - 20	111.376186	103.139536
2 - 3	2.864549	0.000001
4 - 5	0.000001	0.000001
6 - 7	2.546248	0.000001
8 - 9	0.000001	0.000001
10 - 11	2.279626	0.000001
12 - 13	0.000001	0.000001
14 - 15	1.267409	0.000001
16 - 17	0.000001	0.000001
18 - 19	0.000001	0.000001
Volume (cm ³) →	896148.364 (+0.94%)	887799.004 (melhor solução obtida)

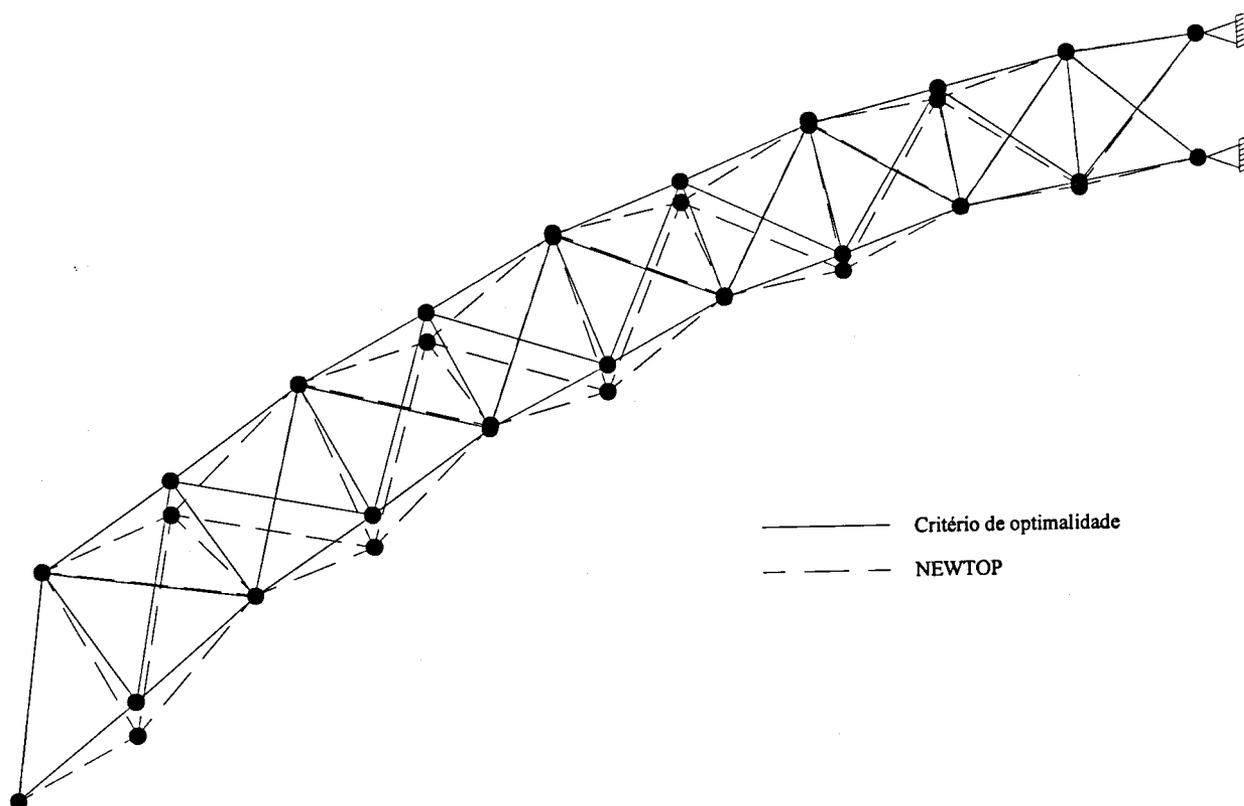


Figura 6.8 - Treliça com 47 barras - comparação das deformadas correspondentes às duas soluções obtidas.

Na Figura 6.9 está representada a evolução do volume da estrutura com o processo iterativo correspondente à otimização baseada num critério de optimalidade. O aspecto mais relevante que aí se pode observar é o da rápida convergência apresentada por este método. Após a realização de apenas duas iterações, o volume da estrutura apresenta já um valor que excede em menos de 1% o correspondente à solução final. Segue-se uma fase de convergência lenta, tendo sido necessário realizar centenas de iterações para obter os resultados do Quadro 6.2. Este número elevado de iterações teve como objectivo a obtenção de soluções muito precisas, de modo a facilitar a comparação entre os dois métodos.

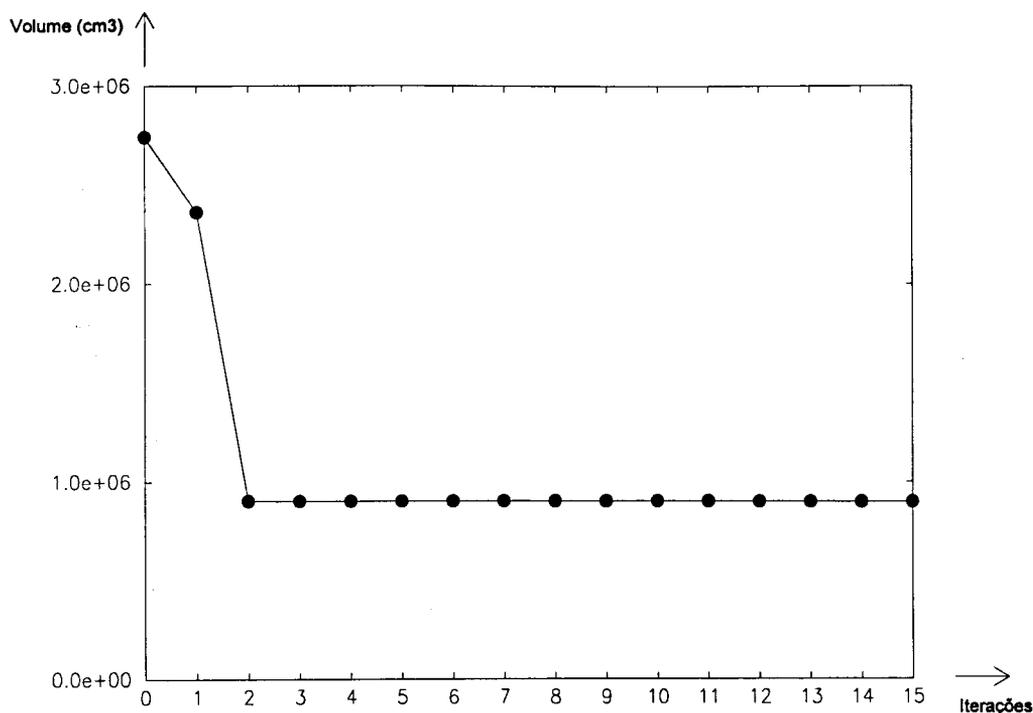


Figura 6.9 - Treliça com 47 barras - otimização baseada num critério de optimalidade.

6.1.5 - Observações finais

A resolução do mesmo problema por dois métodos distintos permitiu constatar que a verificação do critério de optimalidade (6.14) não implica que a solução corresponda ao mínimo global. O aspecto mais atraente dos métodos de otimização baseados em critérios de optimalidade consiste na convergência para uma solução óptima ou quase óptima com um reduzido número de iterações. A capacidade de resolução de problemas com um elevado número de variáveis e os baixos tempos de CPU que estes métodos apresentam são vantagens que resultam do facto de em cada iteração ser exigido ao computador pouco mais do que o esforço correspondente a uma análise da estrutura.

A simplicidade do problema abordado na secção 6.1.4 ocultou algumas dificuldades que surgem frequentemente na formulação dos métodos baseados em critérios de optimalidade. A adaptação do método apresentado na secção 6.1.3 a situações em que estão presentes múltiplas restrições de deslocamento e múltiplos casos de carga é possível, devendo nesse caso ser utilizadas expressões que vão actualizando os valores dos diversos multiplicadores de Lagrange durante o processo iterativo [Kir81]. A esta fase estão já associados alguns problemas, tais como a selecção dos valores iniciais dos multiplicadores de Lagrange e a definição do conjunto de restrições activas. Se no problema de optimização

estiverem também presentes restrições de tensão, não é possível deduzir uma fórmula de redimensionamento com as características da relação (6.25) [Ber87]. Esta dificuldade associada ao facto de as restrições de tensão estarem presentes em quase todos os problemas de optimização de estruturas constitui uma grave desvantagem para os métodos baseados em critérios de optimalidade. Estes métodos encontram-se limitados à resolução de problemas de minimização do peso da estrutura, não sendo possível com a formulação actual considerar outras grandezas como função objectivo [Van93].

Conclui-se portanto que a elevada eficiência associada aos métodos de optimização baseados em critérios de optimalidade só pode ser explorada em situações particulares, pelo facto de surgirem grandes dificuldades na formulação de problemas mais genéricos [Ram94]. As soluções que satisfazem critérios de optimalidade simples podem constituir uma referência na preparação de soluções iniciais destinadas a serem utilizadas em algoritmos de optimização mais versáteis.

6.2 - ANÁLISE DE SENSIBILIDADES

No algoritmo de optimização de estruturas apresentado na Figura 6.1, a fase de modificação da estrutura pode ser efectuada com recurso à programação matemática. Nesta fase os métodos habitualmente utilizados requerem o cálculo dos gradientes da função objectivo e das restrições. Na optimização de estruturas cujo comportamento é traduzido pelo método dos deslocamentos, é necessário calcular primeiras derivadas de funções que dependem das variáveis de projecto e dos deslocamentos dos nós. Se o comportamento da estrutura for linear, é possível e vantajoso considerar os deslocamentos de um modo implícito para que o número de variáveis do problema de optimização não seja muito elevado. Uma vez que a função objectivo não depende em geral dos deslocamentos dos nós, as considerações que se seguem apenas se referem ao cálculo de derivadas das restrições em ordem às variáveis de projecto, considerando os deslocamentos de um modo implícito. Este procedimento é habitualmente designado análise de sensibilidades.

6.2.1 - Formulação

Considerando, sem perda de generalidade, que no problema de optimização existe uma só restrição desigualdade g e uma só variável de projecto x , tem-se

$$g(x, \underline{u}(x)) \leq 0 \quad (6.26)$$

Nesta expressão \underline{u} representa o vector dos deslocamentos segundo os diversos graus de liberdade da estrutura. A consideração de múltiplas restrições e múltiplas variáveis de projecto implica apenas uma repetição dos procedimentos que serão em seguida apresentados. Na expressão (6.26) a função g depende explícita e implicitamente da variável x , uma vez que os deslocamentos também dependem do valor de x . A derivada total de g em ordem a x tem portanto que ser calculada do seguinte modo

$$\frac{dg}{dx} = \frac{\partial g}{\partial x} + \left[\frac{\partial g}{\partial \underline{u}} \right]^T \frac{d \underline{u}}{dx} \quad (6.27)$$

Se, por exemplo, a restrição g for uma restrição de tensão do tipo

$$g(x, \underline{u}) = \sigma(x, \underline{u}) - \sigma_{\max} \leq 0 \quad (6.28)$$

as derivadas de g coincidem com as derivadas de σ , uma vez que σ_{\max} é constante.

Em (6.27) o termo $\partial g / \partial x$ é em geral fácil de obter, quer por derivação de expressões simples, quer por diferenças finitas. No cálculo deste termo apenas é considerada informação relativa ao elemento que contém o ponto em que a tensão é restringida. No caso da optimização das secções das barras em treliças, o termo $\partial g / \partial x$ é nulo, porque a tensão instalada em cada barra não depende da área da respectiva secção transversal.

A derivada parcial $\partial g / \partial \underline{u}$ que figura em (6.27) é também fácil de obter pelos mesmos motivos atrás referidos para o caso de $\partial g / \partial x$. No vector $\partial g / \partial \underline{u}$ apenas são não nulas as derivadas em ordem aos deslocamentos dos nós do elemento que contém o ponto em que a tensão é restringida.

A derivada du/dx não pode em geral ser obtida directamente, pelo facto de não se dispor de expressões que relacionem os deslocamentos com as variáveis de projecto. Se o seu cálculo for efectuado por diferenças finitas, é necessário efectuar tantas análises da estrutura quantas as variáveis de projecto. Exceptuando situações particulares, cada perturbação de uma

variável de projecto introduz alterações na matriz de rigidez global, dando assim origem a um novo conjunto de deslocamentos em toda a estrutura. Recorrendo às equações de equilíbrio, o cálculo de \underline{du}/dx pode ser efectuado de um modo mais eficiente

$$\underline{K}(x)\underline{u}(x) = \underline{F}(x) \quad (6.29)$$

$$\underline{K} \frac{d\underline{u}}{dx} = \frac{d\underline{F}}{dx} - \frac{d\underline{K}}{dx} \underline{u} \quad (6.30)$$

Nesta expressão o termo $d\underline{F}/dx$ é facilmente obtido, quer por derivação de expressões simples, quer por diferenças finitas. Na generalidade dos casos só são não nulos os termos relativos aos graus de liberdade do elemento afectado pela variação de x . Estas considerações são extensivas ao cálculo de $d\underline{K}/dx$. Neste caso, em vez de derivar a matriz de rigidez global da estrutura, é possível e vantajoso efectuar a derivação ao nível do elemento e proceder em seguida ao espalhamento dessas derivadas.

No caso de existir apenas uma variável de projecto, o vector $d\underline{u}/dx$ é obtido com uma única análise da estrutura. Se existirem múltiplas variáveis de projecto, a expressão (6.30) corresponde a um sistema de equações com um número de vectores de termos independentes igual ao número de variáveis de projecto. Se o sistema de equações for resolvido por eliminação de Gauss ou factorização, a resolução de (6.30) com n vectores de termos independentes apresenta um custo muito inferior ao de n análises da estrutura. Este procedimento para o cálculo de $d\underline{u}/dx$ é designado método directo [Kod93].

Se se explicitar $d\underline{u}/dx$ na expressão (6.30), pode-se efectuar a sua substituição em (6.27), resultando

$$\frac{d\underline{g}}{dx} = \frac{\partial \underline{g}}{\partial x} + \left[\frac{\partial \underline{g}}{\partial \underline{u}} \right]^T \underline{K}^{-1} \left(\frac{d\underline{F}}{dx} - \frac{d\underline{K}}{dx} \underline{u} \right) \quad (6.31)$$

A utilização desta expressão não apresenta qualquer vantagem, porque nela figura a inversa da matriz de rigidez global, cujo cálculo tem geralmente um custo inoportável. Efectuando uma análise da estrutura com $\partial \underline{g} / \partial \underline{u}$ como vector solicitação, obtém-se um conjunto de deslocamentos $\underline{\Lambda}$, que serão em seguida utilizados no cálculo de $d\underline{g}/dx$. O recurso a esta análise auxiliar da estrutura permite a utilização de (6.31) sem a necessidade de inverter a matriz de rigidez global. O vector $\partial \underline{g} / \partial \underline{u}$ é designado solicitação da estrutura adjunta e os deslocamentos $\underline{\Lambda}$ são as variáveis adjuntas [Kod93]. O cálculo do vector $\underline{\Lambda}$ implica a resolução do seguinte sistema de equações lineares

$$\tilde{K} \tilde{\Lambda} = \frac{\partial \tilde{g}}{\partial \tilde{u}} \quad (6.32)$$

Efectuando a transposição de ambos os membros, seguida da multiplicação por \tilde{K}^{-1} , e atendendo à simetria de \tilde{K} e de \tilde{K}^{-1} , obtém-se

$$\left[\frac{\partial \tilde{g}}{\partial \tilde{u}} \right]^T \tilde{K}^{-1} = \left[\tilde{\Lambda} \right]^T \quad (6.33)$$

A substituição de (6.33) em (6.31) conduz à expressão final que permite efectuar o cálculo de dg/dx de um modo eficiente

$$\frac{dg}{dx} = \frac{\partial g}{\partial x} + \left[\tilde{\Lambda} \right]^T \left(\frac{dF}{dx} - \frac{dK}{dx} u \right) \quad (6.34)$$

Nesta expressão o cálculo de $\tilde{\Lambda}$ recorrendo a (6.32) é geralmente a fase que mais recursos exige do computador. Conforme foi atrás referido, as derivadas que figuram em (6.34) são em geral calculadas de um modo eficiente. O cálculo de dg/dx recorrendo a (6.32) e (6.34) é designado método da estrutura adjunta.

Em ambos os métodos apresentados, a fase que mais recursos exige do computador consiste na resolução de um sistema de equações, cuja matriz dos coeficientes é a matriz de rigidez global da estrutura. De acordo com (6.30), e designando por n o número de variáveis de projecto e por nc o número de casos de carga presentes no problema de optimização de estruturas, verifica-se que a utilização do método directo requer uma análise da estrutura com um número de vectores de termos independentes igual a $n \times nc$. Se for utilizado o método da estrutura adjunta, o número de vectores em (6.32) é igual ao número de restrições (m). A eficiência relativa dos dois métodos depende das características do problema de optimização, devendo ser utilizado aquele que apresentar um menor número de vectores de termos independentes em (6.30) ou (6.32).

6.2.2 - Exemplos

Apresenta-se em seguida um exemplo muito simples destinado a esclarecer a aplicação do método directo e do método da estrutura adjunta ao cálculo da derivada total de uma

restrição em ordem a uma variável de projecto. Na Figura 6.10 encontram-se as características de uma viga em consola de secção quadrada solicitada apenas pelo peso próprio.

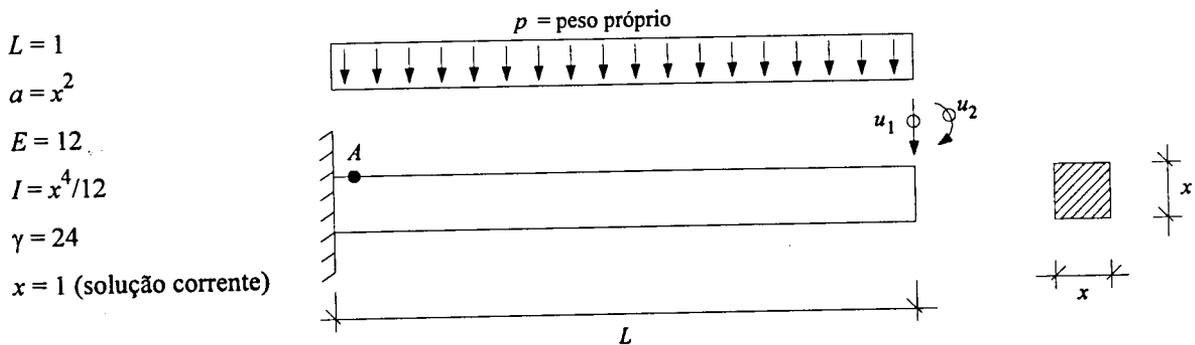


Figura 6.10 - Análise de sensibilidades numa viga em consola.

Neste problema existe uma única variável de projecto, que é a largura da secção quadrada. A restrição cuja derivada se pretende conhecer resulta do facto de a tensão normal no ponto A não poder exceder σ_{\max} .

$$g = \sigma_A - \sigma_{\max} \leq 0 \quad (6.35)$$

Devido ao facto de a consola ser uma estrutura isostática, neste problema seria simples calcular $g(x)$ e dg/dx recorrendo apenas a equações de equilíbrio. No entanto, tendo em vista a exemplificação do cálculo de dg/dx em estruturas hiperstáticas, são considerados dois graus de liberdade (u_1, u_2) e as respectivas equações de equilíbrio. A análise de sensibilidades é efectuada com base num valor unitário da variável x .

Recorrendo às expressões habituais relativas à aplicação do método dos deslocamentos a estruturas porticadas, obtém-se a seguinte expressão para a restrição g

$$g = \frac{6}{x^3} \left(\frac{6EI}{L^2} u_1 - \frac{2EI}{L} u_2 + \frac{pl^2}{12} \right) - \sigma_{\max} \leq 0 \quad (6.36)$$

Substituindo as constantes de acordo com os dados resulta

$$g = 36x u_1 - 12x u_2 + \frac{12}{x} - \sigma_{\max} \quad (6.37)$$

Uma vez que σ_{\max} é uma constante, a restrição g apenas depende de x e de u , encontrando-se portanto nas condições da expressão (6.28). A obtenção de dg/dx pelo método

directo requer os seguintes cálculos preliminares relativos à análise da estrutura pelo método dos deslocamentos

$$K = \begin{bmatrix} \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} = \begin{bmatrix} 12x^4 & -6x^4 \\ -6x^4 & 4x^4 \end{bmatrix} = \begin{bmatrix} 12 & -6 \\ -6 & 4 \end{bmatrix} \quad (6.38)$$

$$F = \begin{bmatrix} \frac{pL}{2} \\ -\frac{pL^2}{12} \end{bmatrix} = \begin{bmatrix} 12x^2 \\ -2x^2 \end{bmatrix} = \begin{bmatrix} 12 \\ -2 \end{bmatrix} \quad (6.39)$$

$$\tilde{K} \tilde{u} = \tilde{F} \Rightarrow \tilde{u} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \quad (6.40)$$

Procede-se em seguida ao cálculo das derivadas parciais de (6.37)

$$\frac{\partial g}{\partial x} = 36u_1 - 12u_2 - \frac{12}{x^2} = 48 \quad (6.41)$$

$$\frac{\partial g}{\partial u} = \begin{bmatrix} 36x \\ -12x \end{bmatrix} = \begin{bmatrix} 36 \\ -12 \end{bmatrix} \quad (6.42)$$

A obtenção de $d\tilde{u}/dx$ com (6.30) requer o cálculo das seguintes derivadas

$$\frac{d\tilde{F}}{dx} = \begin{bmatrix} 24x \\ -4x \end{bmatrix} = \begin{bmatrix} 24 \\ -4 \end{bmatrix} \quad (6.43)$$

$$\frac{d\tilde{K}}{dx} = \begin{bmatrix} 48x^3 & -24x^3 \\ -24x^3 & 16x^3 \end{bmatrix} = \begin{bmatrix} 48 & -24 \\ -24 & 16 \end{bmatrix} \quad (6.44)$$

Substituindo (6.38), (6.43), (6.44) e (6.40) em (6.30) e resolvendo o sistema de equações que daí resulta, obtém-se

$$\frac{d\tilde{u}}{dx} = \begin{bmatrix} -6 \\ -8 \end{bmatrix} \quad (6.45)$$

A substituição de (6.41), (6.42) e (6.45) em (6.27) conduz ao valor final de dg/dx que é -72.

O cálculo desta mesma derivada pelo método da estrutura adjunta requer a resolução do sistema de equações (6.32), cuja solução é

$$\tilde{\Lambda} = \begin{bmatrix} 6 \\ 6 \end{bmatrix} \quad (6.46)$$

Substituindo (6.41), (6.46), (6.43), (6.44) e (6.40) em (6.34), obtém-se também -72 como valor de dg/dx . Esta coincidência de resultados era esperada porque em ambos os métodos foram utilizados os valores exactos das derivadas e não foram efectuadas simplificações nem arredondamentos. As diferenças fundamentais entre os dois métodos envolvem apenas questões relacionadas com a eficiência computacional de cada um deles.

Para que a aplicação dos métodos de análise de sensibilidades não se restringisse ao exemplo simples atrás apresentado, foi desenvolvido um programa de computador destinado ao cálculo de derivadas de tensões em ordem às variáveis de projecto. Uma vez que diversas fases do algoritmo são semelhantes às que já se encontram disponíveis em programas de análise de estruturas, optou-se por efectuar o desenvolvimento a partir de um programa simples de análise de treliças planas. Devido ao facto de os dois métodos de análise de sensibilidades atrás descritos apresentarem grandes semelhanças entre si, apenas foi programado o da estrutura adjunta. Os resultados que podem ser obtidos com este programa consistem essencialmente nas derivadas de tensões e deslocamentos em ordem às variáveis de projecto, que são as áreas das barras.

Na Figura 6.11 encontra-se a representação gráfica da derivada da tensão na barra AB (σ_{AB}) em ordem a cada variável de projecto (x_i). Cada barra foi representada com uma espessura que é proporcional ao valor absoluto de $d\sigma_{AB}/dx_i$. Para distinguir os valores positivos dos negativos foram utilizados dois tons de cinzento. O maior valor absoluto da derivada está associado à variação da tensão em AB quando varia a área da própria barra AB . Na maior parte das barras, a um aumento da área da secção transversal corresponde uma diminuição da tensão em AB , implicando um sinal negativo da derivada. Na Figura 6.11 pode-se verificar que só um reduzido número de barras apresenta derivada positiva.

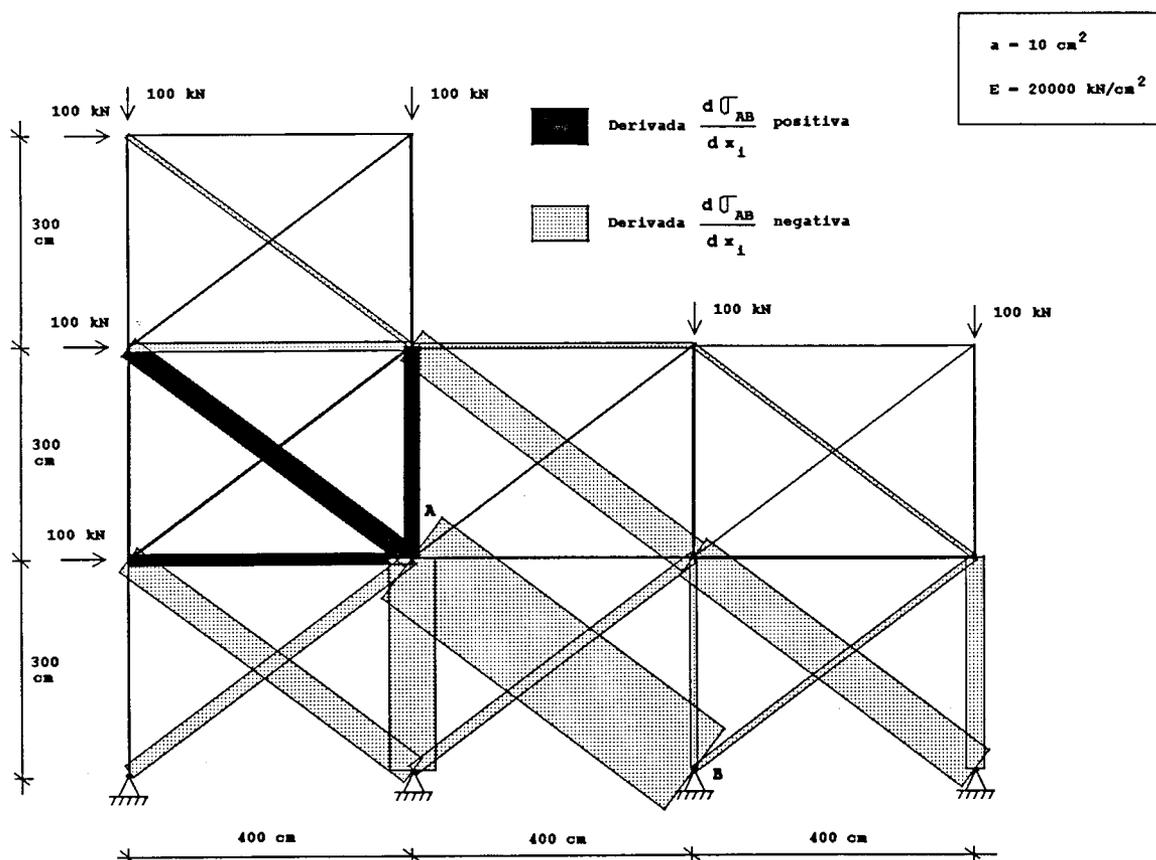


Figura 6.11 - Treliça sujeita a acções horizontais - representação de $d\sigma_{AB} / dx_1$ com barras cuja espessura é proporcional ao valor da derivada.

6.2.3 - Observações finais

Os exemplos simples atrás apresentados destinaram-se apenas a transmitir parte da experiência adquirida com a utilização dos métodos de análise de sensibilidades. Para além das formulações aqui apresentadas existem outras, que apesar de serem semelhantes, se revelam vantajosas em certas circunstâncias [Bel82]. Estes métodos são ainda extensíveis ao cálculo de segundas derivadas [Kod93] e à análise de sensibilidades de estruturas com comportamento não linear [Aro87a].

Os métodos de análise de sensibilidades são geralmente associados a programas convencionais de análise de estruturas e a módulos de programação matemática, que são utilizados como caixa negra [Mot88]. De um modo geral, apenas são calculadas primeiras derivadas da função objectivo e restrições e apenas é abordada a optimização de estruturas com comportamento linear. Em certas situações, tais como a optimização de forma em meios

contínuos, algumas das derivadas utilizadas na análise de sensibilidades não são passíveis de ser calculadas analiticamente, tornando necessário o recurso ao respectivo cálculo por diferenças finitas [Esp87] [Bec91]. Nestes casos, surge imediatamente a dificuldade da escolha do valor da perturbação a utilizar no cálculo dessas derivadas. De um modo geral, verifica-se que o rigor dos resultados depende significativamente do valor desse parâmetro [Kod93].

Os métodos tradicionais de optimização de estruturas, baseados em critérios de optimalidade ou na análise de sensibilidades, foram desenvolvidos com o objectivo de apresentarem uma *performance* que tornasse possível a resolução de problemas reais. De um modo geral, esse aumento de eficiência foi conseguido recorrendo a uma simplificação da aproximação do problema inicial. Esta estratégia teve como consequência uma considerável diminuição da robustez e da versatilidade dos métodos de optimização [Aro87b]. Estudos comparativos entre diversos métodos aplicados à optimização de estruturas revelaram a existência de grandes dispersões de resultados [Bel82] e de falta de fiabilidade na obtenção do mínimo global [Van84]. Uma comparação mais pormenorizada entre esses métodos e os que foram desenvolvidos no âmbito do presente trabalho é apresentada no Capítulo 5 e no final deste capítulo. Numa tentativa de resolver os problemas dos métodos tradicionais, sem comprometer significativamente a eficiência, optou-se no presente trabalho por abordar a optimização de estruturas por intermédio da formulação integrada (Figura 6.2) e recorreu-se a um método de segunda ordem em que todas as derivadas são calculadas analiticamente pelo próprio programa de computador (ver Capítulo 5). Os pormenores desta formulação são em seguida apresentados.

6.3 - OPTIMIZAÇÃO DE TRELIÇAS PELO MÉTODO DE NEWTON - FORMULAÇÃO INTEGRADA

O programa de cálculo automático que recorre ao método de Newton para a resolução do problema de minimização de uma função com restrições (NEWTOP) não utiliza nenhum dos métodos de análise de sensibilidades atrás descritos. Este programa necessita de um conjunto de ficheiros de dados que descrevam o programa matemático e respectiva solução inicial, procedendo internamente ao cálculo de todas as primeiras e segundas derivadas. É portanto necessário acrescentar os deslocamentos ao conjunto das variáveis de projecto (formulação integrada), aumentando assim a dimensão do programa matemático [Haf85] [Bar89]. Na prática, este facto só se afigura como um grave inconveniente quando o número

de casos de carga é elevado, porque as variáveis de projecto relativas a deslocamentos têm de ser repetidas para cada um deles. Este aumento do número de variáveis também se torna significativo quando a estrutura tem poucas variáveis de projecto independentes e muitos graus de liberdade. A opção pela formulação integrada baseia-se no facto de se pretender abordar a optimização estrutural de um modo mais versátil e robusto, que permite uma extensão simples às estruturas com comportamento não linear. Surpreendentemente, o sacrifício da eficiência acabou por se revelar pouco significativo.

6.3.1- Geração do programa matemático

O programa de computador que procede à geração das expressões que constituem o programa matemático foi desenvolvido a partir do programa FEMIX [Aze92]. Este programa permite efectuar a análise de estruturas com comportamento linear, podendo a respectiva discretização ser efectuada por elementos finitos planos, tridimensionais, laminares ou prismáticos. Nestes últimos pode ser considerado um comportamento apenas axial (treliça) ou axial e de flexão (pórtico). Neste capítulo apenas são referidos os elementos de treliça e no Capítulo 7 os resultados da optimização são validados recorrendo ao elemento de pórtico.

A geração dos ficheiros que constituem os dados do programa NEWTOP é efectuada a partir de dois ficheiros que contêm a descrição da treliça e alguns elementos específicos do problema de optimização.

O primeiro ficheiro contem toda a informação que pode ser partilhada com o programa de análise linear (FEMIX) e que consiste essencialmente nos seguintes blocos de dados:

- geometria;
- condições fronteira;
- propriedades dos materiais;
- características das secções transversais das barras;
- acções em cada caso de carga .

Os dados que não são fixos do ponto de vista da optimização (e.g., secção transversais) são incluídos na solução inicial. Em cada caso de carga têm de figurar todas as acções que constituem uma combinação, ao contrário do que é habitual na análise linear de estruturas.

O segundo ficheiro contém os dados específicos do problemas de optimização, sendo constituído fundamentalmente pelos seguintes blocos de dados:

- custo específico dos materiais;
- valores mínimos e máximos dos parâmetros que definem a secção transversal;
- parâmetros associados ao relacionamento entre o momento de inércia e a área da secção transversal;
- valores mínimos e máximos das tensões nos materiais para cada caso de carga;
- valores mínimos e máximos de deslocamentos de nós para cada caso de carga.

Os valores limite das tensões nos materiais dependem do caso de carga, porque, no caso de coexistirem combinações relativas a diferentes estados limites, pode haver necessidade de limitar nuns casos as tensões ao seu valor de cálculo e noutros a valores de segurança.

Os ficheiros de dados do programa de optimização (NEWTOP) têm de conter a seguinte informação:

- ⇒ Função objectivo
- ⇒ Restrições igualdade
 - equações de equilíbrio dos nós
- ⇒ Restrições desigualdade
 - restrições de área mínima
 - restrições de área máxima
 - restrições de deslocamento em nós
 - restrições de tensão máxima e mínima
 - restrições de encurvadura local
- ⇒ Substituição de variáveis
 - deslocamentos fixos
 - ligação entre variáveis
- ⇒ Restrições adicionais acrescentadas pelo utilizador
- ⇒ Declaração dos nomes das variáveis
- ⇒ Solução inicial

Função objectivo

Para o tipo de problemas abordados neste capítulo, a função objectivo pode sempre ser considerada como um custo que se pretende minimizar. A sua expressão genérica é a seguinte

$$C = \sum_{i=1}^{NB} c_i L_i a_i \quad (6.47)$$

sendo C o custo associado à solução, NB o número de barras, c_i o custo por unidade de volume relativo ao material da barra i , L_i e a_i respectivamente o comprimento e a área da barra. Se se pretender minimizar o volume da estrutura os coeficientes c_i devem ser unitários. Se se pretender minimizar o peso da estrutura os coeficientes c_i devem representar o peso por unidade de volume do material da barra.

Uma vez que a geometria da estrutura e as propriedades dos materiais se supõe constantes, a função objectivo é uma função linear das áreas das barras.

Restrições igualdade

Cada grau de liberdade não impedido necessita de uma equação de equilíbrio de forças que constitui uma restrição igualdade. Estas restrições são funções que dependem das áreas das barras que convergem no nó e dos deslocamentos dos nós associados a essas barras. Em cada termo dessas funções surgem produtos área-deslocamento que fazem com que as equações de equilíbrio sejam restrições não lineares.

Nas equações de equilíbrio figuram também as acções exteriores e o peso próprio das barras, que depende da respectiva área.

As equações de equilíbrio relativas a graus de liberdade impedidos podem ser omitidas porque cada uma delas introduz uma nova variável, que é a reacção de apoio. Por este motivo estas restrições não influenciam a solução do problema de optimização. Exceptuam-se as raras situações em que uma ou mais reacções de apoio são referidas nas restrições desigualdade.

Restrições desigualdade

As restrições desigualdade que limitam as áreas das secções transversais e os deslocamentos dos nós impõe valores mínimos e máximos das variáveis de projecto (*side constraints*), sendo portanto restrições lineares do tipo

$$x_{\min} \leq x_i \leq x_{\max} \quad (6.48)$$

As restrições de área mínima devem ser sempre incluídas na formulação, para evitar que na solução apareçam áreas negativas, que não têm significado físico, nem áreas nulas, que implicam uma alteração da topologia da estrutura.

As restrições de área máxima devem ser evitadas porque podem facilmente provocar a inexistência de uma solução admissível. A sua inclusão conduz também a um aumento da probabilidade de se obter uma solução com valores negativos nos multiplicadores de Lagrange associados a restrições desigualdade.

Nas estruturas articuladas, a tensão instalada em cada barra depende linearmente dos deslocamentos dos respectivos nós de extremidade, sendo independente da área da secção transversal. As expressões das tensões intervêm directamente nas restrições de tensão máxima, mínima e de encurvadura local, que apresentam as seguintes expressões genéricas

$$\sigma_{\min} \leq \sigma(d) \leq \sigma_{\max} \quad (6.49)$$

$$\sigma_{cr} \leq \sigma(d) \quad (6.50)$$

Em (6.50) σ_{cr} depende da menor inércia da secção transversal, que na generalidade dos casos pode ser expressa em função da área.

$$\sigma_{cr} = -\frac{\pi^2 E}{L_0^2} \frac{I(a)}{a} \quad (6.51)$$

O programa de computador que procede à geração automática do programa matemático destinado à minimização do custo da estrutura considera as seguintes simplificações:

- todos os nós da estrutura articulada estão de alguma forma contraventados;
- o comprimento de encurvadura (L_0) coincide com o comprimento da barra (L);
- não é considerada qualquer curva de transição entre a encurvadura em regime elástico e a ausência total de encurvadura;
- a relação $I(a) / a$ é aproximada por um polinómio do terceiro grau.

Substituição de variáveis

As equações de equilíbrio dos nós e as restrições de tensão são formuladas e gravadas no ficheiro que contém o programa matemático sem considerar os deslocamentos impedidos, que são colocados num ficheiro auxiliar destinado a armazenar restrições susceptíveis de serem directamente substituídas no programa matemático. Deste modo consegue-se uma redução do número de variáveis de projecto e uma simplificação do algoritmo de geração do programa matemático. Estas restrições podem ser do tipo

$$x_i = \text{const.} \quad (6.52)$$

$$x_i = \text{const.} * x_j \quad (6.53)$$

A substituição destas restrições é efectuada pelo programa NEWTOP de um modo automático e independente do problema que está a ser tratado (ver secção 5.3).

Nos problemas de optimização pode haver interesse em obter uma solução em que um determinado grupo de variáveis de projecto apresente o mesmo valor final. É também frequente pretender-se uma solução simétrica quando as acções não são simétricas. Este procedimento designa-se por ligação entre variáveis (*variable linking*) e deve ser incluído no programa matemático por intermédio da substituição de variáveis, utilizando as restrições do tipo (6.53).

Restrições adicionais acrescentadas pelo utilizador

O utilizador pode ainda acrescentar ao programa matemático restrições igualdade ou desigualdade distintas das que são automaticamente geradas. Estas restrições adicionais têm de ser criteriosamente estudadas de modo a não tornar o problema de optimização impossível. Apresenta-se como exemplo da necessidade de incluir restrições adicionais uma situação em que se pretende uma solução óptima para uma estrutura que possui um ângulo que tem de permanecer constante quando actua a acção exterior. Pretende-se ainda que esta condição seja verificada sem efectuar qualquer alteração da topologia da estrutura. Na Figura 6.12 apresenta-se um exemplo em que se pretende que os segmentos \overline{AB} e \overline{BC} permaneçam ortogonais.

A restrição que tem de ser acrescentada ao programa matemático é neste caso

$$L_{AB}d_{c_2} - L_{AB}d_{B_2} + L_{BC}d_{A_1} - L_{BC}d_{B_1} = 0 \quad (6.54)$$

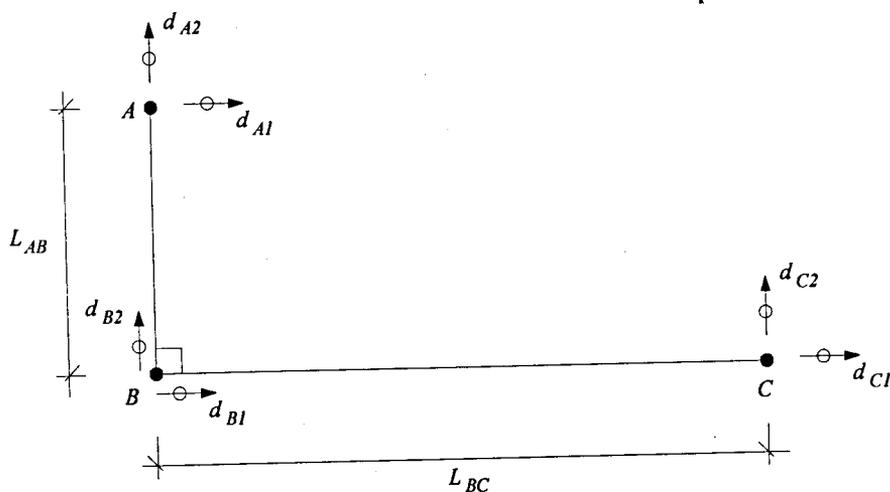


Figura 6.12 - Restrições adicionais destinadas a garantir que os segmentos \overline{AB} e \overline{BC} permanecem ortogonais.

Declaração dos nomes das variáveis

A declaração dos nomes das variáveis destina-se, tal como nas linguagens de programação, a permitir uma validação o mais completa possível de todas as componentes do programa matemático. Esta validação é particularmente importante quando o programa matemático é preparado ou modificado pelo utilizador.

Solução inicial

O aspecto mais delicado da preparação do conjunto de dados para o programa NEWTOP é o da selecção da solução inicial. Na optimização das secções das barras de treliças, a solução inicial engloba três grandes grupos de variáveis: áreas das barras, deslocamentos segundo graus de liberdade não impedidos e deslocamentos segundo graus de liberdade impedidos.

O programa que procede à geração do programa matemático considera como valores iniciais das áreas das barras os valores correspondentes à solução corrente que se encontra no ficheiro de dados que é comum à análise de estruturas. Contudo, essas áreas iniciais têm na generalidade dos casos que ser pré-dimensionadas recorrendo a um critério mais ou menos elaborado. Se a solução inicial se encontrar muito distante da solução final do problema de optimização, corre-se o risco de não conseguir convergência.

Os deslocamentos segundo graus de liberdade não impedidos devem ter como valores iniciais os correspondentes ao conjunto de secções transversais considerado nessa mesma

solução inicial. Se isto não se verificar, torna-se mais difícil conseguir uma solução para o problema de optimização. No caso da optimização de treliças com comportamento elástico é fácil satisfazer este requisito, porque o programa NEWTOP calcula os valores iniciais de n variáveis declaradas como variáveis dependentes, desde que existam n restrições igualdade que constituam um sistema de equações. No presente caso, este sistema é constituído por n equações lineares, que são as equações de equilíbrio relativas aos graus de liberdade não impedidos, sendo as n incógnitas os respectivos deslocamentos. Se a estrutura não for hipostática as n equações são linearmente independentes e a solução do sistema de equações obtém-se com uma única iteração (ver Capítulo 5).

Os deslocamentos segundo graus de liberdade impedidos podem ter qualquer valor como solução inicial, porque se tratam de variáveis que vão ser substituídas antes de se calcular os valores iniciais dos restantes deslocamentos.

Uma vez que os valores iniciais dos deslocamentos ficam automaticamente definidos depois de escolhidas as áreas iniciais das barras, são apenas estas as variáveis cujo valor inicial tem de ser arbitrado pelo utilizador. Apresentam-se em seguida três estratégias alternativas para determinar os valores iniciais das áreas das barras.

- 1) Começar com uma solução constante, iniciar o processo iterativo com um máximo de 4 e 6 iterações e utilizar a solução corrente após este limitado número de iterações como solução inicial de um novo processo iterativo. Pode ser necessário repetir mais do que uma vez este procedimento. Esta estratégia deve ser utilizada quando se parte de uma solução inicial em que todas as barras têm a mesma secção, ou sempre que a solução inicial se encontre muito distante da solução final. Uma vez que os parâmetros associados ao *scaling* são calculados com base na solução inicial, revela-se importante para a fiabilidade do processo iterativo que esses parâmetros sejam calculados com base numa solução tão próxima quanto possível da solução final. Na generalidade dos casos, incluindo aqueles em que se inicia o processo iterativo a partir de uma solução distante da final, verifica-se que nas primeiras iterações a solução evolui na direcção correcta, justificando-se assim a estratégia atrás sugerida.
- 2) Nos casos em que apenas se estejam a considerar restrições de tensões e de área mínima, deve-se calcular em primeiro lugar a solução correspondente ao *Fully Stressed Design* (FSD). Quer se trate de um problema com um caso de carga ou com vários casos de carga, a solução FSD pode ser um mínimo local e não se dispõe de multiplicadores de Lagrange para a diagnosticar. Por este motivo, deve-se tentar obter uma nova solução ou confirmar a anterior recorrendo ao programa

NEWTOP. Nestes casos a solução inicial deve ser a solução FSD com as secções transversais das barras multiplicadas por um factor arbitrário compreendido entre 1.1 e 1.5.

- 3) Nos casos em que são também consideradas restrições relativas a deslocamentos de nós, pode-se adoptar uma estratégia semelhante à do ponto anterior, com a única diferença de ser neste caso utilizado um factor multiplicativo das áreas das barras que torne a solução admissível. Este factor pode ser facilmente determinado percorrendo todas as restrições deslocamento em todos os casos de carga e calculando em cada caso a razão entre o deslocamento na solução FSD e o deslocamento admissível. O maior destes valores é o factor procurado, que será adiante referido por K_{FSD} . Para evitar que uma ou mais restrições deslocamento se encontrem activas para a solução inicial do programa NEWTOP, deve-se utilizar um valor de K_{FSD} ligeiramente superior ao que foi determinado pelo processo atrás referido.

O tempo de CPU associado à determinação da solução inicial pelos processos atrás referidos é muito inferior ao correspondente à fase iterativa pelo método de Newton. Verifica-se ainda que a solução FSD se obtém com uma precisão satisfatória efectuando 50 a 100 ciclos de análise e dimensionamento da estrutura. Na generalidade dos casos não se revela necessário recorrer a outros processos para a obtenção da solução inicial. As situações em que não se obtém convergência são em geral ultrapassadas recorrendo a modificações de alguns parâmetros que condicionam o comportamento do processo iterativo. Referem-se em seguida algumas situações em que se torna necessário proceder de um modo diferente do habitual.

Sempre que se verifiquem acentuadas variações da solução nas primeiras iterações ou quando variáveis que têm de ser positivas (e.g., áreas das secções) assumirem valores negativos, deve-se diminuir o valor máximo do parâmetro de pesquisa unidimensional (α_{\max}) e/ou aumentar o valor inicial dos multiplicadores de Lagrange. O parâmetro α_{\max} poderá regressar ao valor unitário depois do processo iterativo ter estabilizado.

Se o parâmetro de pesquisa unidimensional (α) assumir valores muito pequenos durante várias iterações consecutivas, apresentando-se já a solução com um resíduo também pequeno, deve-se interromper o processo iterativo e reiniciá-lo a partir da solução corrente multiplicada por um factor arbitrário. Este comportamento do processo iterativo indica que a solução se dirigiu para uma região em que o resíduo é pequeno, mas não existem pontos em que o gradiente do Lagrangeano seja nulo (ver secção 5.5). Nas proximidades de um ponto com estas características, a aproximação quadrática não é suficientemente correcta para originar um salto para uma zona em que exista uma solução. No caso da optimização de

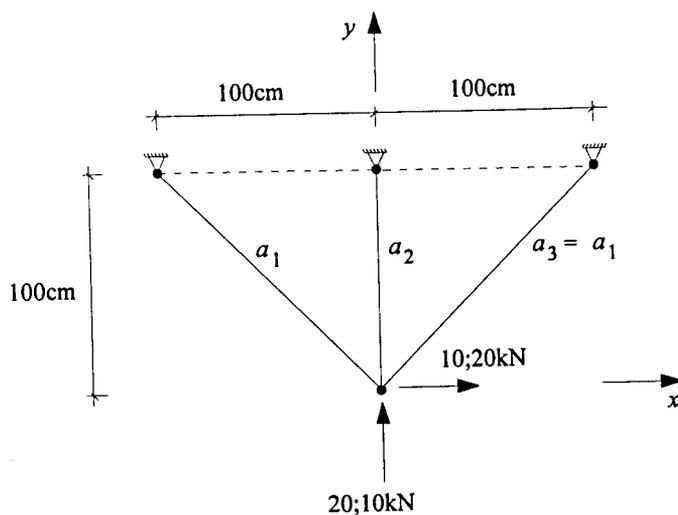
treliças, esta situação ocorre quando um número reduzido de barras assume o respectivo valor mínimo e, nessas circunstâncias, não existe nenhuma solução do problema de optimização. A probabilidade de ocorrência desta situação aumenta quando o valor mínimo das áreas das secções transversais é muito pequeno. Neste caso é conveniente efectuar uma primeira optimização com um valor mínimo maior do que o pretendido e em seguida partir desta solução com os dados correctos.

Por último refere-se a estratégia a adoptar quando se obtém uma solução final com valores negativos em multiplicadores de Lagrange associados a restrições desigualdade (λ_j^g) . Este facto indica que existem outras soluções admissíveis às quais corresponde um valor inferior da função objectivo. Os λ_j^g negativos podem aparecer em restrições de deslocamento, de tensão ou de área mínima. No primeiro caso é suficiente repetir o processo iterativo a partir da última solução multiplicada por um factor arbitrário compreendido entre 1.1 e 1.5. Nos restantes casos, além da multiplicação por um factor, deve-se aumentar significativamente as áreas das barras associadas às restrições que apresentaram um λ_j^g negativo.

6.3.2 - Treliça simétrica com três barras e dois casos de carga

Com o objectivo de clarificar todos os aspectos relacionados com a optimização de treliças, apresenta-se em seguida um exemplo com um reduzido número de variáveis. Nestas circunstâncias, torna-se possível representar graficamente a função objectivo e a região admissível, bem como deduzir o valor exacto das diversas soluções do programa matemático. O conteúdo dos principais ficheiros de dados e resultados pode neste caso ser publicado, uma vez que possuem um reduzido número de linhas.

A treliça que se pretende optimizar é plana, possui três barras e pode ser solicitada de dois modos distintos (ver Figura 6.13). Apesar de as acções não serem simétricas, pretende-se uma solução simétrica, pelo que se torna necessário acrescentar a restrição $a_3 = a_1$. Este problema é semelhante a outros que são frequentemente utilizados por diversos autores, não sendo habitualmente apresentado numa forma *standard*, uma vez que se destina a ilustrar diversos aspectos da optimização de estruturas [Sch60] [Sve68] [Van84] [Aro89]. O problema de optimização descrito na Figura 6.2 apresenta algumas modificações relativamente aos exemplos da bibliografia, destinadas a salientar o aparecimento de múltiplas soluções.



$$E = 20\,000 \text{ kN/cm}^2$$

$$\bar{\sigma} = 14 \text{ kN/cm}^2$$

$$a_{\min} = 0.15 \text{ cm}^2$$

Nota: não é considerado o peso próprio nem a encurvadura.

Custo do material utilizado em a_1 e $a_3 = 2$ Unidades de Custo/cm³
 Custo do material utilizado em $a_2 = 1$ Unidade de Custo/cm³

Figura 6.13- Treliça simétrica com 3 barras e 2 casos de carga.

Na Listagem 6.1 encontra-se o ficheiro de dados que contém a informação que é partilhada com o programa de análise linear de estruturas (FEMIX) [Aze92]. Este ficheiro pode ser preparado com os utilitários habituais de geração de malhas, genericamente designados por *Computer Aided Design* (CAD). As áreas das barras que figuram no ficheiro de dados são consideradas como solução inicial do problema de optimização. Os dados específicos do problema de optimização estrutural encontram-se num ficheiro distinto do anterior, cujo conteúdo se encontra na Listagem 6.2. Esta estratégia de distribuição dos dados por dois ficheiros possibilita a utilização dos programas relativos à análise de estruturas, quer na geração dos dados, quer na pesquisa por tentativas de uma solução inicial para o método de optimização, tão próxima quanto possível do mínimo global que se pretende determinar. O programa FEMIX e respectivos programas auxiliares são ainda utilizados na verificação da solução fornecida pelo programa de optimização, na visualização gráfica dos vários aspectos relacionados com essa solução e no cálculo de resultados que não são envolvidos no processo de optimização. A verificação da solução final destina-se a detectar eventuais erros na geração do programa matemático ou uma eventual utilização deficiente do programa NEWTOP. Na ausência destes dois tipos de erros, a referida verificação não seria necessária uma vez que o programa de optimização só fornece soluções admissíveis.

Listagem 6.1 - Treliça com 3 barras - dados partilhados com o programa de análise linear de estruturas.

```

### Main title of the problem
Trelica simetrica com 3 barras e 2 casos de carga (kN,cm)

### Main parameters
3 # nelem (n. of elements in the mesh)
4 # npoin (n. of points in the mesh)
4 # nvfix (n. of points with fixed degrees of freedom)
2 # ncase (n. of load cases)
2 # nmats (n. of sets of material properties)
1 # nspen (n. of sets of element nodal properties)
8 # ntype (problem type)
2 # nnode (n. of nodes per element)
2 # ngaus (n. of Gauss points in the integration rule) (element stiffness)
2 # ngstr (n. of Gauss points in the integration rule) (stresses)
3 # ndime (n. of geometric dimensions)
3 # ndofn (n. of degrees of freedom per node)
4 # nprop (n. of material properties used in the formulation)
1 # npren (n. of element nodal properties used in the formulation)

### Material properties index, element nodal properties index and
### list of the nodes of each element
# ielem matno ielnp      lnods ...
  1      1      1      1      2
  2      2      1      1      3
  3      1      1      1      4

### Coordinates of the points
# ipoin      coord-x      coord-y      coord-z
  1          0.0          0.0          0.0 # cm
  2         -100.0         100.0          0.0 # cm
  3          0.0          100.0          0.0 # cm
  4          100.0         100.0          0.0 # cm

### Points with fixed degrees of freedom and fixity codes (1-fixed;0-free)
# ivfix nofix      ifpre ...
  1      1      0 0 1
  2      2      1 1 1
  3      3      1 1 1
  4      4      1 1 1

### Sets of material properties
### (Young modulus, Poisson ratio, mass per unit volume and thermic coeff.)
# imats      young      poiss      dense      alpha
  1      20000.0      0.0      0.0      0.0 # kN/cm2
  2      20000.0      0.0      0.0      0.0 # kN/cm2

### Sets of element nodal properties
### (Cross section area)
# ispen
  1
# inode      barea
  1          1.5 # cm2
  2          1.5 # cm2

# =====

### Title of the load case n. 1
x: 10 kN ; y: 20 kN

### Load parameters
1 # nplod (n. of point loads in nodal points)
0 # ngrav (gravity load flag: 1=yes;0-no)
0 # nedge (n. of edge loads) (F.E.M. only)
0 # nface (n. of face loads) (F.E.M. only)
0 # ntemp (n. of points with temperature variation) (F.E.M. only)
0 # nudis (n. of uniformly distributed loads) (3d frames and trusses only)
0 # nepoi (n. of element point loads) (3d frames and trusses only)

```

```

0 # nprva (n. of prescribed and non zero degrees of freedom)

### Point loads in nodal points (loaded point and load value)
### (global coordinate system)
# iplod   lopop   pload-x pload-y pload-z
      1       1       10.0   20.0   0.0 # kN

# =====

### Title of the load case n. 2
x: 20 kN ; y: 10 kN

### Load parameters
1 # nplod (n. of point loads in nodal points)
0 # ngrav (gravity load flag: 1-yes;0-no)
0 # nedge (n. of edge loads) (F.E.M. only)
0 # nface (n. of face loads) (F.E.M. only)
0 # ntemp (n. of points with temperature variation) (F.E.M. only)
0 # nudis (n. of uniformly distributed loads) (3d frames and trusses only)
0 # nepoi (n. of element point loads) (3d frames and trusses only)
0 # nprva (n. of prescribed and non zero degrees of freedom)

### Point loads in nodal points (loaded point and load value)
### (global coordinate system)
# iplod   lopop   pload-x pload-y pload-z
      1       1       20.0   10.0   0.0 # kN

END_OF_FILE

```

Listagem 6.2 - Treliça com 3 barras - dados específicos do problema de otimização de estruturas.

```

### Main title (optimization)
2 materiais com custo distinto ; restricoes de area minima e de tensao maxima

### Main parameters (optimization)
1 # nmiac (minimum area constraints flag: 1-yes;0-no)
0 # nmaac (maximum area constraints flag: 1-yes;0-no)
1 # nproo (n. of material properties used in the formulation) (opt.)
6 # npreo (n. of element nodal properties used in the formulation) (opt.)
2 # nlcpr (n. of load case dep. material prop.s used in the formul.) (opt.)

### Sets of material properties (optimization)
### (Material cost per unit volume)
# imats   costm
      1     2.0 # cost units/cm3
      2     1.0 # cost units/cm3

### Sets of element nodal properties (optimization)
### (Minimum area, maximum area and buckling coefficients)
# ispen
      1
# inode  armin  armax  coef1  coef2  coef3  coef4
      1   0.15   0.0   0.0   0.0   0.0   0.0 # cm2
      2   0.15   0.0   0.0   0.0   0.0   0.0 # cm2

# =====

### Load case n. 1
Caso de carga n. 1: tensao limitada a 14 kN/cm2

### Constraint parameters
1 # nystc (yield stress constraints flag: 1-yes;0-no)
0 # nbstc (buckling stress constraints flag: 1-yes;0-no)
0 # ndcva (n. of nodal displacement constraints)

### Sets of load case dependent material properties (optimization)
### (Minimum stress and maximum stress) (only when nystc=1)

```

```

# imats      simin      simax
   1      -14.0       14.0 # kN/cm2
   2      -14.0       14.0 # kN/cm2

# =====

### Load case n. 2
Caso de carga n. 2: tensao limitada a 14 kN/cm2

### Constraint parameters
  1 # nystc (yield stress constraints flag: 1=yes;0=no)
  0 # nbstc (buckling stress constraints flag: 1=yes;0=no)
  0 # ndcva (n. of nodal displacement constraints)

### Sets of load case dependent material properties (optimization)
### (Minimum stress and maximum stress) (only when nystc=1)
# imats      simin      simax
   1      -14.0       14.0 # kN/cm2
   2      -14.0       14.0 # kN/cm2

END_OF_FILE

```

Listagem 6.3 - Programa matemático relativo à otimização de uma treliça simétrica com 3 barras e 2 casos de carga.

```

### Main title of the nonlinear program

Treliça simetrica com 3 barras e 2 casos de carga (kN,cm)

Min.

+ 565.68542494923804 * a1 + 100 * a2 ;

s.t.i.c.

1-Min. area 1:      - a1 + 0.15 < 0 ;
2-Min. area 2:      - a2 + 0.15 < 0 ;

3-S c.1_e.1_t.1:    - 100 * d_1_1 + 100 * d_1_2 - 14 < 0 ;
4-S c.1_e.1_t.2:    + 100 * d_1_1 - 100 * d_1_2 - 14 < 0 ;
5-S c.1_e.2_t.1:    + 200 * d_1_2 - 14 < 0 ;
6-S c.1_e.2_t.2:    - 200 * d_1_2 - 14 < 0 ;
7-S c.1_e.3_t.1:    + 100 * d_1_1 + 100 * d_1_2 - 14 < 0 ;
8-S c.1_e.3_t.2:    - 100 * d_1_1 - 100 * d_1_2 - 14 < 0 ;

9-S c.2_e.1_t.1:    - 100 * d_2_1 + 100 * d_2_2 - 14 < 0 ;
10-S c.2_e.1_t.2:   + 100 * d_2_1 - 100 * d_2_2 - 14 < 0 ;
11-S c.2_e.2_t.1:   + 200 * d_2_2 - 14 < 0 ;
12-S c.2_e.2_t.2:   - 200 * d_2_2 - 14 < 0 ;
13-S c.2_e.3_t.1:   + 100 * d_2_1 + 100 * d_2_2 - 14 < 0 ;
14-S c.2_e.3_t.2:   - 100 * d_2_1 - 100 * d_2_2 - 14 < 0 ;

s.t.e.c.

1-c.1_d.1:          + 141.42135623730951 * a1 * d_1_1 - 10 = 0 ;
2-c.1_d.2:          + 141.42135623730951 * a1 * d_1_2 + 200 * a2 * d_1_2 - 20 = 0 ;

3-c.2_d.1:          + 141.42135623730951 * a1 * d_2_1 - 20 = 0 ;
4-c.2_d.2:          + 141.42135623730951 * a1 * d_2_2 + 200 * a2 * d_2_2 - 10 = 0 ;

END_OF_FILE

```

Na Listagem 6.3 encontra-se o programa matemático que foi gerado a partir dos dados que se encontram nas Listagens 6.1 e 6.2. Nos títulos das restrições, S representa tensão, c caso de carga, e elemento/ barra, t tipo de restrição de tensão (1-compressão; 2-tracção) e d representa um deslocamento não impedido. As variáveis são os a_i e os d_{jk} , representando as primeiras a área da barra i e as segundas o deslocamento no caso de carga j segundo o grau de liberdade k . As restrições relativas a deslocamentos fixos e a restrição $a_3 = a_1$ encontram-se já substituídas, tendo dado origem à simplificação de algumas expressões e à eliminação da restrição de área mínima relativa à barra 3.

Neste exemplo é simples explicitar os deslocamentos em função das áreas recorrendo às equações de equilíbrio que figuram no programa matemático como restrições igualdade. A substituição dos deslocamentos nas restrições desigualdade faz com que o programa matemático passe a ter apenas duas variáveis (a_1 e a_2), possibilitando assim a representação gráfica da região admissível e da função objectivo. Na Figura 6.14 encontra-se representada, para cada restrição, a linha que separa o espaço admissível do inadmissível (funções $g(\tilde{x}) = 0$), indicando a seta o lado admissível. As restrições 6, 8, 9, e 12 encontram-se sempre inactivas quando a_1 e a_2 assumem valores positivos. Por este motivo, a respectiva isocurva $g(\tilde{x}) = 0$ não intersecta o domínio representado na figura. As linhas a tracejado que se encontram na Figura 6.14 correspondem a um valor constante da função objectivo ($f(\tilde{x}) = \text{const.}$). No domínio representado, o mínimo global é o ponto G e a região admissível possui dois vértices designados por L1 e L2. Neste exemplo simples foi possível calcular o valor exacto da solução G e as coordenadas dos vértices L1 e L2, encontrando-se esses valores no Quadro 6.3. A comparação dos resultados do método numérico com esta solução exacta permitiu avaliar a influência de diversos parâmetros na precisão dos resultados.

Quadro 6.3 - Optimização de uma treliça simétrica com 3 barras e 2 casos de carga - soluções exactas.

Solução	G	L1	L2
Função objectivo	$500\sqrt{14}/7 + 3750/7$	17000/21	7265/7
(U.C.)	802.975 527 627	809.523 809 524	1037.857 142 857
a_1	$5\sqrt{7}/49 + 5\sqrt{2}/7$	$20\sqrt{2}/21$	$179\sqrt{2}/140$
(cm ²)	1.280 127 168 130	1.346 870 059 403	1.808 173 054 748
a_2	$15\sqrt{14}/49 - 5/14$	10/21	15/100
(cm ²)	0.788 262 465 339	0.476 190 476 190	0.150 000 000 000

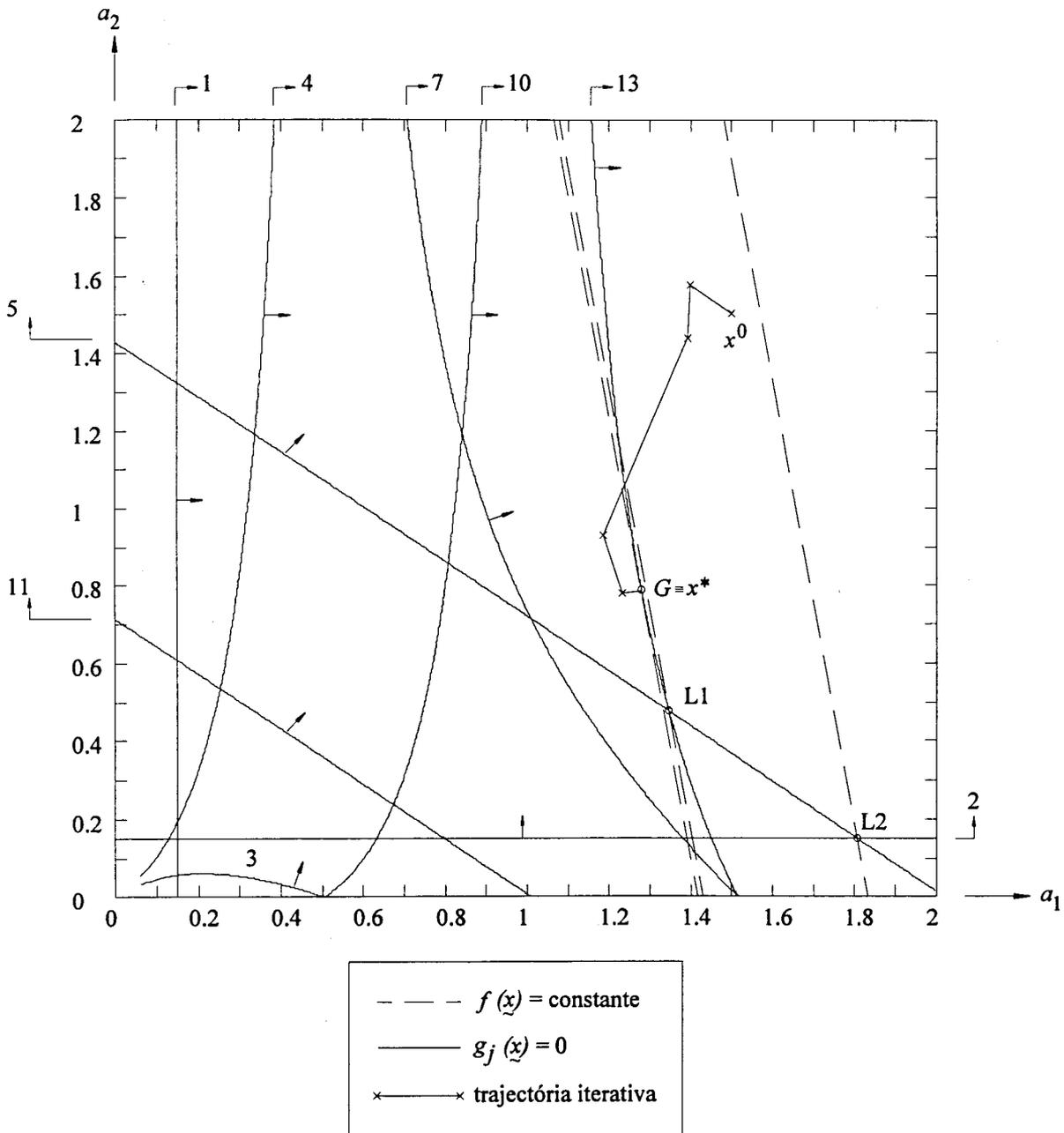


Figura 6.14 - Região admissível, função objectivo e trajetória iterativa relativa à optimização de uma treliça simétrica com 3 barras e 2 casos de carga.

Na Figura 6.14 encontra-se também representada a trajetória da solução desde o ponto inicial $x^0(a_1 = 1.5; a_2 = 1.5)$ até ao mínimo global (ponto G). Uma vez que na quase totalidade dos casos se obtém uma solução com elevada precisão, o facto de algumas soluções intermédias se encontrarem fora da região admissível não constitui qualquer inconveniente. Algumas soluções iniciais conduzem a uma convergência para um dos vértices da região admissível ($L1$ ou $L2$), porque estes são também pontos estacionários do Lagrangeano. No

entanto, verifica-se que neste exemplo os pontos estacionários distintos do mínimo global correspondem a soluções que apresentam pelo menos um multiplicador de Lagrange associado a restrições desigualdade negativo, não respeitando assim as condições de optimalidade de Karush-Kuhn-Tucker. O aparecimento de um ou mais multiplicadores de Lagrange negativos convida o utilizador do programa NEWTOP a recorrer a outra solução inicial.

Na Figura 6.15 encontra-se o painel de visualização do processo iterativo com os gráficos que mostram a evolução de alguns parâmetros até à solução final G e com o histograma que indica, para cada variável, o quociente entre o valor final e o inicial. Na optimização de treliças, e sempre que o processo iterativo converge para um mínimo local ou global, estes gráficos apresentam um aspecto semelhante aos da Figura 6.15. O parâmetro de pesquisa unidimensional (α) apresenta valores unitários nas primeiras e nas últimas iterações e valores consideravelmente inferiores na fase intermédia. A utilização de valores superiores à unidade raramente é vantajosa. No segundo gráfico pode-se observar que o resíduo equivalente é monotonamente decrescente. O patamar entre as iterações 7 e 9 é provocado pela estratégia de terminação do processo iterativo (ver secção 4.6). A função objectivo apresenta entre as iterações 3 e 5 valores inferiores ao da solução final. No entanto, tal como se pode observar na Figura 6.14, estes valores da função objectivo correspondem a soluções não admissíveis. No histograma da Figura 6.15 podem-se visualizar as variações das 6 variáveis de projecto (2 áreas e 2 deslocamentos livres em cada caso de carga), 14 variáveis de desvio, 14 multiplicadores de Lagrange associados a restrições desigualdade (λ^g) e 4 multiplicadores de Lagrange associados a restrições igualdade (λ^h). As 2 áreas das barras diminuíram e os 4 deslocamentos aumentaram. Só uma restrição desigualdade se encontra activa, uma vez que só uma variável de desvio é nula, correspondendo-lhe o único λ_j^g não nulo. Os λ_k^h relativos ao primeiro caso de carga são nulos, devido ao facto de nenhuma restrição desigualdade a ele associada se encontrar activa. Assim se conclui que se o primeiro caso de carga fosse suprimido, a solução do problema de optimização seria a mesma.

A observação durante o processo iterativo da informação que se encontra condensada nos três gráficos e no histograma da Figura 6.15, particularmente quando estão envolvidas no processo um número elevado de variáveis e restrições, permite extrair rapidamente conclusões sobre as causas de uma eventual ausência de convergência. O utilizador do programa é assim auxiliado na tomada de decisões relativas à alteração de estratégia que deve ser adoptada para solucionar o problema.

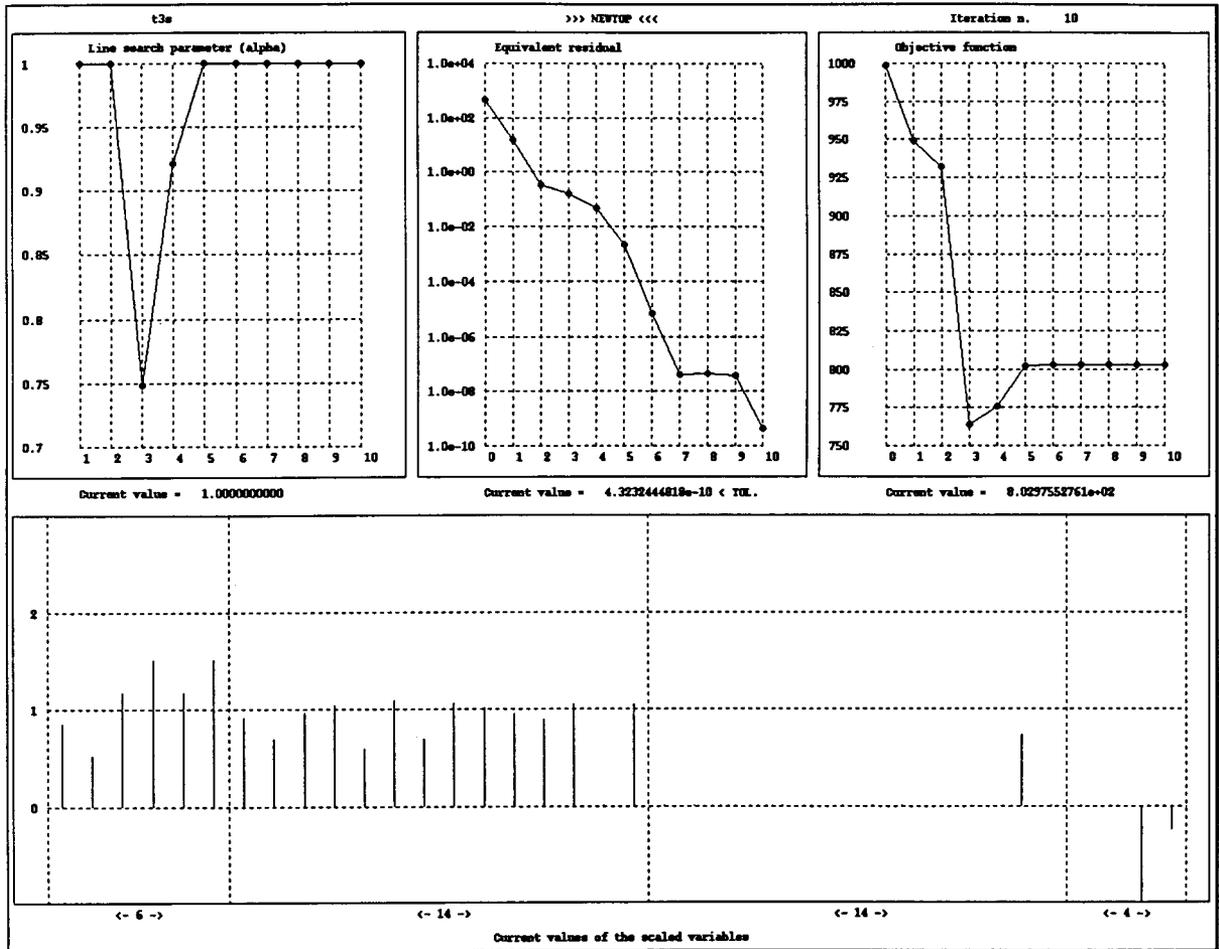


Figura 6.15 - Evolução de alguns parâmetros ao longo do processo iterativo e histograma com a variação da solução final em relação à inicial.

No Quadro 6.4 encontram-se os resultados obtidos com o programa NEWTOP a partir de três soluções iniciais distintas. Neste exemplo simples, essas soluções iniciais foram escolhidas de tal forma que o método iterativo convergisse para cada uma das distintas soluções finais (G, L1 e L2). A comparação dos valores numéricos que se encontram nos Quadros 6.3 e 6.4 permite avaliar a precisão da solução obtida com o programa NEWTOP. Na generalidade dos problemas, é possível obter uma solução com um erro inferior a 10^{-7} , verificando-se nestes casos que os valores numéricos do vector solução apresentam pelo menos 7 algarismos significativos correctos. Uma vez que o referido erro corresponde à norma do vector dos resíduos associados ao sistema de equações não lineares $\nabla L = 0$, em que L é o Lagrangeano do programa matemático após *scaling*, constatou-se que na generalidade dos problemas abordados o número de algarismos significativos correctos que se obtêm na

solução final é aproximadamente igual a $-\log_{10}$ (erro). Os resultados do programa NEWTOP (ver Quadro 6.4) são constituídos por cinco blocos: parâmetros globais (A), variáveis de projecto (B), folgas associadas às restrições desigualdade (C), multiplicadores de Lagrange das restrições desigualdade (D) e multiplicadores de Lagrange das restrições igualdade (E). No bloco C as folgas nulas indicam quais são as restrições desigualdade que se encontram activas em cada solução. A cada folga nula corresponde na generalidade dos casos um multiplicador de Lagrange não nulo. O facto de uma determinada restrição desigualdade apresentar um multiplicador de Lagrange negativo significa que se essa restrição fosse ligeiramente mais limitativa se obteria uma solução final admissível com um valor inferior da função objectivo. No caso da solução L1 (ver Quadro 6.4 e Figura 6.14) uma diminuição do valor absoluto da tensão admissível implicaria uma translação da linha 5 para cima, que faria com que ao ponto L1 passasse a corresponder um valor inferior da função objectivo. No caso da solução L2, um aumento do valor da área mínima dá origem a uma translação da linha 2 para cima, com uma consequente diminuição do valor da função objectivo associado ao ponto L2.

Quadro 6.4 - Optimização de uma treliça simétrica com 3 barras e 2 casos de carga - soluções obtidas com o programa NEWTOP.

	G	L1	L2
A) Função objectivo e erro:			
	8.02975526e+02	8.09523808e+02	1.03785714e+03
	2.05386941e-09	3.73838614e-08	2.54016825e-09
B) Variáveis de projecto:			
1-a1	1.280127166167	1.346870061363	1.808173052764
2-a2	0.788262464135	0.476190454035	0.149999999499
3-d_1_1	0.055237229540	0.052499999892	0.039106145294
4-d_1_2	0.059051082267	0.070000000611	0.070000000093
5-d_2_1	0.110474459081	0.104999999784	0.078212290588
6-d_2_2	0.029525541133	0.035000000305	0.035000000046
C) Folgas associadas às restrições desigualdade:			
1-Min. area 1	1.130127166167	1.196870061589	1.658173052764
2-Min. area 2	0.638262464135	0.326190466051	0.000000000000
3-S c.1_e.1_t.1	13.618614727347	12.249999936521	10.910614520106
4-S c.1_e.1_t.2	14.381385272652	15.750000078686	17.089385479893
5-S c.1_e.2_t.1	2.189783546592	0.000000000000	0.000000000000
6-S c.1_e.2_t.2	25.810216453407	28.000000131975	28.000000018647
7-S c.1_e.3_t.1	2.571168819244	1.749999969526	3.089385461246
8-S c.1_e.3_t.2	25.428831180755	26.250000051768	24.910614538753
9-S c.2_e.1_t.1	22.094891794751	20.999999951041	18.321229054198
10-S c.2_e.1_t.2	5.905108205248	7.000000062028	9.678770945801
11-S c.2_e.2_t.1	8.094891773296	6.999999948246	6.999999990676
12-S c.2_e.2_t.2	19.905108226703	21.000000064345	21.000000009323
13-S c.2_e.3_t.1	0.000000000000	0.000000000000	2.678770936478
14-S c.2_e.3_t.2	28.000000021455	28.000000008971	25.321229063521

Quadro 6.4 - Optimização de uma treliça simétrica com 3 barras e 2 casos de carga - soluções obtidas com o programa NEWTOP (cont.).

D) Multiplicadores de Lagrange das restrições desigualdade:

1-Min. area 1	0.000000027662	0.000000377282	-0.000000027479
2-Min. area 2	0.000000043319	0.000038417834	-700.000000000000
3-S c.1_e.1_t.1	-0.000000002994	0.000000091430	0.000000003618
4-S c.1_e.1_t.2	-0.000000002938	0.000000051239	-0.000000002380
5-S c.1_e.2_t.1	0.000000006592	-5.668939807967	81.632652843766
6-S c.1_e.2_t.2	-0.000000001765	0.000000026752	-0.000000001373
7-S c.1_e.3_t.1	0.000000007660	0.000000946625	0.000000008826
8-S c.1_e.3_t.2	0.000000001794	-0.000000011946	0.000000001569
9-S c.2_e.1_t.1	0.000000001981	0.000000025612	0.000000002108
10-S c.2_e.1_t.2	0.000000005011	-0.000000164941	-0.000000004264
11-S c.2_e.2_t.1	-0.000000004514	-0.000000204253	0.000000005854
12-S c.2_e.2_t.2	-0.000000002247	0.000000024333	0.000000001904
13-S c.2_e.3_t.1	57.355394654622	63.492067091961	0.000000008240
14-S c.2_e.3_t.2	0.000000001623	-0.000000001539	0.000000001522

E) Multiplicadores de Lagrange das restrições igualdade:

1-c.1_d.1	-0.000000006041	-0.000000336458	-0.000000006001
2-c.1_d.2	-0.000000003487	3.968257452248	-57.142857066746
3-c.2_d.1	-31.681530999244	-33.333335093159	-0.000000003000
4-c.2_d.2	-16.934490641039	-22.222223506368	-0.000000006447

6.3.3 - Treliças parametrizadas

Tendo em vista a avaliação da *performance* do programa NEWTOP na optimização de estruturas com um número de variáveis de projecto crescente, foi preparado um problema tipo baseado na minimização do volume de treliças. Este problema encontra-se definido de tal forma que se torna possível aumentar indefinidamente o número de barras e de graus de liberdade da treliça mantendo-se as características gerais, quer dos dados, quer da solução final. A regularidade deste conjunto de treliças torna possível uma extrapolação das suas características variáveis, podendo-se assim prever tempos de resolução e parâmetros associados à solução final de problemas que não podem presentemente ser resolvidos devido às limitações dos meios informáticos disponíveis. O estudo deste conjunto de exemplos contribuiu para a identificação dos aspectos que exigem mais recursos informáticos e que serão prioritariamente merecedores de futuros aperfeiçoamentos. As conclusões desse estudo permitem ainda avaliar qual a maior dimensão dos problemas que podem ser resolvidos em computadores com recursos mais limitados.

6.3.3.1 - Formulação geral

O conjunto de problemas de otimização utilizado na avaliação das capacidades do programa NEWTOP foi preparado de acordo com os seguintes princípios:

- facilidade de publicação de todos os dados de modo a permitir que outros investigadores possam reproduzir os resultados aqui divulgados;
- formulação geral simples;
- estrutura com comportamento linear;
- possibilidade de se poder gerar um problema com um número de variáveis e restrições indefinidamente grande;
- facilidade de geração dos dados a partir de um reduzido número de parâmetros;
- ausência de simplificações devidas à existência de planos de simetria;
- ausência de ligações entre variáveis de projecto (*variable linking*);
- inclusão de restrições de tensão, deslocamento e área mínima;
- pelo menos uma restrição de cada tipo activa na solução final.

Tendo em vista estes objectivos, é proposto o seguinte problema:

- Minimizar o volume de uma treliça espacial destinada a cobrir uma área $l_x \times l_y$, com as características indicadas na Figura 6.16.

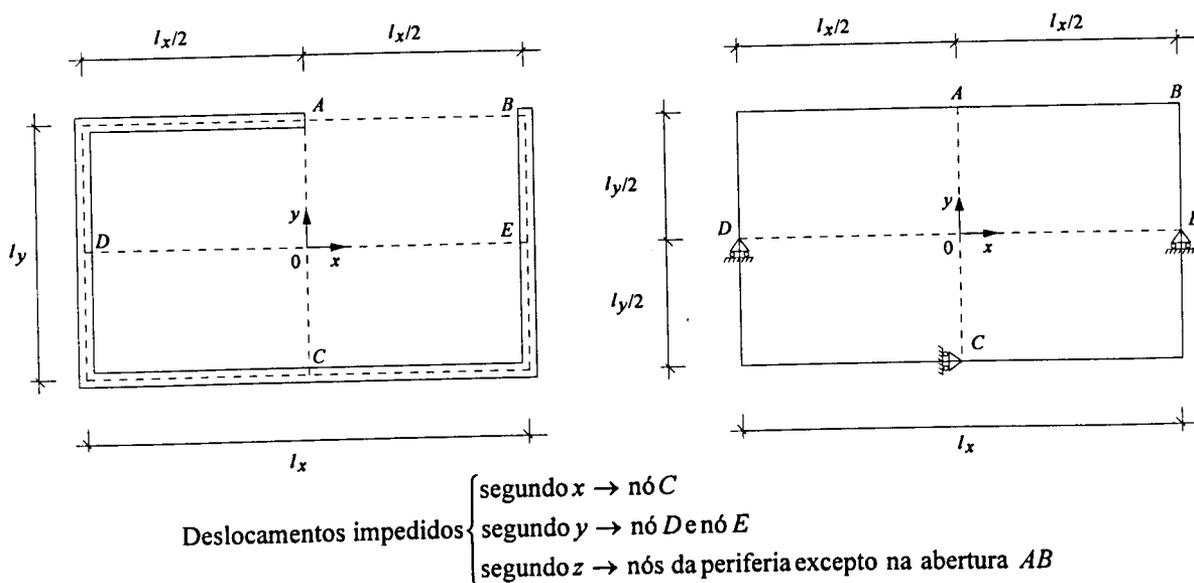


Figura 6.16 - Características gerais do problema de otimização parametrizado.

A topologia de cada treliça é fixa, assim como as respectivas coordenadas dos nós, e consiste numa repetição dos módulos representados na Figura 6.17.

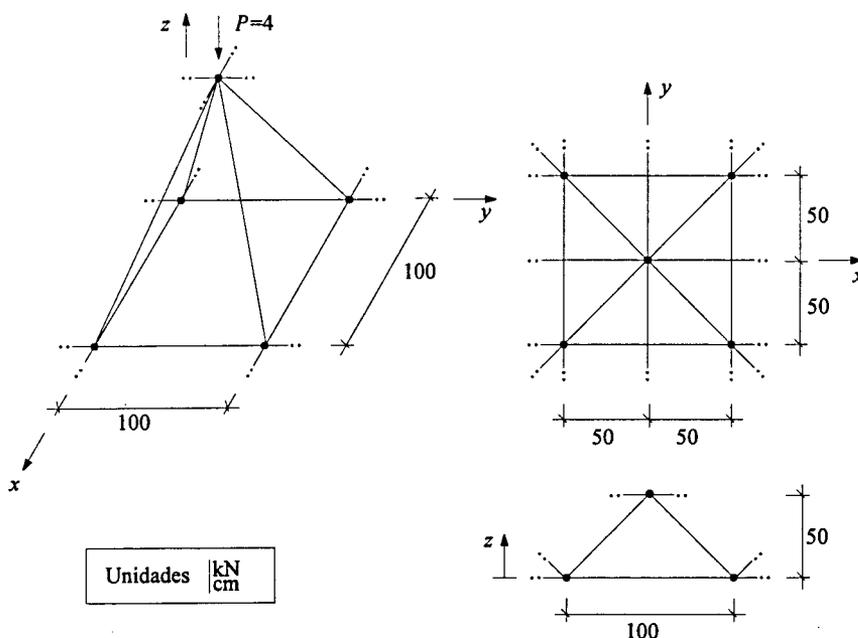


Figura 6.17 - Topologia das treliças utilizadas no problema de optimização parametrizado.

No Quadro 6.5 encontram-se as restantes características gerais que definem o problema de optimização parametrizado.

Quadro 6.5 - Características gerais do problema de optimização parametrizado.

l_x/l_y	2
l_y	múltiplo de 100 cm
Módulo de elasticidade	20 000 kN/cm ²
Acção em cada nó com $z = 50$ cm	4 kN segundo z negativo
Tensão máxima em cada barra	14 kN/cm ²
Tensão mínima em cada barra	-14 kN/cm ²
Deslocamento vertical limitado nos nós livres com $z = 0$ ①	
Restrições de área mínima em todas as barras ①	
Não é considerado o peso próprio	
Não são consideradas restrições de encurvadura	

① O valor limite depende da dimensão da treliça (ver Quadro 6.8).

De acordo com a formulação geral atrás descrita, foram preparados os dados relativos a oito treliças designadas genericamente por $t l_y \times l_x$, com l_x e l_y em metros e com l_y a percorrer os números pares do intervalo [2,16]. As topologias da menor e da maior treliça deste conjunto encontram-se representadas na Figura 6.18. No Quadro 6.6 encontram-se os principais parâmetros, que dependem aproximadamente do número de barras da treliça. A geração do programa matemático foi efectuada de acordo com a formulação descrita na secção 6.3.2, tendo as equações de equilíbrio sido consideradas como restrições igualdade. O principal parâmetro que condiciona o maior problema que pode ser resolvido num determinado computador é o que quantifica a memória central ocupada durante a resolução do sistema de equações lineares.

$$\tilde{H} \Delta \tilde{X} + \nabla L = 0 \quad (6.55)$$

Considerando a matriz Hessiana dividida em submatrizes e efectuando a eliminação de Gauss com o algoritmo descrito no Capítulo 4, verifica-se que a fase que se torna predominante, quer em tempo de resolução, quer em memória ocupada, é a resolução de um sistema de equações lineares com a matriz simétrica e de dimensão $nt \times nt$ (ver Quadro 6.6). Atendendo à grande quantidade de memória central que se encontra disponível nos computadores mais recentes, verifica-se que se torna inoportável resolver problemas que recorram a sistemáticas trocas de informação entre a memória central e a memória de massa. Por este motivo considera-se que o maior problema que pode ser resolvido num determinado computador é aquele que ocupa aproximadamente a totalidade da memória central. Do ponto de vista do tempo de resolução, admite-se como máximo período de espera pelos resultados um reduzido número de dias. De acordo com os parâmetros que se encontram no Quadro 6.6 e uma vez que foi utilizado um computador com 256 MBytes de memória central, o maior problema passível de ser abordado foi o $t14 \times 28$.

Tendo em vista uma redução da quantidade de memória central necessária e um conseqüente aumento do número de variáveis do maior problema que se consegue resolver, foi desenvolvido um modo alternativo de geração do programa matemático relativo à optimização de treliças. A alteração introduzida consiste na geração de pares de restrições desigualdade em substituição de cada restrição igualdade.

$$h = 0 \rightarrow \begin{cases} h < 0 \\ -h < 0 \end{cases} \quad (6.56)$$

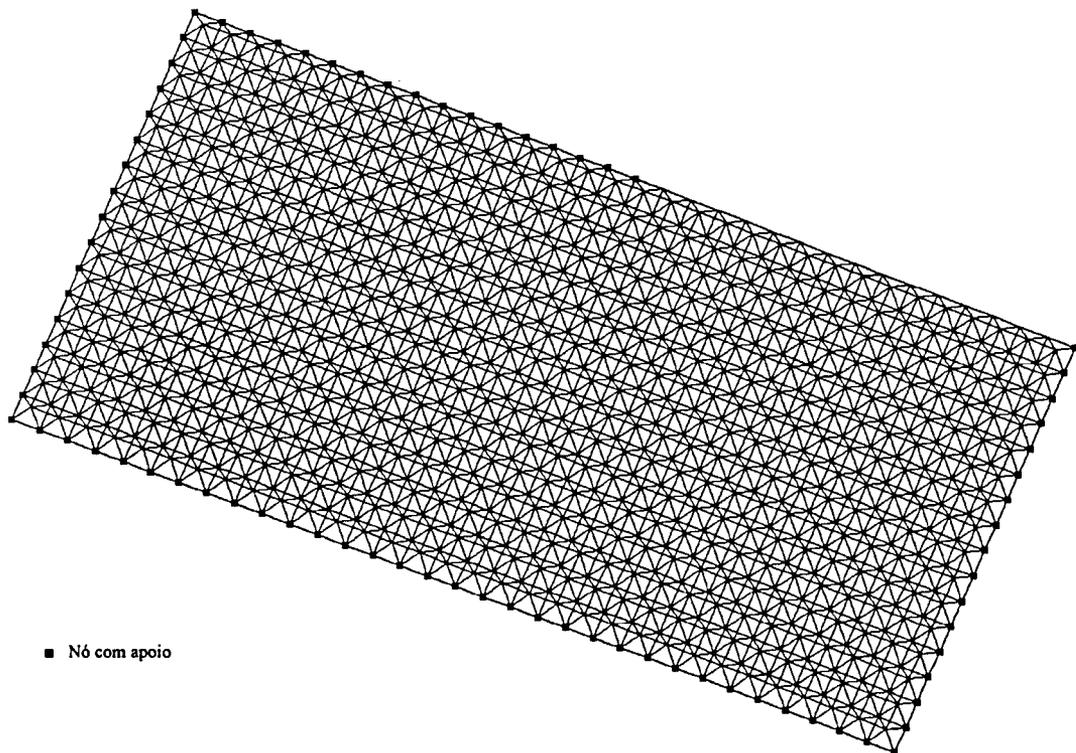
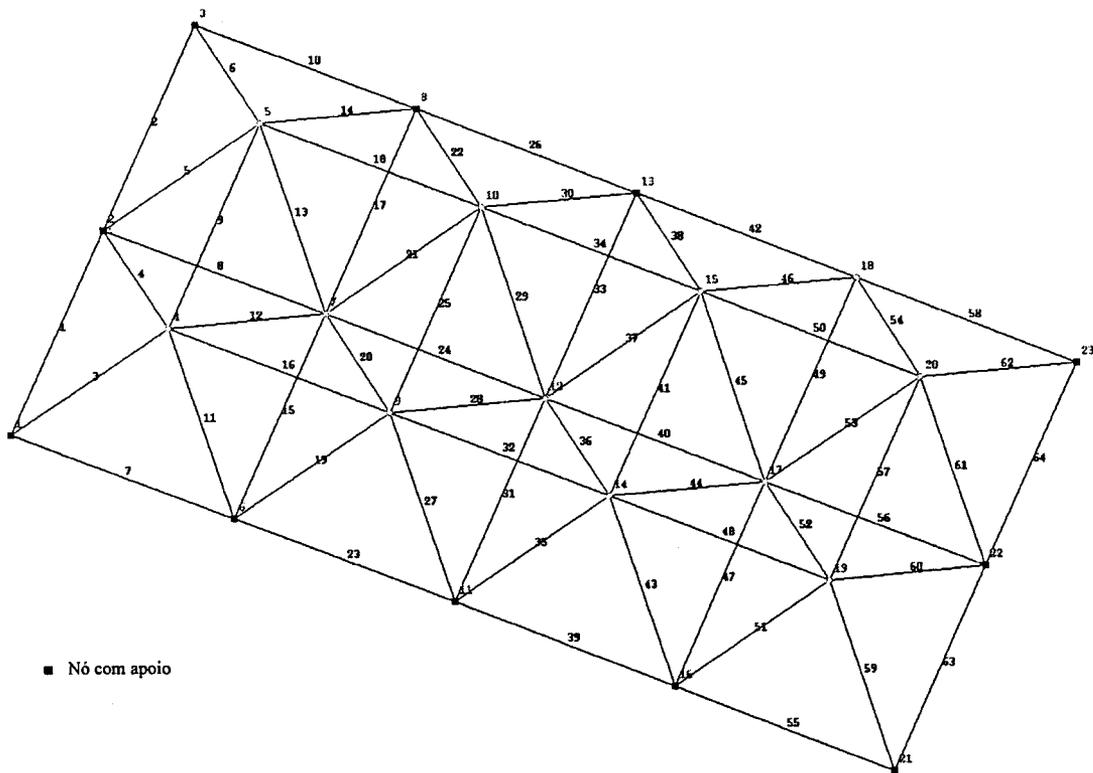


Figura 6.18 - Topologia das treliças 2×4 e 16×32 .

Quadro 6.6 - Treliças parametrizadas - parâmetros dependentes da dimensão da treliça (formulação com restrições igualdade).

Treliça ly x lx	NB	NN	NL	NTGL	NGL	ND	n	m	l	ne	nt	RAM (MB)
t2x4	64	23	11	69	55	4	119	196	55	566	174	0.122
t4x8	256	77	21	231	207	24	463	792	207	2254	670	1.798
t6x12	576	163	31	489	455	60	1031	1788	455	5062	1486	8.839
t8x16	1024	281	41	843	799	112	1823	3184	799	8990	2622	27.510
t10x20	1600	431	51	1293	1239	180	2839	4980	1239	14038	4078	66.537
t12x24	2304	613	61	1839	1775	264	4079	7176	1775	20206	5854	137.101
t14x28	3136	827	71	2481	2407	364	5543	9772	2407	27494	7950	252.842
t16x32	4096	1073	81	3219	3135	480	7231	12768	3135	35902	10366	429.857

Simbologia:

NB	- n. de barras
NN	- n. de nós
NL	- n. de nós com ligação ao exterior
NTGL	- n. total de graus de liberdade
NGL	- n. de graus de liberdade não impedidos
ND	- n. de deslocamentos restringidos
n	- n. de variáveis de projecto
m	- n. de restrições desigualdade
l	- n. de restrições igualdade
ne	- n. de equações não lineares no sistema $\nabla L = 0$
nt	- n. de equações lineares na fase que é resolvida com a matriz triangular superior
RAM	- memória ocupada pelo triângulo superior da matriz $nt \times nt$ (MBytes)

Observações:

NTGL	= 3*NN
NGL	= NTGL-NL-3
n	= NB+NGL
m	= 3*NB+ND
l	= NGL
ne	= n+2*m+l
nt	= n+l
RAM	= 8*[nt*(nt+1)/2] (supondo sizeof(double)=8)

A vantagem deste procedimento baseia-se no facto de o programa NEWTOP ser pouco penalizado com o aumento do número de restrições desigualdade (ver Capítulo 4). A redução do número de restrições igualdade alivia significativamente a fase crítica do algoritmo, que é a resolução do sistema de equações $nt \times nt$. No Quadro 6.7 são apresentados os parâmetros

relativos a esta formulação alternativa. Uma vez que a necessidade de memória passa a ser cerca de metade, torna-se possível abordar problemas da ordem de grandeza do $t16 \times 32$. A comparação entre os resultados e as *performances* destas duas formulações alternativas será adiante apresentada.

Quadro 6.7 - Treliças parametrizadas - parâmetros dependentes da dimensão da treliça (formulação sem restrições igualdade).

Treliça $ly \times lx$	NB	NN	N L	NTGL	NGL	ND	n	m	l	ne	nt	RAM (MB)
t2x4	64	23	11	69	55	4	119	306	0	731	119	0.057
t4x8	256	77	21	231	207	24	463	1206	0	2875	463	0.859
t6x12	576	163	31	489	455	60	1031	2698	0	6427	1031	4.256
t8x16	1024	281	41	843	799	112	1823	4782	0	11387	1823	13.301
t10x20	1600	431	51	1293	1239	180	2839	7458	0	17755	2839	32.251
t12x24	2304	613	61	1839	1775	264	4079	10726	0	25531	4079	66.569
t14x28	3136	827	71	2481	2407	364	5543	14586	0	34715	5543	122.922
t16x32	4096	1073	81	3219	3135	480	7231	19038	0	45307	7231	209.178

Simbologia:

- NB - n. de barras
- NN - n. de nós
- NL - n. de nós com ligação ao exterior
- NTGL - n. total de graus de liberdade
- NGL - n. de graus de liberdade não impedidos
- ND - n. de deslocamentos restringidos
- n - n. de variáveis de projecto
- m - n. de restrições desigualdade
- l - n. de restrições igualdade
- ne - n. de equações não lineares no sistema $\nabla L = 0$
- nt - n. de equações lineares na fase que é resolvida com a matriz triangular superior
- RAM - memória ocupada pelo triângulo superior da matriz $nt \times nt$ (MBytes)

Observações:

- NTGL = $3 \cdot NN$
- NGL = $NTGL - NL - 3$
- n = $NB + NGL$
- m = $3 \cdot NB + ND + 2 \cdot NGL$
- l = 0
- ne = $n + 2 \cdot m + l$
- nt = $n + l$
- RAM = $8 \cdot [nt \cdot (nt + 1) / 2]$ (supondo $\text{sizeof}(\text{double}) = 8$)

6.3.3.2 - Preparação da solução inicial

Uma vez que se pretendia estudar o comportamento do programa NEWTOP na optimização de um conjunto de treliças de características semelhantes e com um número de barras crescente, foi adoptado um critério constante na geração da respectiva solução inicial. Na generalidade dos casos consegue-se uma maior fiabilidade e velocidade de convergência quando a solução inicial apresenta as seguintes características:

- todas as restrições igualdade e desigualdade são respeitadas (solução admissível);
- todas as restrições desigualdade estão inactivas e com uma folga significativa;
- o valor inicial das variáveis de projecto é da mesma ordem de grandeza do correspondente à solução final.

Tendo em vista a obtenção de uma solução inicial com estas características, foi efectuado um cálculo preliminar da solução FSD (*Fully Stressed Design*), recorrendo ao algoritmo descrito na secção 6.1.1. Em todos os casos foram realizadas 100 iterações a partir de uma solução com 1000 cm^2 em todas as barras. Este procedimento conduziu sempre a uma solução FSD com uma precisão aceitável, tendo em conta que ainda não se trata de uma solução final.

A solução FSD serviu também para determinar o valor da área mínima e do deslocamento máximo a utilizar na optimização das diversas treliças. Pretendia-se que na solução final estivesse activa pelo menos uma restrição de tensão, uma restrição de deslocamento e uma restrição de área mínima. No Quadro 6.8 encontram-se alguns parâmetros relativos à solução FSD e os valores adoptados em cada treliça como área mínima e deslocamento máximo. Relativamente a estes dois parâmetros são também apresentadas as expressões que conduziram aos respectivos valores. Presume-se que estas expressões continuam a fornecer valores nas condições atrás referidas para treliças com um número de barras superior ao da $t16 \times 32$.

Quadro 6.8 - Treliças parametrizadas - solução FSD e parâmetros dela dependentes.

Treliça ly × lx	NB	min_a (cm ²)	FSD_e (kN/cm ²)	FSD_f (cm ³)	FSD_a (cm ²)	FSD_d (cm)	max_d (cm)	KFSD'	KFSD
t2x4	64	0.125	1.0e-9	1072	0.539	0.391	0.3	1.303	1.5
t4x8	256	0.250	1.0e-9	10246	2.611	1.088	0.6	1.813	2.0
t6x12	576	0.500	7.0e-5	46045	6.472	2.238	0.9	2.487	3.0
t8x16	1024	1.000	5.0e-8	148438	12.051	3.660	1.2	3.050	3.5
t10x20	1600	2.000	1.0e-6	406672	20.557	5.355	1.5	3.570	4.0
t12x24	2304	4.000	2.0e-5	1030111	33.785	7.148	1.8	3.971	5.0
t14x28	3136	8.000	1.0e-4	2549969	52.334	8.571	2.1	4.081	5.0
t16x32	4096	16.000	5.0e-4	6280694	72.322	9.636	2.4	4.015	5.0

Nota:

No cálculo da solução FSD não foram consideradas as restrições de deslocamento.

Simbologia:

- NB - número de barras
- min_a - mínima área admissível
- FSD_e - erro da solução FSD (maior violação de restrições de tensão em valor absoluto)
- FSD_f - valor da função objectivo para a solução FSD
- FSD_a - área da maior secção transversal da solução FSD
- FSD_d - maior deslocamento vertical para a solução FSD (valor absoluto)
- max_d - máximo deslocamento vertical admissível (valor absoluto)
- KFSD' - menor factor pelo qual se devem multiplicar as áreas das barras para eliminar as violações de restrições deslocamento (=FSD_d/max_d)
- KFSD - valor adoptado de modo a não haver restrições deslocamento violadas nem activas

Observações:

$\min_a = 2^{(ly/2-4)}$ (com ly em metros e min_a em cm²)

$\max_d = 0.15 \cdot ly$ (com ly em metros e max_d em cm)

Apresenta-se em seguida o conjunto de procedimentos adoptado no cálculo da solução inicial que vai ser utilizada pelo programa NEWTOP:

- cálculo da solução FSD com as áreas mínimas indicadas no Quadro 6.8 e sem considerar as restrições de deslocamento. Em todos os casos devem ser realizadas 100 iterações a partir de uma solução inicial constante com as áreas de todas as barras iguais 1000 cm²;
- determinação do maior deslocamento vertical (em valor absoluto);

- cálculo do factor K_{FSD}' como sendo o quociente entre o maior deslocamento da solução FSD e o deslocamento admissível;
- selecção de um valor para K_{FSD} ligeiramente maior do que K_{FSD}' ;
- multiplicação das áreas de todas as barras por K_{FSD} .

O tempo de CPU necessário à obtenção da solução inicial por intermédio deste procedimento é muito inferior ao dedicado à fase de optimização com o programa NEWTOP. Em exemplos com outras características este procedimento poderá ter de ser sujeito a algumas adaptações. Referem-se, por exemplo, treliças em que o deslocamento admissível não apresente um valor constante, treliças com vários casos de carga, treliças cuja solução FSD não viola nenhuma restrição deslocamento, etc. Convém por fim salientar o facto de a solução FSD multiplicada por K_{FSD} não ser garantidamente uma boa solução inicial para o programa NEWTOP. O utilizador deve portanto admitir a hipótese de ter de utilizar uma solução inicial obtida de um modo distinto do aqui sugerido.

6.3.3.3 - Apresentação dos resultados

Para passar da solução inicial para a solução óptima recorrendo ao programa NEWTOP, foi inicialmente adoptada uma única estratégia para a totalidade das treliças. As opções que mais condicionam a trajectória iterativa começaram por ser as seguintes:

- equações de equilíbrio como restrições igualdade;
- valor inicial de todos os multiplicadores de Lagrange = 1.0;
- menor valor que uma variável pode assumir durante o processo iterativo = 10^{-5} ;
- maior valor admissível do parâmetro de pesquisa unidimensional = 1.0;
- resolução do sistema de equações $\underline{H} \Delta \underline{X} + \nabla L = \underline{0}$ pelo método de eliminação de Gauss com a matriz simétrica e com pivotagem na diagonal.

Nota: as treliças $t14 \times 28$ e $t16 \times 32$ apenas foram abordadas com as equações de equilíbrio como pares de restrições desigualdade, pelo facto de não se dispor de memória central suficiente para a resolução com restrições igualdade.

Com estas opções e com a solução inicial sugerida na secção 6.3.3.2 foi possível obter um ponto estacionário do Lagrangeano para as treliças $t2 \times 4$, $t4 \times 8$ e $t12 \times 24$. Nos restantes casos, para obter uma primeira solução foi necessário alterar algumas das opções e/ou reiniciar o processo iterativo a partir de uma solução intermédia. Este procedimento faz com que os coeficientes associados ao *scaling* sejam recalculados com base numa solução mais próxima da solução final. A observação do comportamento do processo iterativo nos casos em que com as opções *standard* não se consegue convergência fornece em geral indicações que auxiliam o utilizador a decidir quais as alterações que devem ser efectuadas. Nas treliças $t6 \times 12$ e $t8 \times 16$ foi mais fácil obter uma solução final partindo de uma solução inicial constante.

Na impossibilidade de publicar a totalidade dos resultados devido à sua extensão, apresentam-se no Quadro 6.9 alguns parâmetros associados às soluções obtidas com o programa NEWTOP. O primeiro grupo de parâmetros foram extraídos da primeira solução que foi obtida nas condições atrás referidas. Exceptuando os casos das treliças $t4 \times 8$ e $t10 \times 20$, a primeira solução obtida apresentou alguns multiplicadores de Lagrange negativos (ver Quadro 6.9). As soluções deste tipo, apesar de admissíveis, não respeitam as condições de Karush-Kuhn-Tucker e, por esse motivo, não são sequer mínimos locais. O segundo grupo de parâmetros baseia-se na solução que respeita as condições de Karush-Kuhn-Tucker e que foi obtida após várias tentativas com diversas soluções iniciais e com estratégias distintas das habitualmente utilizadas. Nos casos em que a solução KKT é distinta da primeira solução obtida, verifica-se que a diferença entre elas é inferior a 0.1%. Por este motivo, na generalidade dos casos não se justifica o esforço dispendido na obtenção da solução KKT, sendo aceitável considerar a primeira solução como solução óptima.

Com base nos valores que se encontram no Quadro 6.9, pode-se concluir que o número de iterações efectuadas pelo programa NEWTOP não aumenta significativamente com o número de variáveis do problema. Exceptuando situações anómalas, neste tipo de problemas de optimização o número de iterações efectuadas encontra-se compreendido entre 15 e 30. Apesar de apenas se ter exigido um erro inferior a 10^{-5} , foi possível obter na maior parte dos casos soluções com um erro inferior a 10^{-8} .

Quadro 6.9 - Treliças parametrizadas - soluções obtidas com o programa NEWTOP.

Dados		Primeira solução obtida					Solução sem M.L. negativos		
Treliça ly x lx	NB	NI	NEWT_e	NEWT_f (cm3)	NEWT_a (cm2)	NML	NEWT_e	NEWT_f (cm3)	NEWT_a (cm2)
t2x4	64	18	1e-9	1135.3370	0.643	3	1e-9	1134.8193	0.620
t4x8	256	17	2e-9	14452.653	3.605	0	←	←	←
t6x12	576	24	6e-7	82872.083	10.469	5	5e-9	82868.016	10.464
t8x16	1024	23	6e-9	304297.80	27.090	1	1e-6	303982.41	26.671
t10x20	1600	32	5e-9	868994.44	62.623	0	←	←	←
t12x24	2304	28	8e-7	2138254.3	112.944	2	?	?	?
t14x28 *	3136	33	4e-4	4761888.1	220.722	4	?	?	?
t16x32 *	4096	38	2e-5	10173799.	388.658	7	?	?	?

Simbologia:

- * - problemas resolvidos com as equações de equilíbrio como pares de restrições desigualdade
- M.L. - multiplicadores de Lagrange das restrições desigualdade
- NB - número de barras
- NI - número de iterações
- NEWT_e - erro da solução NEWTOP (com scaling)
- NEWT_f - valor da função objectivo para a solução NEWTOP
- NEWT_a - área da maior secção transversal da solução NEWTOP
- NML - número de multiplicadores de Lagrange negativos em restrições desigualdade
- ← - a primeira solução obtida já não apresentava multiplicadores de Lagrange negativos em restrições desigualdade

O quociente entre a área da maior secção transversal e a mínima área admissível situa-se entre 15 e 30, exceptuando a treliça t2x4 que devido às suas reduzidas dimensões apresenta um valor inferior. Em certos casos verifica-se que quando este quociente apresenta valores elevados, torna-se mais difícil conseguir a convergência para uma solução admissível. Nestes casos é mais fácil chegar a uma solução executando duas ou três vezes o programa NEWTOP com valores decrescentes da mínima área admissível. Na Figura 6.19 é evidenciado o crescimento regular com o número de barras do valor da função objectivo e da área da maior secção transversal.

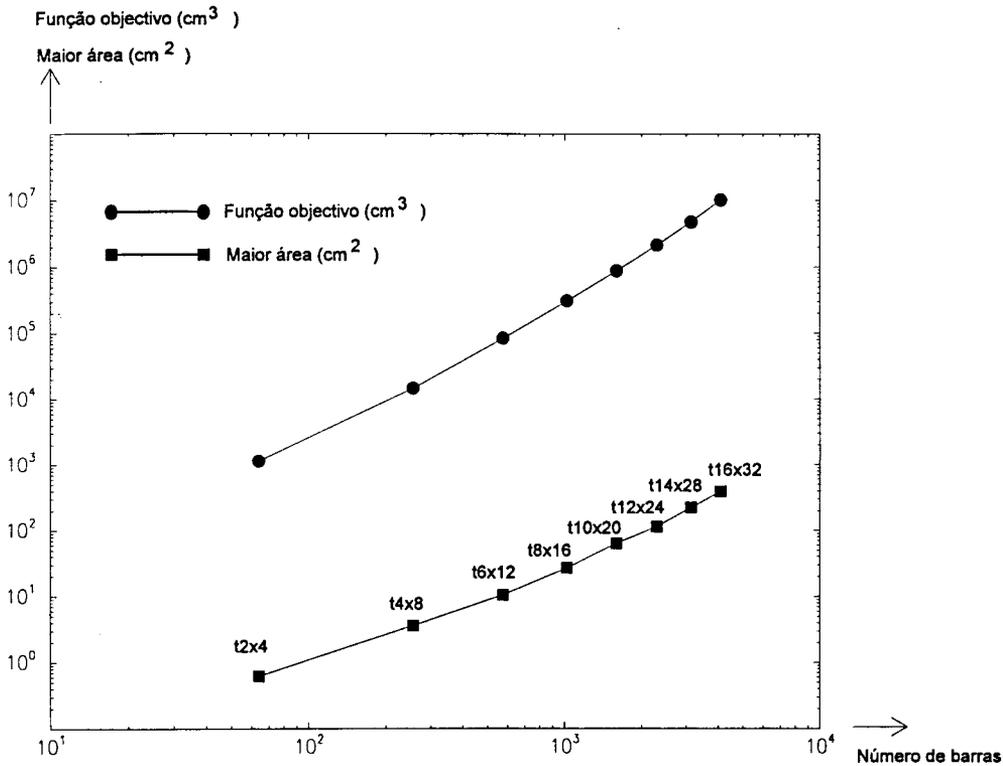


Figura 6.19 - Treliças parametrizadas - solução NEWTOP - valor da função objectivo e da área da maior secção transversal.

Durante a fase iterativa é possível visualizar graficamente a evolução de um conjunto de parâmetros e a solução corrente. A elevada quantidade de informação que é condensada num único gráfico, vem facilitar a tomada de decisões sobre eventuais intervenções destinadas a alterar o comportamento do algoritmo. Quando o número de variáveis é elevado as vantagens desta visualização gráfica tornam-se ainda mais importantes. As treliças parametrizadas apresentam, de um modo geral, um comportamento ao longo do processo iterativo semelhante ao representado na Figura 6.20. Nessa figura, que corresponde à treliça $t8 \times 16$, pode observar-se que o parâmetro de pesquisa unidimensional apresenta valores unitários na primeira e nas últimas iterações. O parâmetro associado ao erro da solução é monotonamente decrescente e apresenta na última iteração uma diminuição para um valor quase 100 000 vezes inferior. A elevada precisão da solução final é consequência do facto de o método de Newton apresentar convergência quadrática. Antes de atingir o valor final a função objectivo apresenta valores inferiores, que não correspondem a soluções admissíveis. O comportamento da totalidade das variáveis pode ser observado qualitativamente no histograma representado na metade inferior da Figura 6.20. Nesta figura apenas se encontram os valores correspondentes à solução final. O programa de computador que efectua a representação gráfica da solução é interactivo e permite ao utilizador uma rápida comutação

entre as soluções correspondentes às diversas iterações. Nos terminais gráficos de maior *performance* consegue-se mesmo um efeito de animação, desde que o número de variáveis não seja muito elevado. No histograma da Figura 6.20 encontram-se representados os valores após *scaling* das seguintes variáveis:

- 1024 áreas de barras;
- 799 deslocamentos de nós;
- 3184 variáveis de desvio;
- 3184 multiplicadores de Lagrange de restrições desigualdade (λ^g);
- 799 multiplicadores de Lagrange de restrições igualdade (λ^h).

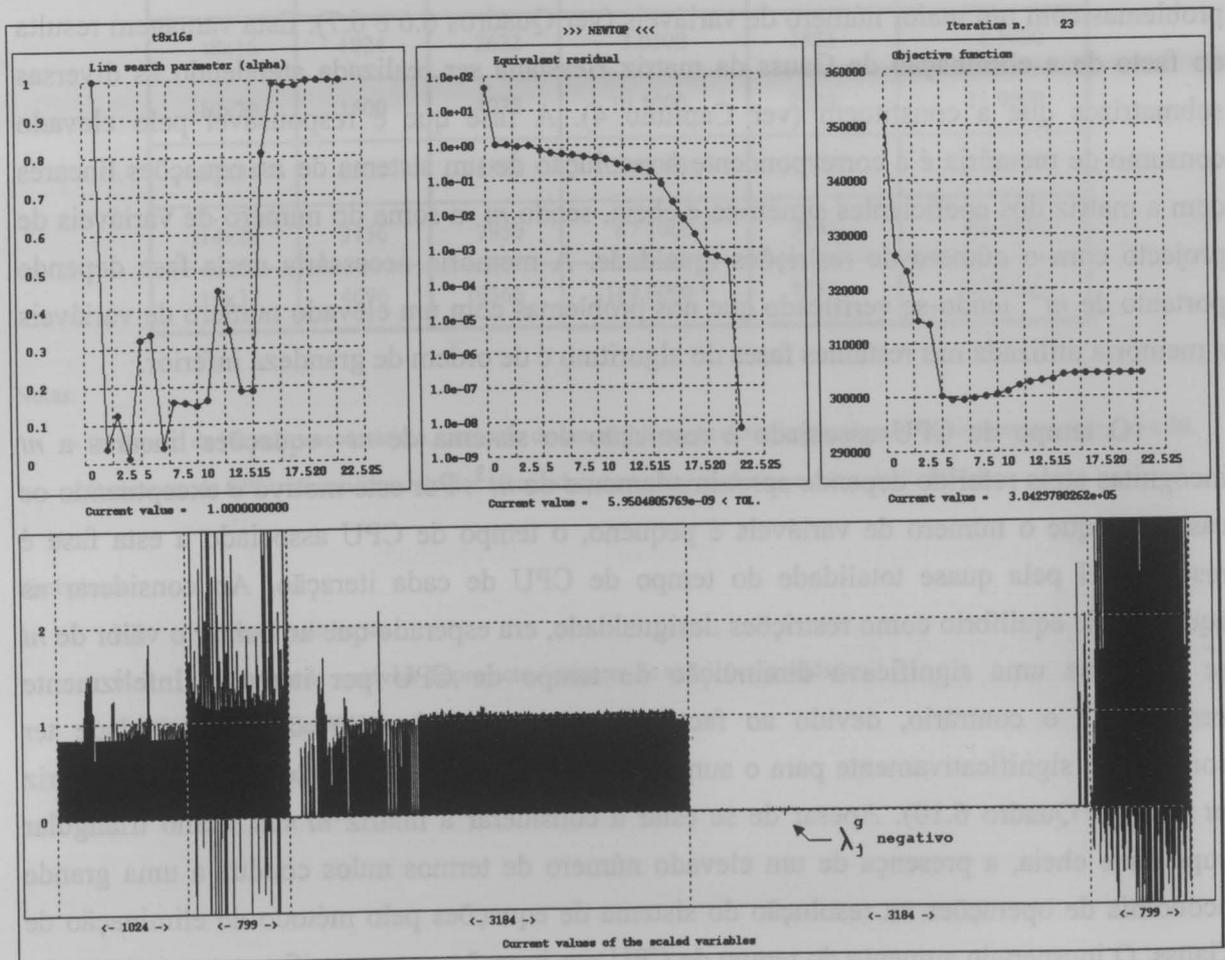


Figura 6.20 - Treliça $t8 \times 16$ - visualização da evolução de alguns parâmetros e histograma com a solução final.

Encontra-se assinalada na Figura 6.20 a presença de um λ_j^g negativo. Este facto indica que existe pelo menos uma solução admissível de custo inferior ao da solução representada na figura.

Formulação sem restrições igualdade

Conforme foi atrás referido, foram testadas duas alternativas na geração do programa matemático relativo à optimização de treliças. Na primeira, as equações de equilíbrio são consideradas como restrições igualdade e na segunda são decompostas em pares de restrições desigualdade, de acordo com (6.56). Atendendo apenas às características do método de Newton, a segunda formulação nunca apresentaria vantagens porque conduz a um aumento significativo do número de restrições, do número de primeiras e segundas derivadas a calcular e do número de equações e incógnitas presentes no sistema de equações que é necessário resolver em cada iteração. No entanto, ao considerar as características do programa NEWTOP, verifica-se que a formulação sem restrições igualdade é capaz de abordar problemas com um maior número de variáveis (ver Quadros 6.6 e 6.7). Esta vantagem resulta do facto de a eliminação de Gauss da matriz Hessiana ser realizada atendendo às diversas submatrizes que a constituem (ver Capítulo 4). A fase que é responsável pelo elevado consumo de memória é a correspondente à resolução de um sistema de nt equações lineares com a matriz dos coeficientes simétrica e cheia, sendo nt a soma do número de variáveis de projecto com o número de restrições igualdade. A memória necessária nesta fase depende portanto de nt^2 , tendo-se verificado que nos problemas com um elevado número de variáveis a memória utilizada nas restantes fases do algoritmo é de ordem de grandeza inferior.

O tempo de CPU associado à resolução do sistema de nt equações lineares a nt incógnitas atrás referido depende aproximadamente de nt^3 . Por este motivo e exceptuando os casos em que o número de variáveis é pequeno, o tempo de CPU associado a esta fase é responsável pela quase totalidade do tempo de CPU de cada iteração. Ao considerar as equações de equilíbrio como restrições desigualdade, era esperado que ao baixar o valor de nt se obtivesse uma significativa diminuição do tempo de CPU por iteração. Infelizmente verificou-se o contrário, devido ao facto de o aumento de restrições desigualdade ter contribuído significativamente para o aumento do número de termos não nulos na submatriz $nt \times nt$ (ver Quadro 6.10). Apesar de se estar a considerar a matriz $nt \times nt$ como triangular superior e cheia, a presença de um elevado número de termos nulos conduz a uma grande economia de operações na resolução do sistema de equações pelo método de eliminação de Gauss. O inesperado aumento do tempo de CPU por iteração que se verificou ao passar para a formulação sem restrições igualdade é também consequência do aumento do *fill-in* provocado pelo referido aumento no número de termos não nulos da matriz dos coeficientes. A influência

da esparsidade desta matriz e do respectivo *fill-in* no tempo de CPU por iteração é difícil de prever e depende muito das características do problema que se está a resolver. No caso das treliças parametrizadas, o facto de as suas características serem aproximadamente constantes faz com que a evolução do tempo de CPU por iteração com o número de barras apresente uma grande regularidade (ver Figura 6.21).

Quadro 6.10 - Treliças parametrizadas - programa NEWTOP - tempo de CPU por iteração.

Dados		Com restrições igualdade		Sem restrições igualdade	
Treliça ly × lx	NB	nt	T_CPU (min.)	nt	T_CPU (min.)
t2x4	64	174	0.0041	119	0.0059
t4x8	256	670	0.0680	463	0.0889
t6x12	576	1486	0.5580	1031	0.7050
t8x16	1024	2622	3.0500	1823	3.4200
t10x20	1600	4078	10.1000	2839	11.9000
t12x24	2304	5854	30.1000	4079	33.6000
t14x28	3136	7950	67.1000 *	5543	81.2000
t16x32	4096	10366	149.0000 *	7231	183.0000

Notas:

- O número de iterações necessário à resolução de um problema de optimização situa-se habitualmente entre 15 e 30.
- Os tempos de CPU referem-se a cálculos efectuados num computador com cerca de 40 MFlops.

Simbologia:

- NB - número de barras
- nt - número de variáveis de projecto + número de restrições igualdade
- T_CPU - tempo de CPU por iteração
- * - valor extrapolado

Da comparação entre o comportamento do algoritmo com e sem restrições igualdade pode-se concluir o seguinte:

→ Vantagens da formulação com restrições igualdade

- número de iterações ligeiramente inferior
- tempo de CPU por iteração ligeiramente inferior
- processo iterativo mais estável

→ Vantagens da formulação sem restrições igualdade

- grande economia de memória central
- possibilidade de resolução de problemas com um número mais elevado de variáveis e restrições

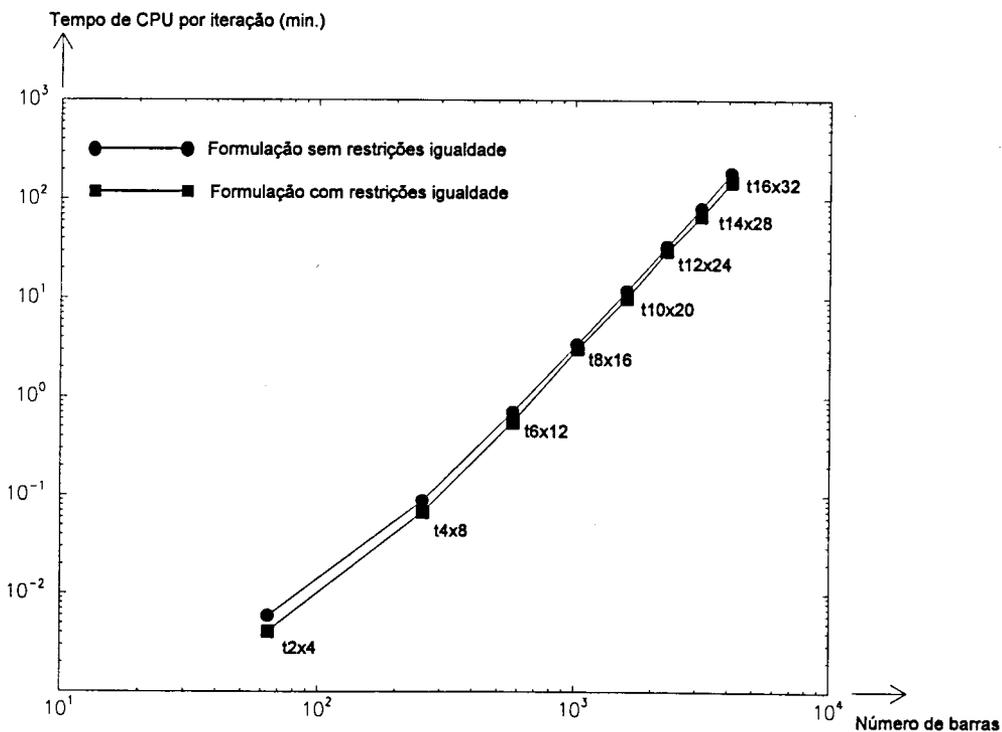


Figura 6.21- Treliças parametrizadas - evolução do tempo de CPU por iteração com o número de barras.

Pelos motivos atrás referidos conclui-se que se deve sempre optar pela formulação com restrições igualdade, exceptuando os casos em que a memória central disponível é insuficiente.

Resolução do sistema de equações pelo método dos gradientes conjugados

O método de Newton requer que em cada iteração seja resolvido um sistema de equações lineares cuja solução é um vector que indica a direcção a seguir para se chegar ao ponto óptimo. No programa NEWTOP existe a possibilidade de efectuar a resolução deste sistema de equações pelo método de eliminação de Gauss adaptado às características da matriz Hessiana ou pelo método dos gradientes conjugados (ver Capítulo 4). É em seguida apresentada uma comparação entre as principais características dos dois métodos, tendo-se recorrido de novo ao conjunto de treliças parametrizadas e com um número de variáveis crescente.

Na Figura 6.22 está representada a evolução do erro associado à resolução do sistema de equações $\tilde{H} \Delta \tilde{X} + \nabla L = 0$ pelo método dos gradientes conjugados. O gráfico representado na figura corresponde à primeira das 17 iterações de Newton necessárias à optimização da treliça $t4 \times 8$. Quer nas restantes iterações de Newton, quer nas restantes treliças do conjunto parametrizado, o gráfico com a história iterativa do método dos gradientes conjugados apresenta sempre um comportamento semelhante ao representado na Figura 6.22, sendo o número de iterações o único aspecto que varia significativamente. Ao gráfico da Figura 6.22 corresponde a resolução de um sistema com 2254 equações, em que foi exigida uma diminuição do erro inicial para um valor 10^{10} vezes inferior. Para se obter uma solução com esta precisão foi necessário efectuar 12 282 iterações. A curva representada na Figura 6.22 apresenta três zonas em que a evolução do erro possui características distintas e que neste caso correspondem aos seguintes intervalos:

$A \rightarrow B$ - acentuado decréscimo do erro;

$B \rightarrow C$ - pequenas oscilações e pequeno decréscimo do erro;

$C \rightarrow D$ - grandes oscilações acompanhadas de um significativo decréscimo do erro.

Se se atender apenas ao valor do erro, facilmente se conclui que somente existem duas estratégias lógicas de interrupção do processo iterativo. A primeira consiste na realização de um pequeno número de iterações até às proximidades do ponto B , tirando assim partido do rápido decréscimo do erro para um valor cerca de 10^4 vezes inferior ao inicial. À segunda estratégia está associada a exigência de um erro significativamente menor, que obriga a um prolongamento do ciclo iterativo até à região entre C e D . Na maior parte dos casos, entre B e C a diminuição do erro é insignificante não havendo portanto interesse em aí interromper o processo iterativo. Nalguns casos chegou a verificar-se um ligeiro aumento do erro entre B e C , que só começou a diminuir após a realização de um elevado número de iterações.

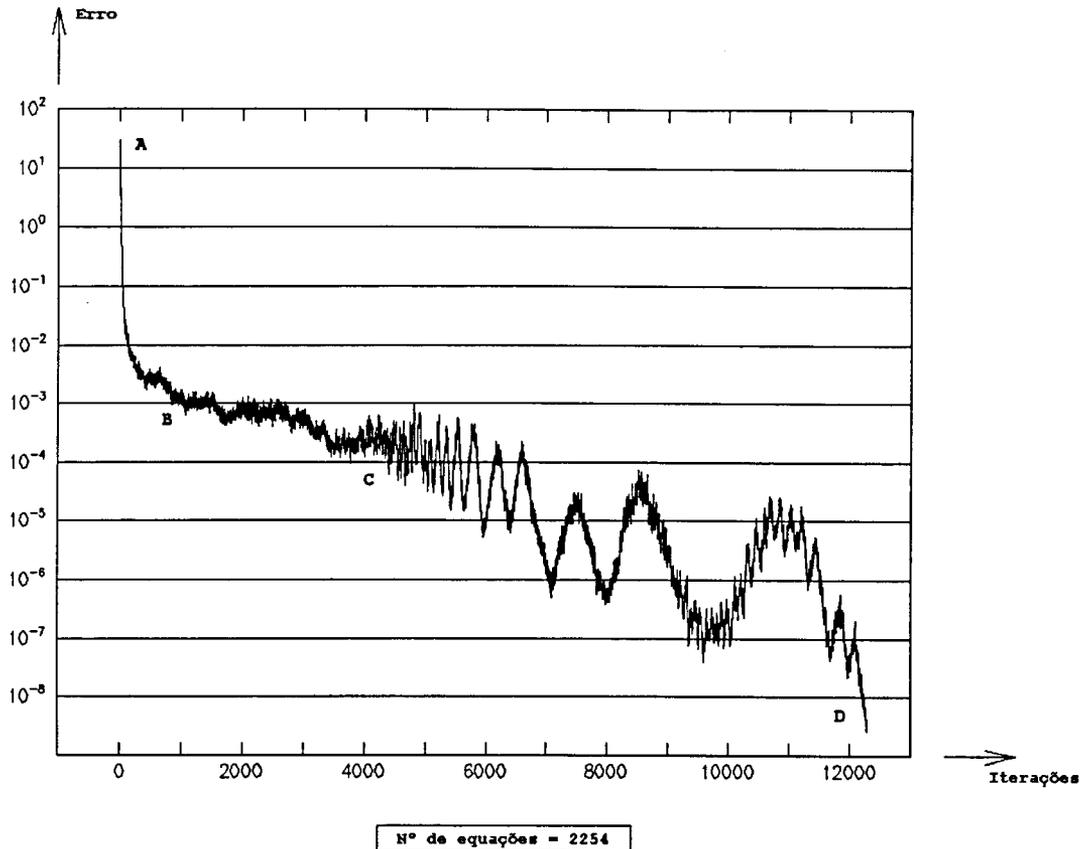


Figura 6.22 - Método dos gradientes conjugados aplicado à resolução de $\tilde{H} \Delta \tilde{X} + \tilde{\nabla} L = 0$. Primeira iteração de Newton relativa à optimização da treliça $t4 \times 8$.

Existe outro aspecto importante que influencia a decisão relativa ao número de iterações que devem ser efectuadas na resolução do sistema de equações $\tilde{H} \Delta \tilde{X} + \tilde{\nabla} L = 0$ pelo método dos gradientes conjugados. Uma vez que este processo iterativo é interior ao processo iterativo relativo ao método de Newton, a ausência de precisão na resolução do referido sistema de equações lineares vai pôr em risco a possibilidade de se chegar a uma solução do programa matemático. Este problema agrava-se significativamente à medida que o número de variáveis de projecto aumenta. Verificou-se também que um aumento do erro na resolução do sistema de equações lineares implica um aumento do número de iterações a realizar pelo método de Newton. Nas proximidades do ponto óptimo, se o sistema de equações lineares não for resolvido com elevada precisão, não se consegue obter uma solução rigorosa para o problema de optimização.

Conclui-se portanto que na abordagem de problemas de optimização com recurso ao método dos gradientes conjugados para a resolução do sistema de equações lineares, se deve

calcular a solução de cada um desses sistemas com um número de iterações da mesma ordem de grandeza do número de equações. Assim, evita-se o inconveniente de ter de realizar um número exagerado de iterações para conseguir obter uma solução significativamente mais rigorosa. Nos casos em que esta estratégia não conduz a uma solução do problema de optimização, tem de passar a ser realizado um número de iterações da ordem de 5 a 10 vezes o número de equações. No Quadro 6.11 encontram-se os tempos de CPU por iteração relativos à resolução dos problemas de optimização das treliças $t2 \times 4$ a $t8 \times 16$ com quatro estratégias distintas. A consulta dos valores do Quadro 6.11 permite concluir que nos casos em que o sistema de equações é resolvido com uma precisão elevada, os tempos de CPU tornam-se também muito elevados, sendo claramente mais vantajoso o recurso ao método de eliminação de Gauss. Quando o número de iterações é igual ao número de equações, o tempo de CPU é da mesma ordem de grandeza do correspondente às estratégias A e B, mas sempre significativamente superior.

Quadro 6.11 - Tempo de CPU por iteração de Newton (minutos). Entre parênteses encontra-se o número de iterações efectuadas pelo método dos gradientes conjugados em cada iteração de Newton.

Treliça $ly \times lx$	NB	A (min.)	B (min.)	C (min.)	D (min.)
t2x4	64	0.0041	0.0059	0.0182 (566)	0.0328 (1000)
t4x8	256	0.0680	0.0889	0.2790 (2254)	2.7000 (25000)
t6x12	576	0.5580	0.7050	2.0500 (5062)	18.5000 (50000)
t8x16	1024	3.0500	3.4200	7.4700 (8990)	85.5000 (100000)

Simbologia:

NB - número de barras

Resolução do sistema de equações:

- A - método de eliminação de Gauss - formulação com restrições igualdade
- B - método de eliminação de Gauss - formulação sem restrições igualdade
- C - método dos gradientes conjugados com um número de iterações igual ao número de equações
- D - método dos gradientes conjugados com um número de iterações que conduza a uma solução do sistema de equações lineares com um erro da mesma ordem de grandeza do obtido em A

Nota:

Os resultados relativos ao método dos gradientes conjugados correspondem à formulação com restrições igualdade.

No Quadro 6.12 é evidenciada a grande vantagem do método dos gradientes conjugados relativamente ao método de eliminação de Gauss, que consiste na grande economia de memória central. A causa desta acentuada diferença é o facto de ao método de eliminação de Gauss estar associada a necessidade de armazenar o triângulo superior da matriz Hessiana, enquanto que o método iterativo apenas requer o armazenamento dos respectivos termos não nulos. A diferença entre as necessidades de memória central dos dois métodos torna-se cada vez mais significativa à medida que o número de equações cresce.

Quadro 6.12 - Memória central utilizada durante a resolução do sistema de equações lineares
 $\tilde{H} \Delta \tilde{X} + \nabla L = 0$ (MBytes).

Treliça $l_y \times l_x$	NB	A (MB)	B (MB)	C (MB)
t2x4	64	0.122	0.057	0.060
t4x8	256	1.798	0.859	0.250
t6x12	576	8.839	4.256	0.550
t8x16	1024	27.510	13.301	0.980

Simbologia:

NB - número de barras

Resolução do sistema de equações:

- A - método de eliminação de Gauss - formulação com restrições igualdade
- B - método de eliminação de Gauss - formulação sem restrições igualdade
- C - método dos gradientes conjugados - armazenamento dos termos não nulos da matriz Hessiana - i,j,H_{ij} (4+4+8 bytes) e vectores auxiliares

Como conclusão final, refere-se que a grande vantagem do método dos gradientes conjugados, que consiste na economia de memória central, é fortemente penalizada com o agravamento do tempo de CPU. Sempre que a memória central disponível é suficiente, deve ser utilizado o método de eliminação de Gauss e a formulação com restrições igualdade.

Características das soluções óptimas

Devido ao elevado número de variáveis de projecto envolvidas no processo de optimização do conjunto de treliças atrás apresentado, revela-se imprescindível o recurso à visualização gráfica para se conseguir ter uma percepção clara das decisões que foram tomadas pelo algoritmo de optimização (ver Figuras 6.23 a 6.30). Para evidenciar a

distribuição das áreas das secções transversais, foi desenvolvido um programa de computador que executa o seguinte conjunto de operações:

- Substituição de cada barra da treliça por um cilindro de eixo coincidente com o eixo da barra. A altura do cilindro é igual ao comprimento da barra e a área da base obtém-se multiplicando a área da secção transversal por um factor que é igual para todas as barras. Este factor varia de treliça para treliça e foi escolhido de modo a realçar o mais possível as características da solução representada.
- Geração de uma malha de elementos finitos sobre a superfície de cada cilindro. São utilizados elementos curvilíneos de oito nós. A superfície lateral do cilindro é discretizada com quatro elementos e cada base é representada por um único elemento degenerado em círculo (ver Figura 6.31).
- Visualização da malha de elementos finitos constituída pelo conjunto de cilindros. É utilizado um algoritmo de iluminação de objectos, são eliminadas as superfícies escondidas recorrendo a um *Z buffer* e os tons de cinzento são simulados por *dithering* [San89].

A observação das soluções óptimas representadas nas Figuras 6.23 a 6.30 permite concluir que todas elas apresentam determinadas características comuns. Segundo o menor vão formaram-se sempre duas nervuras, que são mais evidentes nas treliças com maior número de barras. Nas proximidades da zona periférica sem apoios formou-se também uma nervura com uma orientação perpendicular às outras duas. Na confluência desta nervura com a nervura central existe um conjunto de barras diagonais que se evidenciam em relação às restantes. As barras que apresentam maior secção transversal situam-se no início da zona periférica sem apoios, pelo facto de aí existir uma acentuada concentração de esforços.

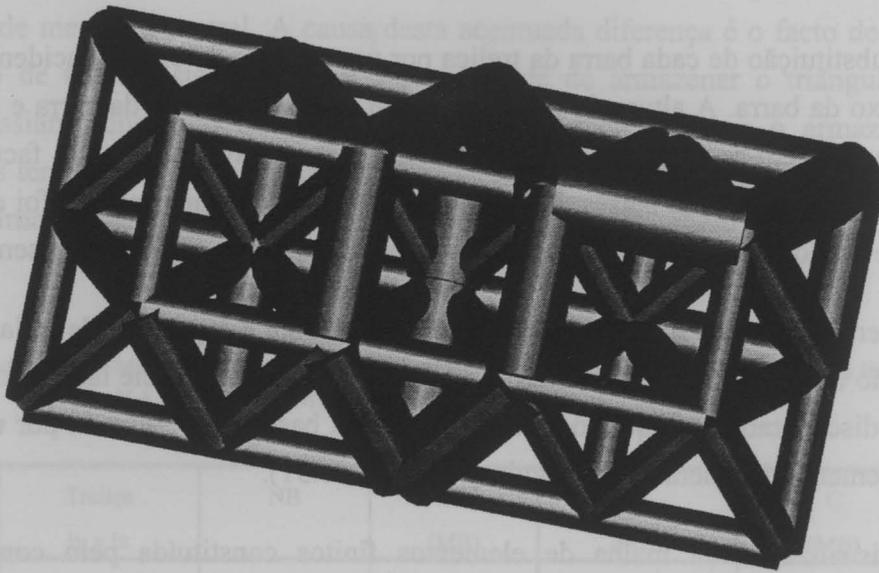


Figura 6.23 - Treliça 2×4 - visualização da solução óptima (factor de ampliação das áreas das barras = 3200).

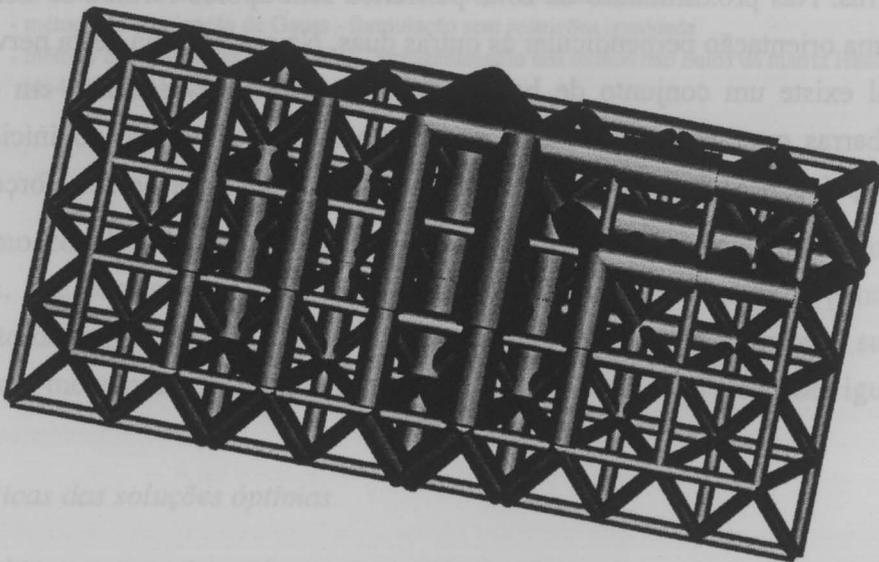


Figura 6.24 - Treliça 4×8 - visualização da solução óptima (factor de ampliação das áreas das barras = 830).

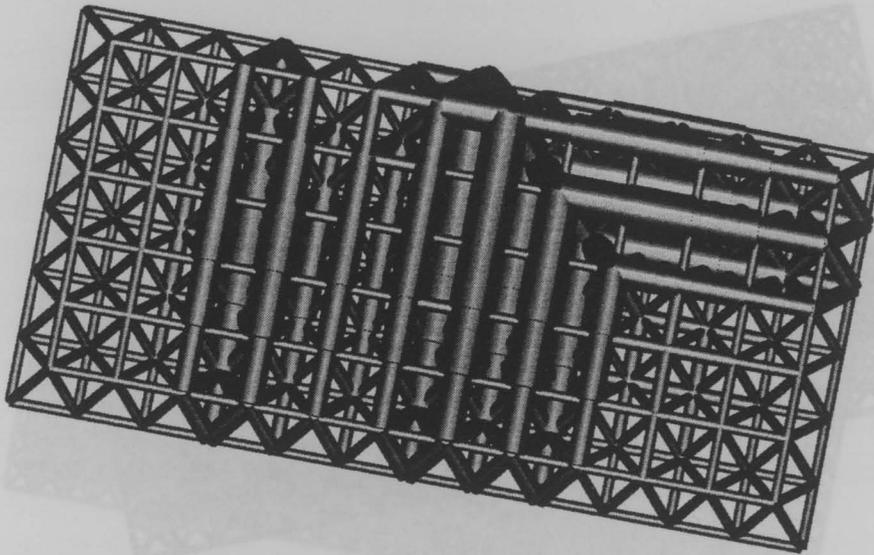


Figura 6.25 - Treliça $t6 \times 12$ - visualização da solução óptima (factor de ampliação das áreas das barras = 380).

Figura 6.29 - Treliça $t4 \times 24$ - visualização da solução óptima (factor de ampliação das áreas das barras = 15).

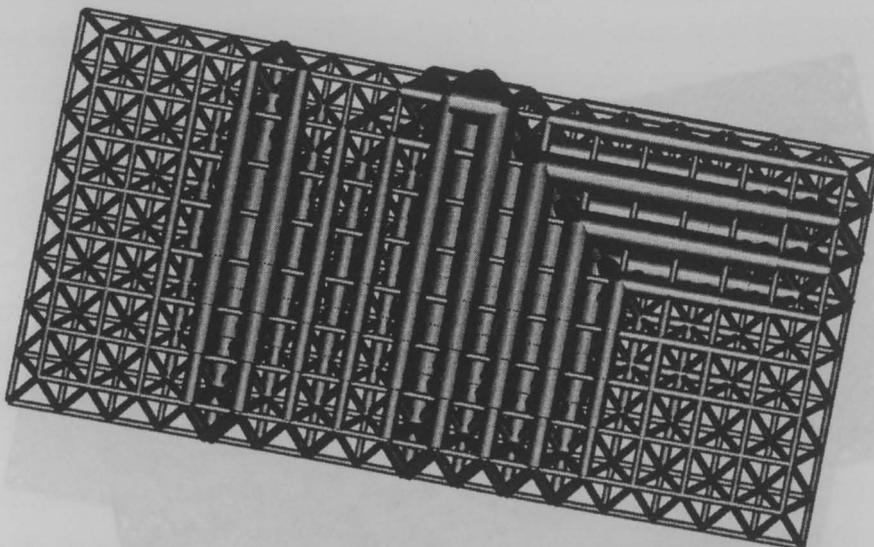


Figura 6.26 - Treliça $t8 \times 16$ - visualização da solução óptima (factor de ampliação das áreas das barras = 190).

Figura 6.28 - Treliça $t3 \times 24$ - visualização da solução óptima (factor de ampliação das áreas das barras = 80).
Figura 6.30 - Treliça $t6 \times 32$ - visualização da solução óptima (factor de ampliação das áreas das barras = 25).

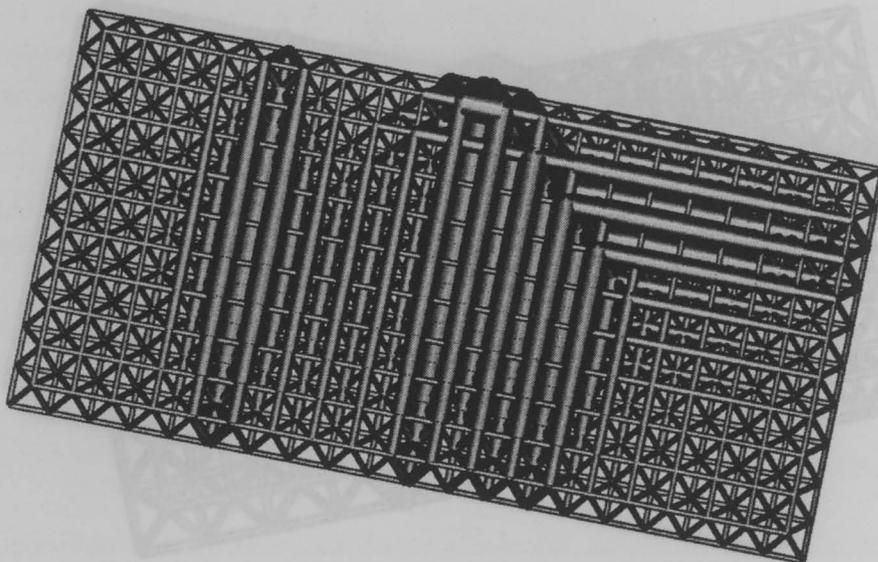


Figura 6.27 - Treliça $t10 \times 20$ - visualização da solução óptima (factor de ampliação das áreas das barras = 100).

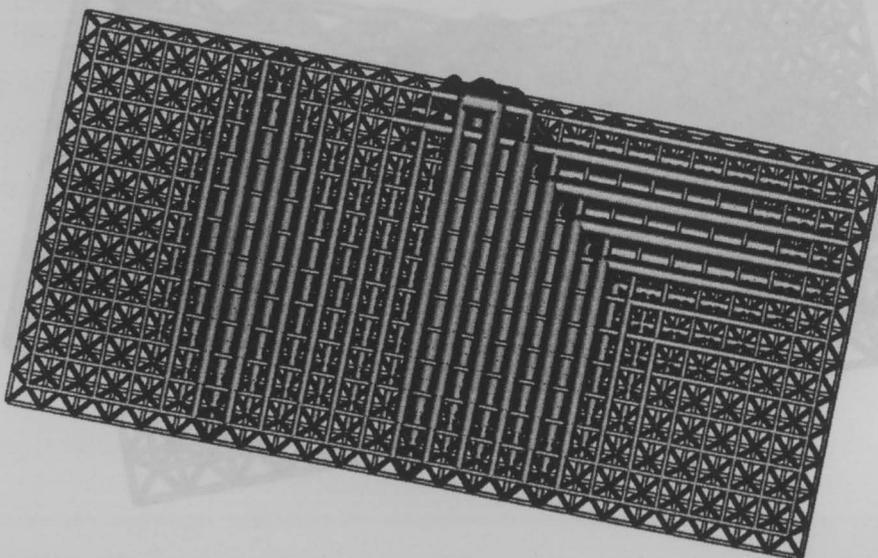


Figura 6.28 - Treliça $t12 \times 24$ - visualização da solução óptima (factor de ampliação das áreas das barras = 60).

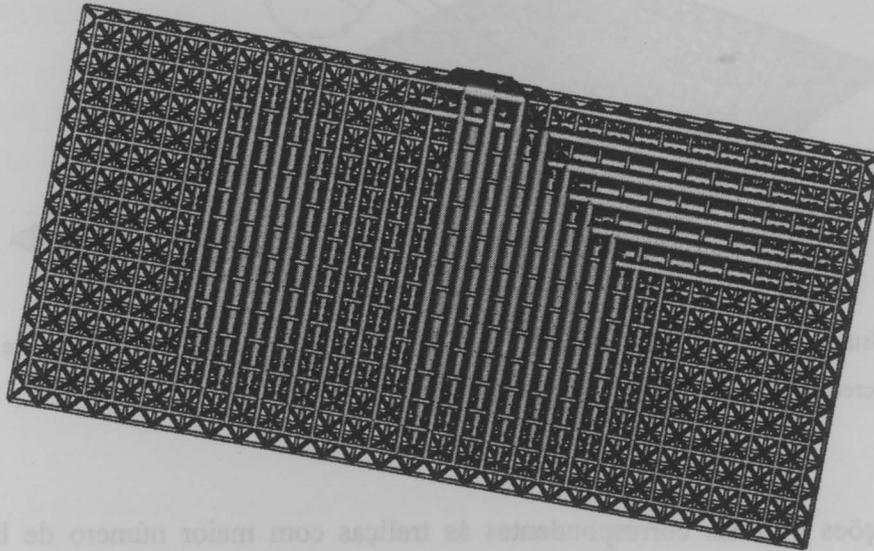


Figura 6.29 - Treliça $t14 \times 28$ - visualização da solução óptima (factor de ampliação das áreas das barras = 35).

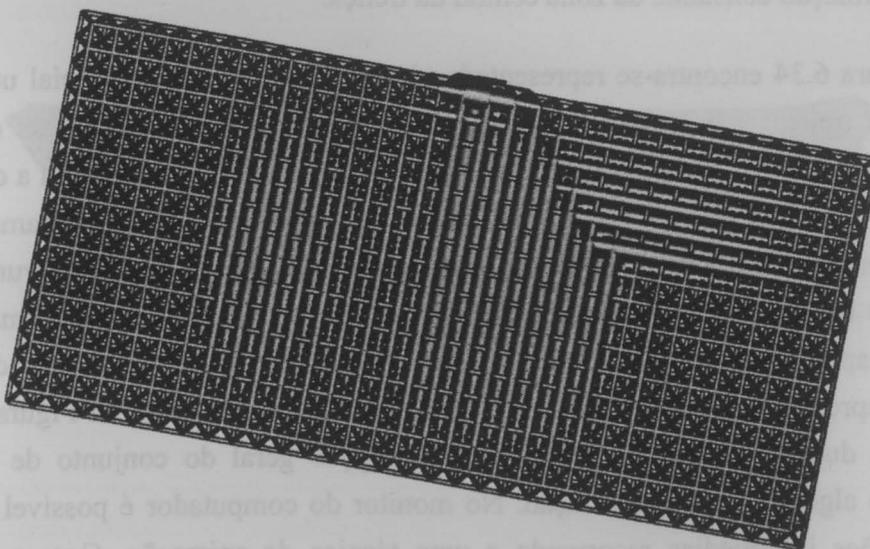


Figura 6.30 - Treliça $t16 \times 32$ - visualização da solução óptima (factor de ampliação das áreas das barras = 25).

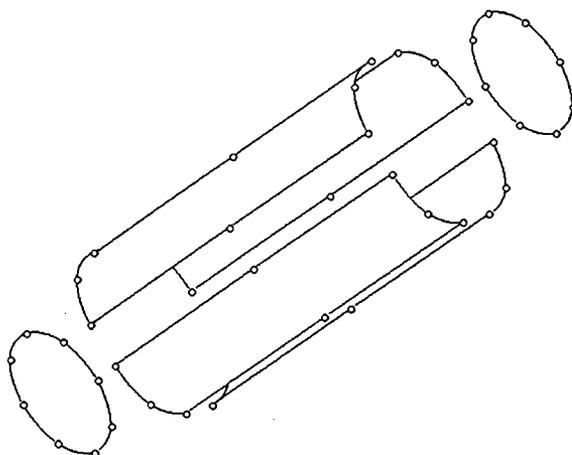


Figura 6.31 - Na visualização de treliças tridimensionais cada barra é substituída por um cilindro que por sua vez é discretizado em seis elementos de casca

As soluções ótimas correspondentes às treliças com maior número de barras são condicionadas pelas restrições de deslocamento vertical dos nós. Por este motivo, a estrutura otimizada apresenta um elevado número de nós com deslocamento vertical igual ou próximo do máximo permitido. Estes nós situam-se na zona central e na zona periférica sem apoios. Nas Figuras 6.32 e 6.33 encontra-se representada a deformada correspondente à solução óptima da treliça $t16 \times 32$. Na primeira, todas as barras foram representadas com a mesma secção transversal, enquanto que na segunda é retomada a representação com cilindros de secção proporcional à área de cada barra na solução óptima. Esta representação gráfica torna evidente a deformação constante da zona central da treliça.

Na Figura 6.34 encontra-se representada vista de baixo a solução inicial utilizada na optimização da treliça $t12 \times 24$. Esta solução foi obtida multiplicando as áreas das barras correspondentes ao *Fully Stressed Design* (FSD) por um factor constante e igual a cinco. Este aumento das áreas das barras teve como objectivo uma diminuição dos deslocamentos para que a solução inicial fosse admissível. A solução FSD apenas apresenta uma nervura segundo o menor vão e um reduzido número de barras com área superior à área mínima na zona periférica sem apoios. O algoritmo de optimização modificou esta distribuição de secções transversais e apresentou como solução óptima a estrutura representada na Figura 6.35. Ao comparar estas duas figuras fica-se com uma percepção geral do conjunto de alterações efectuadas pelo algoritmo de optimização. No monitor do computador é possível visualizar todas as soluções intermédias recorrendo a uma técnica de animação. Com o *hardware* presentemente disponível consegue-se representar cerca de 15 imagens por segundo com uma resolução de 832×720 pixels e 256 cores simultâneas. Esta animação permite visualizar o faseamento das alterações efectuadas pelo algoritmo de optimização.

6.3.4 - Comparação com os resultados de outros autores

Para comparar os resultados de um determinado método de otimização proposto com outros métodos cujo objetivo é a obtenção de estruturas ótimas, foram apresentados três exemplos de minimização de peso. O primeiro exemplo é um problema de otimização de uma treliça plana, que é um trabalho em que se pretende obter a estrutura ótima para um determinado estado de tensão. O terceiro exemplo é um problema de otimização de uma treliça espacial, cuja formulação é semelhante à do primeiro exemplo.

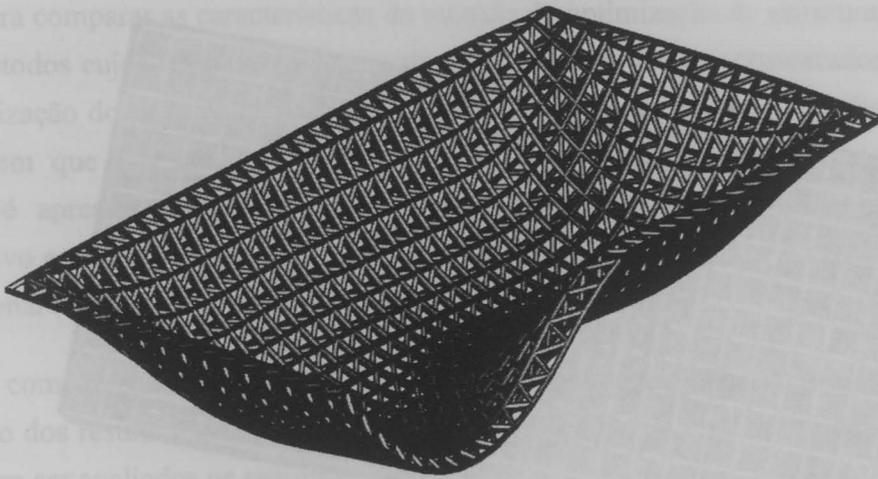


Figura 6.32 - Treliça $t16 \times 32$ - deformada correspondente à solução óptima.

1) Precisão - as restrições devem ser devidamente respeitadas para que a solução seja admissível. É também desejável que o critério de optimalidade de um método de otimização seja o mesmo que o utilizado no método de referência.

2) Eficiência - a comparação entre os diferentes métodos de otimização computacional executados em computadores com características diferentes é uma tarefa complicada. Os resultados desta comparação são influenciados por fatores como a rapidez do método, a qualidade da implementação do método de otimização, a eficiência da linguagem de programação utilizada, etc.

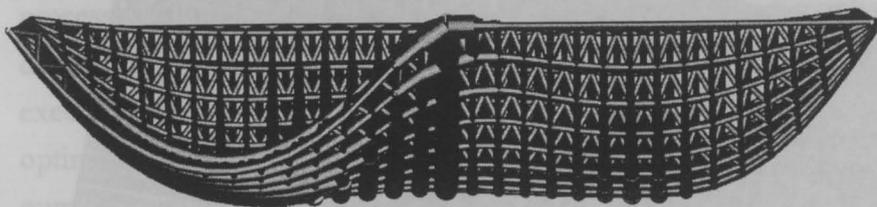


Figura 6.33 - Treliça $t16 \times 32$ - deformada correspondente à solução óptima.

(00) - ...

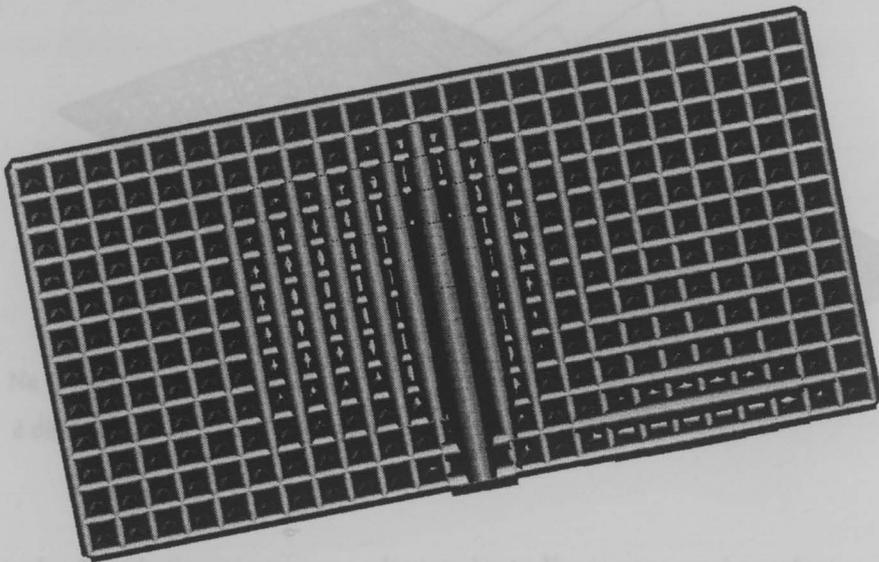
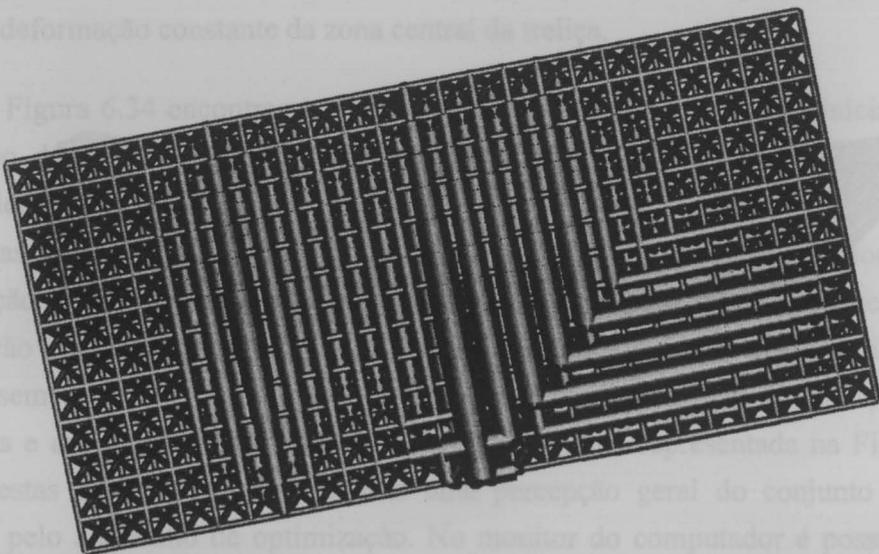


Figura 6.31 - Na Figura 6.31 encontra-se representada a deformada correspondente à solução ótima da treliça $t16 \times 32$. Na primeira, todas as barras foram representadas com a mesma secção transversal, enquanto que na segunda é retomada a representação com cilindros de secção proporcional à área de cada barra na solução ótima. Esta representação gráfica torna evidente a deformação constante da zona central da treliça.

As soluções ótimas correspondentes às treliças com maior número de barras são condicionadas pelas restrições de deslocamento vertical dos nós. Por este motivo, a estrutura otimizada apresenta uma deformação vertical igual ou próxima da deformação da estrutura inicial.

Figura 6.34 - Treliça $t12 \times 24$ vista de baixo - solução inicial (factor de ampliação das áreas das barras = 100).

Nas Figuras 6.32 e 6.33 encontra-se representada a deformada correspondente à solução ótima da treliça $t16 \times 32$. Na primeira, todas as barras foram representadas com a mesma secção transversal, enquanto que na segunda é retomada a representação com cilindros de secção proporcional à área de cada barra na solução ótima. Esta representação gráfica torna evidente a deformação constante da zona central da treliça.



Na Figura 6.34 encontra-se representada a deformada correspondente à solução inicial utilizada na optimização da treliça $t12 \times 24$. As barras centrais são representadas com um cilindro de secção cinco vezes maior do que as barras periféricas. Este factor de ampliação das áreas das barras foi escolhido para que a solução inicial apresentasse uma deformação vertical igual ou próxima da deformação da estrutura segundo a solução ótima. Na Figura 6.35 encontra-se representada a deformada correspondente à solução ótima da treliça $t12 \times 24$. Ao comparar estas duas imagens, torna-se evidente a redistribuição do material para a zona periférica sem alteração da deformação vertical. Esta redistribuição é conseguida através de alterações de secções transversais e de comprimento das barras.

Na Figura 6.35 encontra-se representada a deformada correspondente à solução ótima da treliça $t12 \times 24$. Ao comparar estas duas imagens, torna-se evidente a redistribuição do material para a zona periférica sem alteração da deformação vertical. Esta redistribuição é conseguida através de alterações de secções transversais e de comprimento das barras.

Figura 6.35 - Treliça $t12 \times 24$ vista de baixo - solução ótima (factor de ampliação das áreas das barras = 100).

6.3.4 - Comparação com os resultados de outros autores

Para comparar as características do método de optimização de estruturas proposto com outros métodos cujos resultados se encontram publicados, foram considerados três exemplos de minimização do peso de treliças. Os dois primeiros foram extraídos de [Bel82], que é um trabalho em que são também comparados diversos métodos de optimização. O terceiro exemplo é apresentado em [Haf85] e [Haf87], tendo este autor efectuado um estudo comparativo entre a formulação integrada (*simultaneous analysis and design*) e a formulação convencional (*nested*).

A comparação entre diferentes métodos de optimização é geralmente efectuada por intermédio dos resultados obtidos com os correspondentes códigos computacionais. Em cada caso devem ser avaliados os seguintes aspectos:

- 1) *Precisão* - as restrições devem ser rigorosamente respeitadas para que a solução seja admissível. É também desejável que o critério de optimalidade ou o critério de terminação do processo iterativo sejam verificados com elevada precisão. Se tal não se verificar, a solução não é sequer um mínimo local. A uma elevada precisão está obviamente associado um elevado número de algoritmos significativos correctos nos resultados.
- 2) *Eficiência* - a comparação entre as eficiências de distintos códigos computacionais executados em computadores com características diferentes é uma tarefa complicada e sem respostas únicas. Os resultados desta comparação são influenciados por algumas características do computador, tais como a rapidez do processador, a quantidade de memória central e de massa disponíveis, a qualidade do código máquina produzido pelo compilador, a possibilidade de tirar partido de execuções em paralelo, etc. É também importante o facto de o método de optimização ter sido programado tendo em vista a obtenção da máxima eficiência num computador com determinadas características. Nestes casos existe quase sempre uma perda de portabilidade do código computacional. Devido à constante evolução da informática, com o decorrer do tempo as conclusões de um estudo comparativo de eficiências de códigos computacionais podem mesmo ser invertidas. Existem ainda outros aspectos que também dificultam a comparação entre métodos de optimização, tais como as características do problema abordado e a adaptação do código computacional a essas características. Devido a todas estas dificuldades apenas serão aqui referidas ordens de grandeza dos tempos de resolução correspondentes aos diversos métodos.

- 3) *Fiabilidade* - a probabilidade de sucesso na resolução de um problema de optimização de estruturas depende muito das características da solução inicial. Sempre que a convergência para uma solução falhar, é possível recorrer a outra solução inicial. Só se deve portanto considerar que um método de optimização não é capaz de resolver um determinado problema, depois de múltiplas tentativas falharem. Existe também uma grande dependência relativamente ao tipo de problema que se está a tentar resolver. Por todos estes motivos revela-se muito difícil classificar métodos de optimização quanto à fiabilidade. Só a experiência acumulada na resolução de um elevado número de problemas recorrendo a distintos métodos de optimização pode conduzir à formação de uma opinião relativa à fiabilidade de cada um deles.
- 4) *Convergência global* - uma vez que só em casos particulares é conhecido o mínimo global de um problema de optimização de estruturas, torna-se muito difícil saber se num determinado problema esse mínimo global foi ou não obtido. Existe no entanto a certeza de que uma solução é um mínimo local quando se dispõe de pelo menos uma solução melhor. Uma vez que a convergência para o mínimo global ou para um mínimo local depende da solução inicial, a classificação dos métodos de optimização quanto à convergência global apresenta as mesmas dificuldades referidas atrás para o caso da fiabilidade.
- 5) *Versatilidade* - alguns métodos de optimização apresentam uma elevada eficiência devido ao facto de ser explorada uma característica particular do problema que se pretende resolver. A este procedimento está habitualmente associada uma perda de versatilidade, que dificulta a abordagem de problemas em que algumas características sejam distintas daquelas que foram consideradas na formulação original. Esta situação surge, por exemplo, quando existe a necessidade de acrescentar à formulação do problema um tipo de restrições que não foi inicialmente previsto. A versatilidade é portanto uma característica desejável, mas que poucos programas de optimização de estruturas possuem.
- 6) *Intervenção do utilizador* - todos os métodos de optimização necessitam da definição de alguns parâmetros e tolerâncias que influenciam o comportamento do processo iterativo. É habitual existir um conjunto de valores para esses parâmetros, que conduzem a uma solução do problema de optimização com uma eficiência aceitável. Os problemas de divergência ou de convergência lenta do processo iterativo são algumas vezes resolvidos alterando os valores desses parâmetros. Para que esta intervenção seja bem sucedida, é fundamental a experiência do utilizador e alguns conhecimentos sobre o algoritmo utilizado. A complexidade deste

procedimento depende do método de optimização e da qualidade do respectivo código computacional.

Apresentam-se em seguida os exemplos de optimização de estruturas extraídos de [Bel82], [Haf85] e [Haf87]. As características dos métodos utilizados por estes autores, bem como os resultados por eles obtidos são comparados com os correspondentes ao programa NEWTOP. Nas três treliças apresentadas, o objectivo é a minimização do peso da estrutura com as secções transversais das barras como variáveis de projecto.

6.3.4.1 - Treliça de 10 barras

Este problema é dos mais utilizados na apresentação e comparação de diferentes características dos métodos de optimização [Hau79] [Kir81] [Bel82]. É considerado um problema difícil de resolver, porque a distribuição de esforços nas barras é muito sensível à variação dos valores das variáveis de projecto [Ber87]. Alguns autores introduzem pequenas alterações nos dados do problema de optimização da treliça de 10 barras destinadas a realçar um ou outro aspecto particular. Os resultados aqui apresentados correspondem ao Caso 1a de [Bel82], cujos dados se encontram na Figura 6.36 e no Quadro 6.13.

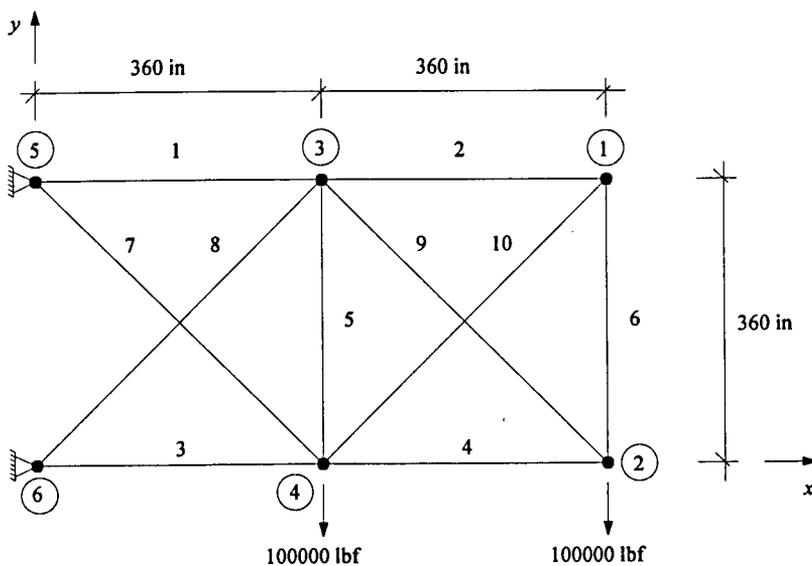


Figura 6.36 - Treliça de 10 barras - geometria e acções.

Quadro 6.13 - Dados relativos ao problema de optimização da treliça de 10 barras.

Designação : tr10			
Objectivo : minimização do peso			
Restrições : área mínima, tensão mínima/máxima e deslocamento mínimo/máximo			
Nota: estas restrições foram consideradas em todas as barras e segundo todos os graus de liberdade.			
Unidade de força	:	libra força (lbf)	(1 lbf = 4.448 N)
Unidade de comprimento	:	polegada (in)	(1 in = 0.0254 m)
Número de barras	-----	10	
Número de variáveis de projecto independentes	----	10	
Número de graus de liberdade não prescritos	-----	8 (2*4)	
Número de restrições desigualdade	-----	46 (10+2*10+2*8)	
Módulo de Young	(E)	: 10e6	lbf/in ²
Peso específico	(γ)	: 0.1	lbf/in ³
Área mínima	(a min)	: 0.1	in ²
Tensão mínima/máxima	(σ min/max)	: -25e3, 25e3	lbf/in ²
Deslocamento mínimo/máximo	(d min/max)	: -2.0, 2.0	in
Melhor solução conhecida	(C)	: 5132.420	lbf
Acções no único caso de carga:			
Peso próprio + forças de -100000 lbf segundo y nos nós 2 e 4			

Convém realçar que não é considerada a encurvadura local nem a global e que é incluído o peso próprio como uma função das variáveis de projecto. Em algumas publicações [Hau79] [Fle88] o peso próprio não é considerado, sendo nesses casos obtida uma solução melhor do que a indicada no Quadro 6.13. A conversão para o SI das unidades utilizadas é elementar, uma vez que apenas são referidas unidades de força e de deslocamento (*lbf* e *in*).

No Quadro 6.14 encontram-se os valores finais da função objectivo correspondentes à resolução do problema com os diversos métodos e algoritmos. Com o programa NEWTOP foi possível obter com elevada precisão a solução que se supõe ser o mínimo global. Para chegar a esta solução foi necessário efectuar um número considerável de tentativas com distintas soluções iniciais, porque na generalidade dos casos a solução que se obtém corresponde a um mínimo local muito próximo do global. O peso correspondente a este mínimo local é apenas 0.3% superior ao peso mínimo referido no Quadro 6.13. A principal diferença entre estas duas soluções reside no facto de à área da barra 6 ser ou não atribuído o valor mínimo. Devido à grande precisão dos resultados obtidos com o programa NEWTOP, é possível distinguir claramente estes dois mínimos. Ao contrário do que é habitual noutros problemas, ambas as soluções apresentam todos os multiplicadores de Lagrange associados a restrições desigualdade com valores positivos, verificando assim as condições de Karush-Kuhn-Tucker. As soluções obtidas com os diversos programas referidos em [Bel82] apresentam pesos 7 a 25% superiores ao mínimo global, exceptuando o caso do programa M-4 que não convergiu. No Quadro 6.15 encontram-se as áreas das barras correspondentes à solução obtida com o

programa NEWTOP e à melhor solução publicada em [Bel82]. Apesar de os respectivos pesos apenas diferirem 7%, verifica-se que algumas barras apresentam áreas significativamente diferentes. Com os restantes métodos foram obtidas soluções que revelam uma grande dispersão de resultados, devido provavelmente à convergência para distintos mínimos locais.

Quadro 6.14 - Treliza de 10 barras - comparação entre a solução obtida com o programa NEWTOP e as soluções publicadas em [Bel82].

Programa	Peso final (lbf)	Método
NEWTOP	5132.420 -	método de Newton
OPTDYN	5472.000 (+7%)	método das direcções admissíveis
CONMIN	5563.000 (+8%)	método das direcções admissíveis
M-3	5719.000 (+11%)	algoritmo de Powell
M-5	5725.000 (+12%)	algoritmo de Fletcher
GRP-UI	5727.000 (+12%)	método do gradiente projectado
SUMT	5932.000 (+16%)	min. sem restr. sequencial-penalidades externas
LINRM	6429.000 (+25%)	programação quadrática sequencial
M-4	11280.000 (+120%)	algoritmo de Powell

Quadro 6.15 - Treliza de 10 barras - comparação entre a solução obtida com o programa NEWTOP e a melhor solução publicada em [Bel82].

Barra	NEWTOP Área (in ²)	OPTDYN Área (in ²)
1	31.114813	25.774018
2	0.100000	0.100000
3	23.570890	25.113741
4	15.317338	19.388141
5	0.100000	0.100000
6	0.551968	0.100000
7	7.516377	15.359036
8	21.541657	20.322401
9	21.620875	20.743674
10	0.100000	1.140175
Peso (lbf) →	5132.420	5472.000

Os tempos de CPU correspondentes à resolução deste problema são muito baixos, não se justificando a realização de comparações entre os diferentes métodos.

6.3.4.2 - Treliça de 200 barras

Em [Bel82] foi efectuado um outro estudo comparativo de características semelhantes. Tratou-se da aplicação de diversos métodos de optimização na resolução de um problema de minimização do peso de uma treliça de 200 barras (ver Figura 6.37). Esta treliça foi inicialmente estudada em [Aro74] e apresenta como principais características o facto de ser solicitada por vários casos de carga e de ser exigida uma solução simétrica com alguma regularidade. Estas restrições adicionais são consideradas por intermédio de um agrupamento de variáveis de projecto (*variable linking*), cuja especificação se encontra no Quadro 6.16. Os restantes dados relativos a este problema encontram-se no Quadro 6.17.

Quadro 6.16 - Definição dos grupos de barras (*variable linking*).

Grupo	Barras...			
1 -	1	2	3	4
2 -	5	8	11	14 17
3 -	19	20	21	22 23 24
4 -	18	25	56	63 94 101 132 139 170 177
5 -	26	29	32	35 38
6 -	6	7	9	10 12 13 15 16 27 28 30 31 33 34 36 37
7 -	39	40	41	42
8 -	43	46	49	52 55
9 -	57	58	59	60 61 62
10 -	64	67	70	73 76
11 -	44	45	47	48 50 51 53 54 65 66 68 69 71 72 74 75
12 -	77	78	79	80
13 -	81	84	87	90 93
14 -	95	96	97	98 99 100
15 -	102	105	108	111 114
16 -	82	83	85	86 88 89 91 92 103 104 106 107 109 110 112 113
17 -	115	116	117	118
18 -	119	122	125	128 131
19 -	133	134	135	136 137 138
20 -	140	143	146	149 152
21 -	120	121	123	124 126 127 129 130 141 142 144 145 147 148 150 151
22 -	153	154	155	156
23 -	157	160	163	166 169
24 -	171	172	173	174 175 176
25 -	178	181	184	187 190
26 -	158	159	161	162 164 165 167 168 179 180 182 183 185 186 188 189
27 -	191	192	193	194
28 -	195	197	198	200
29 -	196	199		

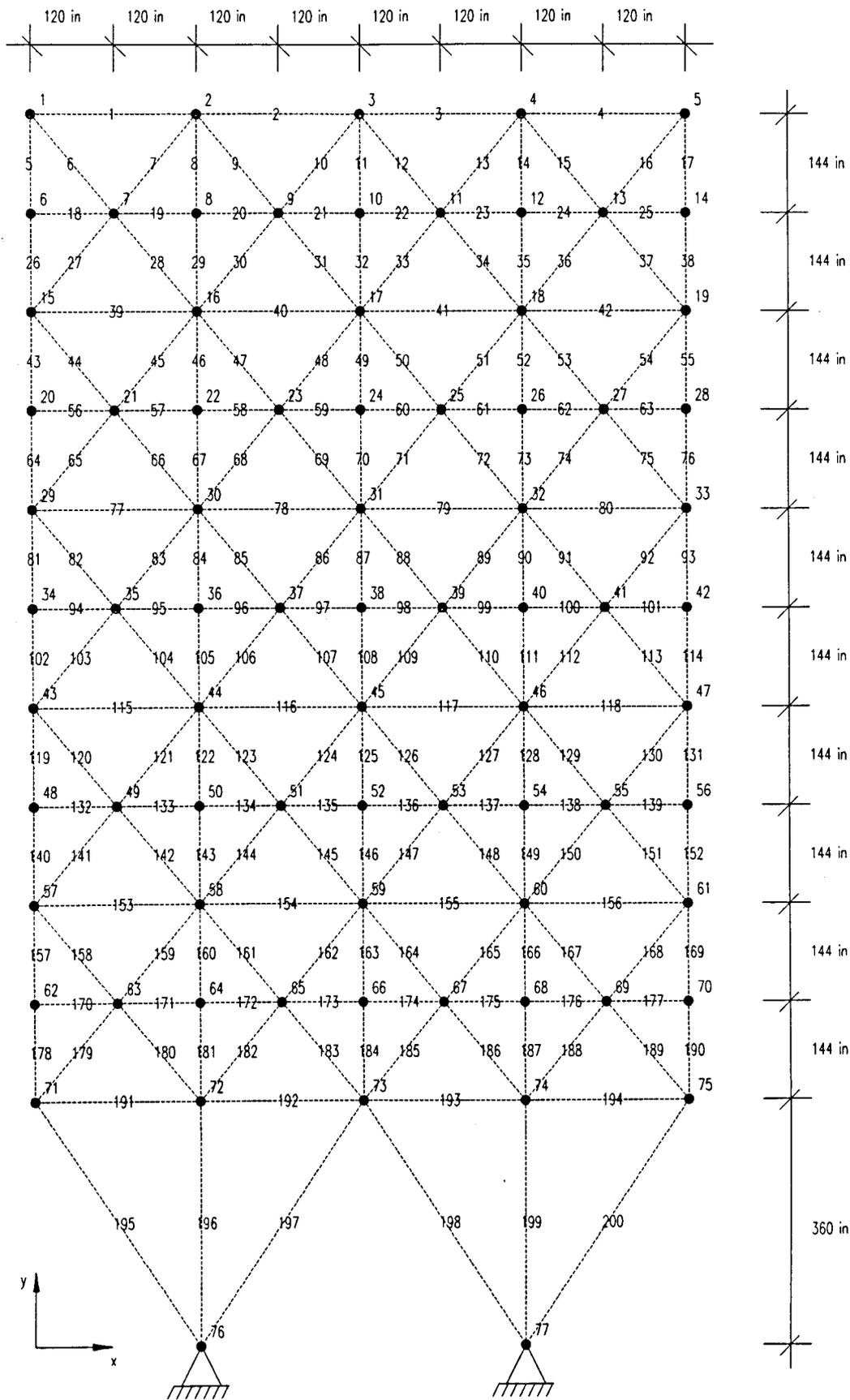


Figura 6.37 - Treliça de 200 barras - geometria e numeração dos nós e das barras.

Quadro 6.17 - Dados relativos ao problema de optimização da treliça de 200 barras.

Designação : tr200																				
Objectivo : minimização do peso																				
Restrições : área mínima e tensão mínima/máxima																				
Nota: estas restrições foram consideradas em todas as barras e em todos os casos de carga.																				
Unidade de força	:	libra força (lbf)	(1 lbf = 4.448 N)																	
Unidade de comprimento	:	polegada (in)	(1 in = 0.0254 m)																	
Número de barras	-----	200																		
Número de variáveis de projecto independentes	----	29																		
Número de graus de liberdade não prescritos	-----	450	(3*2*75)																	
Número de restrições desigualdade	-----	1229	(29+3*2*200)																	
Módulo de Young	(E)	: 30e6	lbf/in ²																	
Peso específico	(γ)	: 0.283	lbf/in ³																	
Área mínima	(a min)	: 0.1	in ²																	
Tensão mínima/máxima	(σ min/max)	: -10e3, 10e3	lbf/in ²																	
Deslocamento mínimo/máximo	(d min/max)	: -	in																	
Melhor solução conhecida	(C)	: 26124.041	lbf																	
Acções em cada caso de carga:																				
1 - Peso próprio + forças de 1000 lbf segundo x nos nós do grupo A																				
2 - Peso próprio + forças de -10000 lbf segundo y nos nós do grupo B																				
3 - Peso próprio + forças do caso de carga 1 + forças do caso de carga 2																				
Nós do grupo A:																				
1 6 15 20 29 34 43 48 57 62 71																				
Nós do grupo B:																				
1	2	3	4	5	6	8	10	12	14	15	16	17	18	19	20	22	24	26	28	
29	30	31	32	33	34	36	38	40	42	43	44	45	46	47	48	50	52	54	56	
57	58	59	60	61	62	64	66	68	70	71	72	73	74	75						

Neste problema não existem restrições de deslocamento e não é considerada a encurvadura local nem a global. O peso próprio é incluído como uma função das variáveis de projecto. O agrupamento de barras faz com que o número de variáveis de projecto independentes passe de 200 para 29. O número de variáveis de projecto relativas aos deslocamentos dos nós aparece multiplicado por três, devido ao facto de existirem três casos de carga. O elevado número de barras que são agrupadas e o facto de existirem diversos casos de carga faz com que este problema, que possui apenas 29 variáveis de projecto independentes, seja formulado com 479 variáveis. Nestas circunstâncias, a formulação integrada apresenta uma grande desvantagem em relação às formulações que consideram os deslocamentos dos nós de um modo implícito.

No Quadro 6.18 encontram-se os valores finais da função objectivo que resultaram da aplicação do programa NEWTOP e dos algoritmos referidos em [Bel82]. O melhor resultado foi obtido com o programa NEWTOP, tendo alguns dos outros algoritmos apresentado resultados com um peso final semelhante. Com quatro dos oito programas testados em

[Bel82] não foi possível obter uma solução satisfatória. O programa NEWTOP forneceu a solução óptima a partir de uma solução constante, tendo sido necessário efectuar apenas um *rescaling*. A execução do programa NEWTOP a partir de outras soluções iniciais revelou a existência de um mínimo local cujo peso final excede apenas em 0.02% o melhor resultado obtido. A solução correspondente a este mínimo local apresenta como principal diferença o facto de as barras 77 a 80 (grupo 12) adoptarem uma área superior à mínima. No Quadro 6.19 encontra-se a melhor solução obtida com o programa NEWTOP e a melhor solução publicada em [Bel82], verificando-se que algumas barras apresentam áreas significativamente diferentes. A solução obtida com o programa NEWTOP encontra-se representada na Figura 6.38 por intermédio da substituição das barras por cilindros (ver secção 6.3.3.3).

Quadro 6.18 - Treliça de 200 barras - comparação entre a solução obtida com o programa NEWTOP e as soluções publicadas em [Bel82].

Programa	Peso final (lbf)	Método
NEWTOP	26124.041 -	método de Newton
M-5	26262.000 (+1%)	algoritmo de Fletcher
M-3	26600.000 (+2%)	algoritmo de Powell
M-4	26654.000 (+2%)	algoritmo de Powell
SUMT	27564.000 (+6%)	minimização sem restrições sequencial-penalidades externas
LINRM	33315.000 (+28%)	programação quadrática sequencial
CONMIN	34800.000 (+33%)	método das direcções admissíveis
GRP-UI	(1) -	método do gradiente projectado
OPTDYN	(2) -	método das direcções admissíveis

Notas provenientes de [Bel82]:

- (1) - Não foi possível obter a solução
- (2) - Não foi executado devido a insuficiência de memória

Quadro 6.19 - Treliza de 200 barras - comparação entre a solução obtida com o programa NEWTOP e a melhor solução publicada em [Bel82].

Grupo	1ª barra do grupo	NEWTOP Área (in ²)	M-5 Área (in ²)
1	1	0.149019	0.100000
2	5	0.950244	0.979285
3	19	0.100000	0.100000
4	18	0.100000	0.100000
5	26	1.956507	1.993239
6	6	0.301119	0.260768
7	39	0.100000	0.104881
8	43	3.154636	3.004996
9	57	0.100000	0.194936
10	64	4.169900	3.999125
11	44	0.402255	0.375500
12	77	0.100000	0.104881
13	81	5.470519	5.159651
14	95	0.100000	0.144914
15	102	6.495240	6.188619
16	82	0.531479	0.463681
17	115	0.453885	0.603324
18	119	8.105283	7.366750
19	133	0.100000	0.236643
20	140	9.140763	8.443222
21	120	0.880180	0.796241
22	153	0.153839	0.339116
23	157	11.270882	10.104454
24	171	0.100000	0.109545
25	178	12.319288	11.117554
26	158	0.932341	0.815475
27	191	6.941319	10.809718
28	195	11.220041	12.242957
29	196	14.315822	12.766362
Peso (lbf) →		26124.041	26262.000

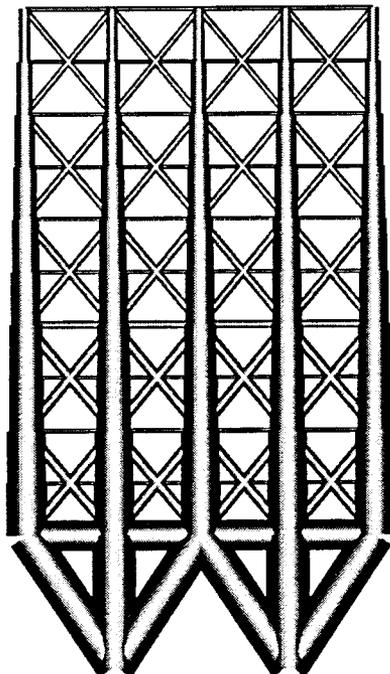


Figura 6.38 - Treliça de 200 barras - visualização da solução óptima obtida com o programa NEWTOP.

O programa NEWTOP necessitou de cerca de 20 minutos de CPU para minimizar o peso da treliça de 200 barras. Este tempo é da mesma ordem de grandeza dos tempos referidos em [Bel82] para os algoritmos que forneceram resultados com um peso semelhante. Uma vez que o computador utilizado em 1982 possuía certamente uma *performance* inferior à dos actuais, a causa da fraca eficiência do programa NEWTOP reside no facto de a formulação integrada dar origem neste problema a um número exagerado de variáveis.

6.3.4.3 - Treliça de 72 barras

Para que este estudo comparativo não fosse efectuado apenas com os exemplos publicados em [Bel82], foi também resolvido um problema de optimização cuja descrição se encontra em [Haf85] e [Haf87]. Este problema consiste na minimização do peso de uma treliça tridimensional, considerando as áreas das 72 barras como variáveis de projecto independentes. Não são consideradas restrições de deslocamento nem de encurvadura. Os dados deste problema encontra-se na Figura 6.39 e no Quadro 6.20. Os resultados obtidos com o programa NEWTOP encontram-se no Quadro 6.21 e na Figura 6.40. Em [Haf87] é

efectuado um estudo comparativo entre a formulação integrada e a convencional (*nested*). A primeira foi a que forneceu uma solução mais próxima da obtida com o programa NEWTOP, com um peso apenas 0.5% superior ao indicado no Quadro 6.21. O peso correspondente à solução obtida com a formulação convencional (*nested*) é 1.5% superior ao mínimo obtido com o programa NEWTOP. Neste problema foi simples obter a solução óptima partindo de uma solução constante, porque o valor da área mínima é relativamente elevado e um número considerável de barras adoptou esse valor mínimo. Esta justificação baseia-se apenas na experiência adquirida com a resolução de outros problemas de optimização.

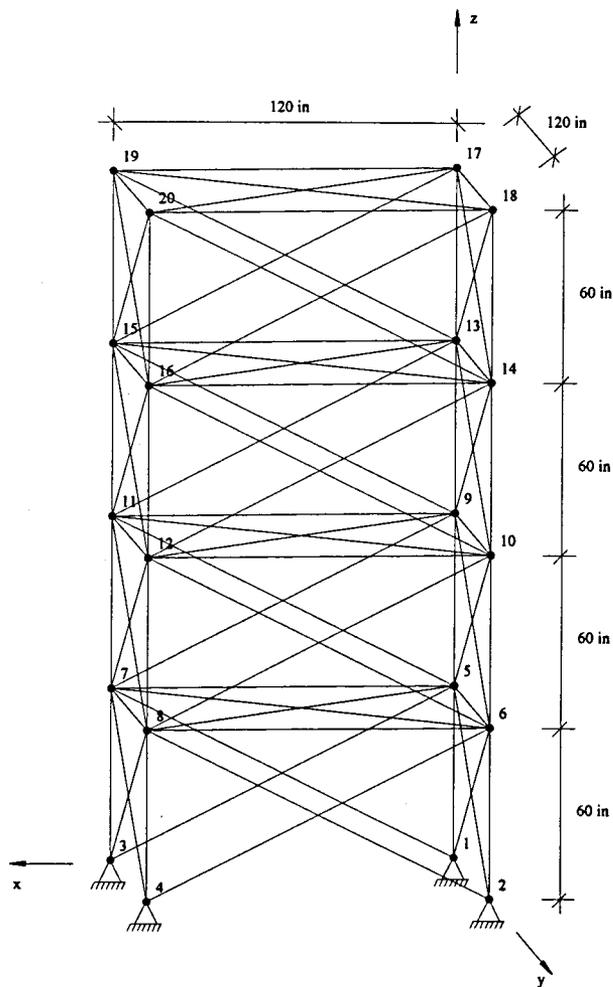


Figura 6.39 - Treliça de 72 barras - geometria e numeração dos nós.

Quadro 6.20 - Dados relativos ao problema de optimização da treliça de 72 barras.

Designação : tr72			
Objectivo : minimização do peso			
Restrições : área mínima e tensão mínima/máxima			
Nota: estas restrições foram consideradas em todas as barras.			
Unidade de força	:	libra força (lbf)	(1 lbf = 4.448 N)
Unidade de comprimento	:	polegada (in)	(1 in = 0.0254 m)
Número de barras	-----	72	
Número de variáveis de projecto independentes	----	72	
Número de graus de liberdade não prescritos	-----	48 (3*16)	
Número de restrições desigualdade	-----	216 (72+2*72)	
Módulo de Young (E)	:	10e6	lbf/in ²
Peso específico (γ)	:	0.1	lbf/in ³
Área mínima (a min)	:	0.1	in ²
Tensão mínima/máxima (σ min/max)	:	-25e3, 25e3	lbf/in ²
Deslocamento mínimo/máximo (d min/max)	:	-	in
Melhor solução conhecida (C)	:	491.320	lbf
Acções no único caso de carga:			
Peso próprio	+ força de	50000 lbf	segundo x no nó 17 +
	+ força de	50000 lbf	segundo y no nó 17 +
	+ força de	-50000 lbf	segundo z no nó 17 +
	+ força de	-50000 lbf	segundo z no nó 18 +
	+ força de	-50000 lbf	segundo z no nó 19 +
	+ força de	-50000 lbf	segundo z no nó 20

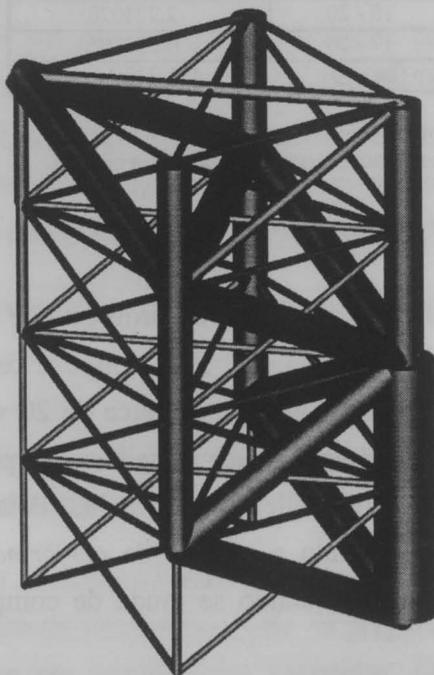


Figura 6.40 - Treliça de 72 barras - visualização da solução óptima obtida com o programa NEWTOP.

Quadro 6.21 - Treliça de 72 barras - solução óptima obtida com o programa NEWTOP.

Barra	Área (in ²)
2 - 6	0.138749
3 - 7	0.138749
4 - 6	1.997709
4 - 7	1.997709
4 - 8	6.087992
6 - 10	1.828149
6 - 12	1.948747
7 - 11	1.828149
7 - 12	1.948747
8 - 10	0.133057
8 - 11	0.133057
8 - 12	5.891305
9 - 13	0.229867
10 - 14	1.860300
11 - 15	1.860300
12 - 14	1.964234
12 - 15	1.964234
12 - 16	2.388254
13 - 17	0.344922
14 - 15	2.141696
14 - 17	1.851784
14 - 18	1.977682
14 - 20	0.250043
15 - 17	1.851784
15 - 19	1.977682
15 - 20	0.250043
16 - 20	2.224070
17 - 20	0.373205
Peso (lbf) →	491.320

Nota:

- as restantes barras possuem área mínima.

Com o programa NEWTOP, a resolução do problema de optimização da treliça de 72 barras demorou aproximadamente 5 segundos. Este tempo é cerca de 50 vezes inferior ao referido em [Haf87] e foi obtido num computador cerca de 20 vezes mais rápido. Assim se conclui que, em computadores com *performance* semelhante, o programa NEWTOP seria 2.5 vezes mais rápido na optimização da treliça de 72 barras. Esta conclusão apenas é válida nestas circunstâncias concretas, podendo a relação de *performance* entre dois métodos de optimização variar significativamente quando se muda de computador, de problema ou de solução inicial.

6.3.4.4 - Conclusões do estudo comparativo

A resolução de um problema de optimização de estruturas por diversos métodos permite tirar algumas conclusões sobre as qualidades e os defeitos desses métodos e dos respectivos códigos computacionais. Este estudo comparativo restringiu-se aos casos de minimização do peso de treliças, mas englobou diversos métodos de optimização utilizados por diferentes autores. Apresentam-se em seguida as conclusões baseadas na resolução dos três problemas atrás descritos. As conclusões gerais relativas à utilização do programa NEWTOP serão incluídas no último capítulo deste trabalho.

Nas três treliças estudadas o programa NEWTOP forneceu sempre a melhor solução. No caso da treliça de 10 barras nenhum dos métodos alternativos se aproximou do melhor resultado, tendo nalguns casos ocorrido uma convergência para mínimos e máximos locais. Nas restantes treliças quase todos os métodos forneceram soluções aceitáveis. A elevada precisão que as soluções obtidas com o programa NEWTOP apresentam dificilmente pode ser alcançada com métodos de primeira ordem. Os resultados fornecidos por estes métodos apresentam uma dispersão considerável que revela, para além da falta de precisão, uma falta de capacidade na distinção de mínimos locais próximos entre si.

A eficiência de um método de optimização e do código computacional que lhe está associado é difícil de avaliar porque depende de inúmeros factores. Os exemplos resolvidos mostram que o programa NEWTOP possui uma eficiência aceitável, exceptuando as situações em que a existência de múltiplos casos de carga conduz a um exagerado número de variáveis. Na secção 6.3.3 foram apresentados exemplos que mostram que o programa NEWTOP é capaz de resolver problemas de optimização com um número de variáveis muito elevado.

O programa NEWTOP apresentou uma boa fiabilidade porque resolveu todos os problemas de optimização abordados. Existe quase sempre a necessidade de o executar a partir de múltiplas soluções iniciais, principalmente nos casos em que não for efectuado um pré-dimensionamento da estrutura. Alguns dos métodos testados em [Bel82] divergem ou fornecem soluções muito distantes do mínimo global.

De um modo geral, a mais importante intervenção que o utilizador de um programa de optimização de estruturas tem de efectuar consiste na preparação da solução inicial e da sua eventual alteração com base em corridas mal sucedidas. O utilizador deve também estar preparado para modificar um ou outro parâmetro que influencie o comportamento do processo iterativo. Supõe-se que a necessidade de efectuar estas intervenções é comum a todos os métodos de optimização de estruturas.

CAPÍTULO 7

OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO NÃO LINEAR

Quer a análise, quer a optimização de estruturas com comportamento não linear pode ser efectuada recorrendo à programação matemática. Quando a hipótese das pequenas deformações não é considerada válida, o comportamento da estrutura é não linear geométrico e a sua análise pode ser formulada como um problema de minimização da energia [Van87]. A resolução deste tipo de problemas pode ser efectuada com um dos métodos de optimização referidos no Capítulo 3. Quando o material que constitui a estrutura apresenta uma fase perfeitamente plástica, é em geral possível formular o problema correspondente ao cálculo do máximo factor de carga como um programa linear [Ser88]. Refere-se como exemplo o clássico problema do cálculo da carga de colapso de um pórtico, admitindo a possibilidade da formação de rótulas plásticas [Nea77]. Nestas circunstâncias a minimização do peso da estrutura pode também ser formulada como um programa linear [Nea77]. A resolução desta classe de programas matemáticos pode ser efectuada recorrendo por exemplo ao método *simplex* [Lue84]. Esta abordagem clássica da análise ou optimização de estruturas porticadas com comportamento não linear apresenta algumas desvantagens, como por exemplo a necessidade de linearizar a superfície de cedência e a incerteza na localização das rótulas plásticas quando existem cargas distribuídas [Tin93]. De um modo geral, na abordagem da optimização de estruturas com programação linear existe ainda o inconveniente de não ser possível considerar restrições de deslocamento ou de deformação.

A abordagem com mais generalidade da optimização de estruturas com comportamento não linear encontra-se ainda pouco desenvolvida, devido ao facto de só recentemente ter sido investigada [Aro87a]. Em [WuA87] é proposto um algoritmo baseado na análise de sensibilidades e que consiste na aplicação sucessiva das seguintes fases:

- 1 - análise não linear da estrutura;
- 2 - selecção das restrições que são consideradas activas;

- 3 - análise de sensibilidades, i.e., cálculo das derivadas das restrições em ordem às variáveis de projecto;
- 4 - modificação da solução recorrendo a um algoritmo de optimização.

Este algoritmo é semelhante a um dos que foi referido na introdução do Capítulo 6, apresentando apenas a particularidade de ter sido adaptado a estruturas com comportamento não linear. Em [WuA87] são efectuadas algumas aplicações relativas à optimização de treliças com comportamento não linear material e geométrico.

O desenvolvimento da formulação que é proposta neste capítulo foi em parte motivado pela frequente necessidade de projectar estruturas cujo comportamento não é perfeitamente elástico nem perfeitamente plástico. Com este objectivo, é sugerido em [Bla93] que no projecto de uma viga contínua sejam arbitrados os valores dos momentos em possíveis rótulas plásticas e em seguida verificados os valores das rotações relativas nessas rótulas. Este procedimento deve ser sucessivamente aplicado até se atingir uma solução que o projectista considere satisfatória. Deste modo é alcançada uma solução final que não apresenta as desvantagens das soluções elásticas (momentos flectores negativos exagerados), nem os inconvenientes das soluções perfeitamente plásticas (deformações plásticas inadmissíveis). O método proposto em [Bla93] é no entanto demasiado dependente de uma interacção programa de computador/projectista e na generalidade dos casos não conduz à solução mais económica. Com a formulação em seguida apresentada é possível efectuar a análise e optimização simultânea de vigas contínuas com comportamento elastoplástico, sendo consideradas restrições relativas a momentos flectores e a deformações plásticas. Deste modo são eliminados alguns dos inconvenientes atrás referidos.

7.1 - FORMULAÇÃO DO COMPORTAMENTO ELASTOPLÁSTICO

A formulação destinada ao estudo de vigas contínuas com comportamento elastoplástico pode ser efectuada independentemente de ter em vista a análise ou a optimização de estruturas. Entre estes dois casos é apenas necessário distinguir a expressão da função objectivo e considerar diferentes grandezas como fixas ou variáveis. No presente estudo são consideradas algumas hipóteses simplificativas destinadas a facilitar a formulação do problema, não sendo contudo alterados os aspectos fundamentais do respectivo significado físico. Algumas das simplificações são justificadas pelo facto de neste trabalho apenas se pretender avaliar a aplicabilidade do programa NEWTOP à resolução de problemas de análise

e optimização de estruturas com comportamento elastoplástico. A formulação considerada no presente trabalho pode ser facilmente desenvolvida, de modo a serem contemplados alguns aspectos importantes que têm de ser incluídos no projecto de estruturas reais (e.g., coeficientes de segurança, múltiplos casos de carga, outros tipos de estrutura).

Na simbologia da formulação que é em seguida apresentada apenas são utilizados caracteres dos intervalos $[a, z]$ e $[A, Z]$, porque as expressões que constituem o programa matemático são manipuladas simbolicamente pelo programa de computador (ver Capítulo 5) e este não admite os tradicionais caracteres do alfabeto grego. Por este motivo é aconselhada uma consulta da simbologia sempre que necessário.

Nas vigas contínuas que são em seguida estudadas é admitido um comportamento linear elástico - perfeitamente plástico com deformações limitadas. Na Figura 7.1 encontra-se representado o diagrama momento-curvatura considerado [Nea77].

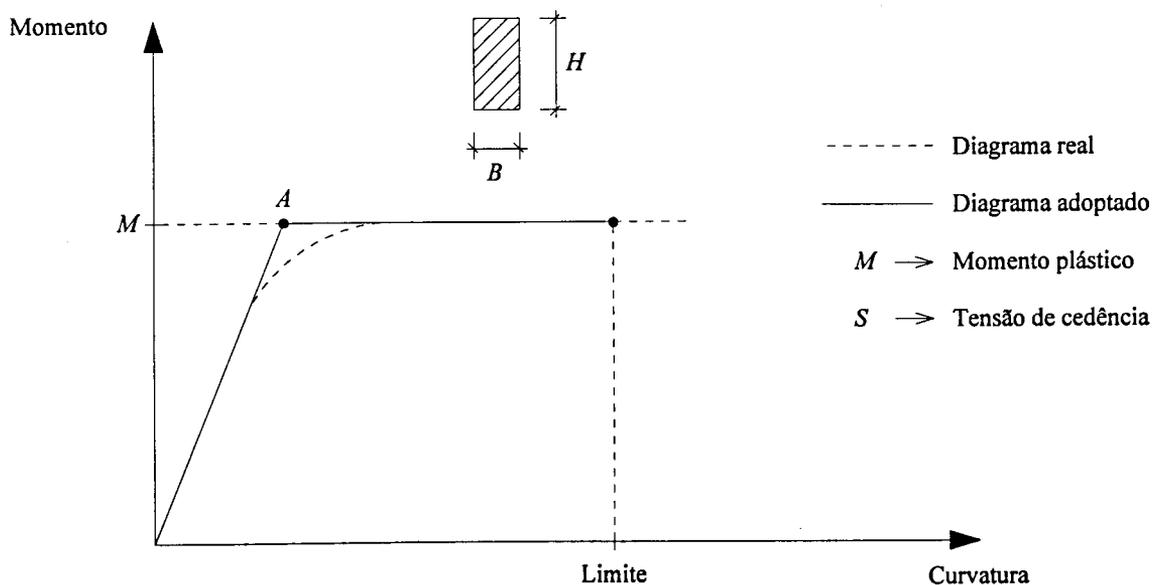


Figura 7.1 - Diagrama momento-curvatura.

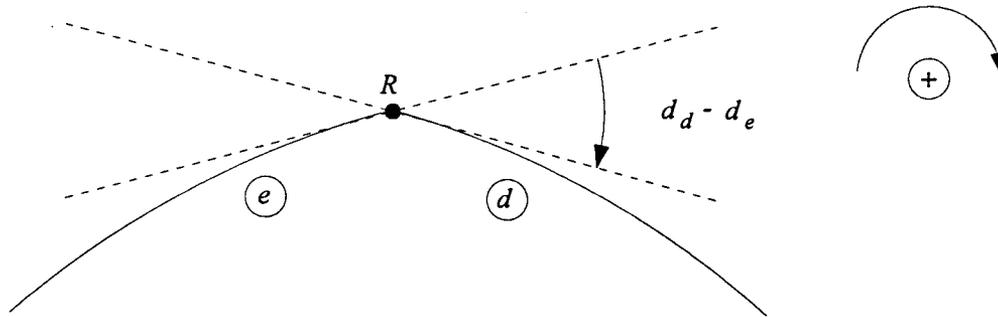
O valor do momento plástico correspondente à secção rectangular representada na Figura 7.1 é fornecido pela seguinte expressão [Maj72]

$$M = S \frac{BH^2}{4} \quad (7.1)$$

Na Figura 7.2 encontra-se definida a rotação relativa que ocorre numa rótula plástica depois de alcançado o ponto A da Figura 7.1. Quando é considerado um comportamento

perfeitamente plástico, não é imposto qualquer limite ao valor de $d_d - d_e$. Na presente formulação é considerada a seguinte restrição em todas as rótulas plásticas

$$-e \leq d_d - d_e \leq e \quad (7.2)$$

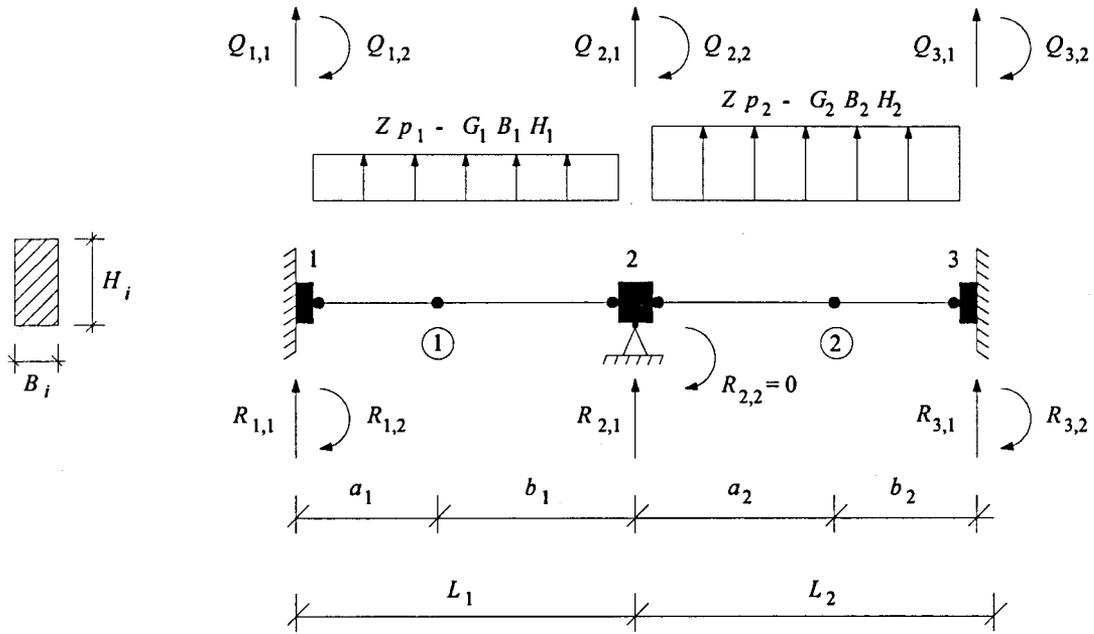


R → rótula plástica
 d_e → rotação à esquerda da rótula
 d_d → rotação à direita da rótula
 $d_d - d_e$ → rotação relativa numa rótula plástica

Figura 7.2 - Rotação relativa numa rótula plástica.

O valor a atribuir ao limite e é da responsabilidade do utilizador e depende essencialmente do material utilizado, de aspectos regulamentares e de informação proveniente de comparações com resultados experimentais. No presente trabalho e no caso de uma viga contínua em aço é atribuído ao parâmetro e o valor adimensional 0.01 .

Na Figura 7.3 encontra-se o esquema estrutural considerado na análise e na optimização de vigas contínuas com comportamento elastoplástico. O exemplo representado na figura apresenta dois tramos (L_1 e L_2), sendo cada um deles constituído por duas barras (a_i e b_i). Em cada tramo é considerada uma secção rectangular constante e uma carga distribuída ($Z p_i - G_i B_i H_i$) também constante, sendo Z um factor de carga positivo e G_i o peso específico do material. É também possível aplicar cargas concentradas $Q_{j,i}$ nos nós. Se existirem cargas concentradas nos tramos, deve ser considerado um nó auxiliar em cada ponto de aplicação. Estes nós auxiliares possuem as características do nó 2, exceptuando o facto de não serem ligados ao exterior.



Nota: nas extremidades pode ser considerado um apoio duplo em vez do encastramento.

Figura 7.3 - Esquema estrutural de uma viga contínua com comportamento elastoplástico.

Em cada tramo i são considerados os deslocamentos e as forças indicados na Figura 7.4.

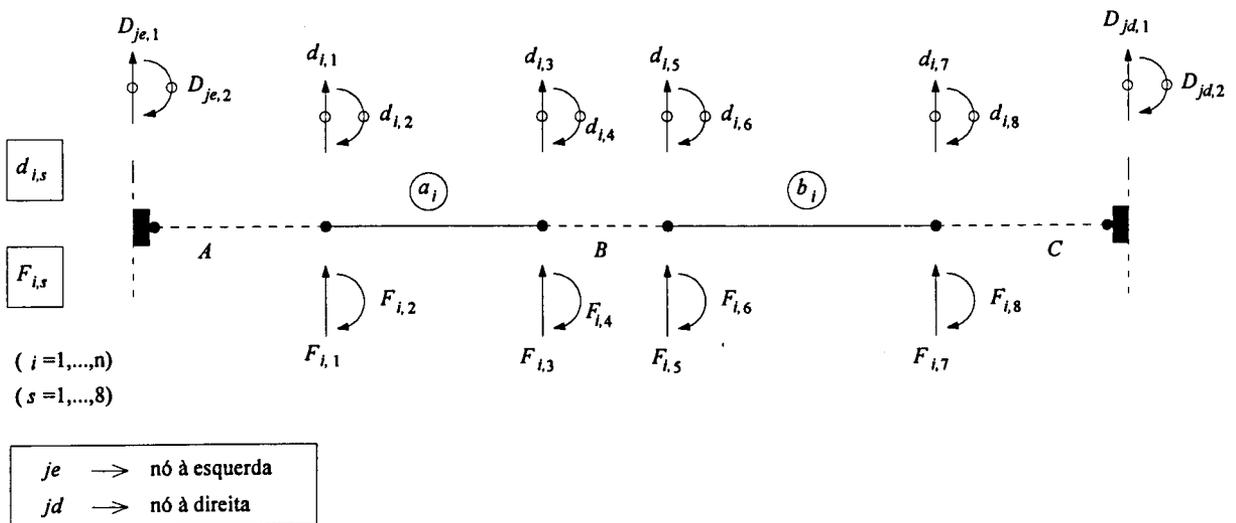


Figura 7.4 - Deslocamentos e forças no tramo i da viga contínua.

No tramo a da barra i devem ser consideradas as seguintes equações de equilíbrio (por conveniência de notação o índice i não é representado)

$$\underset{\sim}{K} \underset{\sim}{d} = \underset{\sim}{P} + \underset{\sim}{F} \quad (7.3)$$

$$\underset{\sim}{K} = EI \begin{bmatrix} \frac{12}{a^3} & -\frac{6}{a^2} & -\frac{12}{a^3} & -\frac{6}{a^2} \\ -\frac{6}{a^2} & \frac{4}{a} & \frac{6}{a^2} & \frac{2}{a} \\ -\frac{12}{a^3} & \frac{6}{a^2} & \frac{12}{a^3} & \frac{6}{a^2} \\ -\frac{6}{a^2} & \frac{2}{a} & \frac{6}{a^2} & \frac{4}{a} \end{bmatrix} \quad (7.4)$$

$$\underset{\sim}{P} = (Z \ p - G \ B \ H) \begin{bmatrix} \frac{a}{2} \\ -\frac{a^2}{12} \\ \frac{a}{2} \\ \frac{a^2}{12} \end{bmatrix} \quad (7.5)$$

Quando se refere à barra a , a equação de equilíbrio (7.3) apenas relaciona as forças e os deslocamentos dos graus de liberdade 1 a 4. No caso da barra b , as equações de equilíbrio obtêm-se substituindo em (7.3) a (7.5) o comprimento da barra a pelo da barra b e considerando os graus de liberdade 5 a 8.

Em cada nó j são considerados os deslocamentos e as forças indicadas na Figura 7.5.

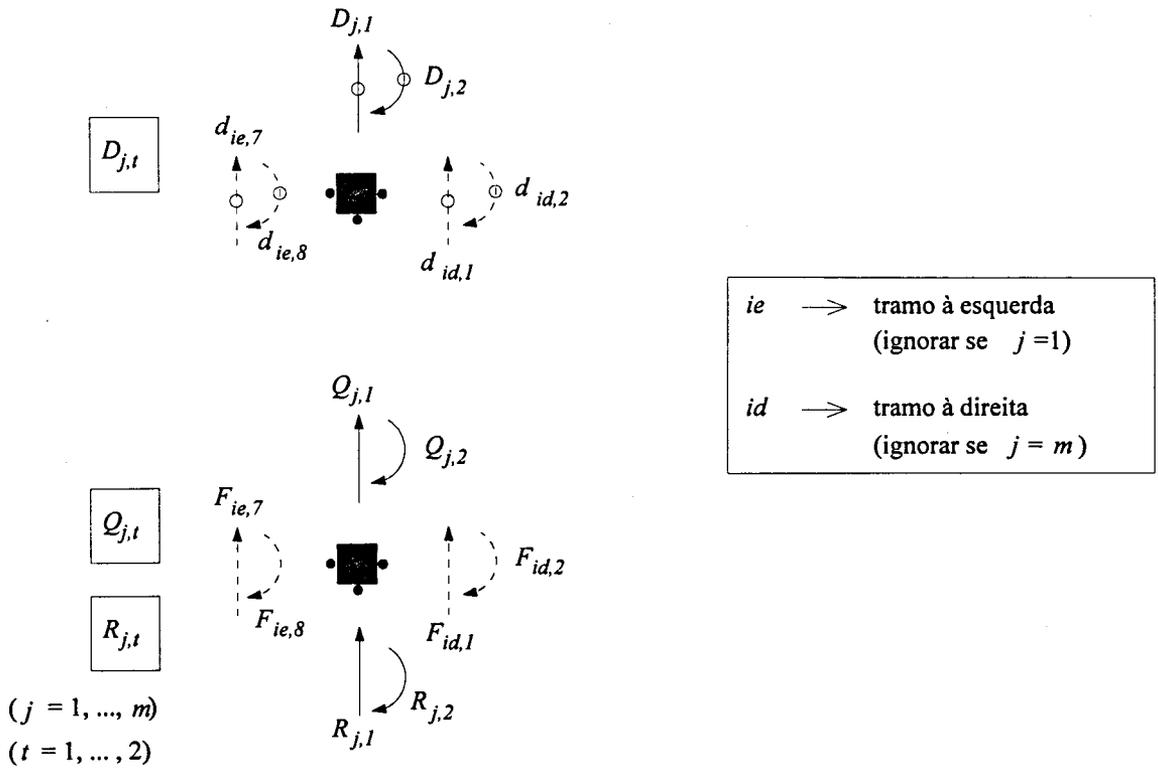


Figura 7.5 - Deslocamentos e forças num dos nós da viga contínua.

Em cada nó j devem ser consideradas as seguintes equações de equilíbrio

$$Q_{j,1} + R_{j,1} = F_{ie,7} + F_{id,1} \tag{7.6a}$$

$$Q_{j,2} + R_{j,2} = F_{ie,8} + F_{id,2} \tag{7.6b}$$

As seguintes equações garantem a existência de compatibilidade entre as diversas barras que constituem a viga contínua (ver Figura 7.4)

$$d_{i,1} = D_{je,1} \tag{7.7a}$$

$$d_{i,5} = d_{i,3} \tag{7.7b}$$

$$d_{i,7} = D_{jd,1} \tag{7.7c}$$

Em cada nó de transição entre a barra a e a barra b (ponto B na Figura 7.4) devem ser impostas as seguintes restrições

$$F_{i,3} = 0 \tag{7.8a}$$

$$F_{i,5} = 0 \tag{7.8b}$$

As restrições (7.8) obrigam o ponto B (ver Figura 7.4) a coincidir com o ponto em que o esforço transversal é nulo e o momento flector é máximo. Qualquer solução que respeite (7.8) respeita também a seguinte equação de equilíbrio

$$F_{i,3} + F_{i,5} = 0 \quad (7.9)$$

Se num determinado tramo não existir nenhum ponto de esforço transversal nulo, as restrições (7.8) têm de ser substituídas pela equação (7.9) e deve ser acrescentada a seguinte equação destinada a definir a posição do ponto B

$$a_i = b_i \quad (7.10)$$

Uma vez que nestas circunstâncias o ponto B não é um ponto de momento máximo, as respectivas restrições não influenciam o resultado final. Esta questão requer uma intervenção criteriosa do utilizador do programa de computador e em certos casos particulares pode causar alguns problemas. Em futuros desenvolvimentos da formulação este problema deverá ser eliminado.

No ponto B de cada tramo i deve ser ainda considerada a seguinte equação de equilíbrio de momentos

$$F_{i,4} + F_{i,6} = 0 \quad (7.11)$$

As seguintes restrições são necessárias para estabelecer algumas relações entre as diversas variáveis

$$L_i = a_i + b_i \quad (7.12)$$

$$I_i = \frac{B_i H_i^3}{12} \quad (7.13)$$

$$M_i = S_i \frac{B_i H_i^2}{4} \quad (7.14)$$

Nos graus de liberdade cujo deslocamento é nulo deve ser imposta a restrição

$$D_{j,t} = 0 \quad (7.15)$$

Nos restantes graus de liberdade a reacção é nula, devendo ser nesses casos considerada a restrição

$$R_{j,t} = 0 \quad (7.16)$$

De acordo com a Figura 7.4 e a equação (7.11) devem ser limitados os valores dos momentos nos pontos A , B e C .

$$-M_i \leq F_{i,2} \leq M_i \quad (7.17a)$$

$$-M_i \leq F_{i,4} \leq M_i \quad (7.17b)$$

$$-M_i \leq F_{i,8} \leq M_i \quad (7.17c)$$

Atendendo às características da presente formulação, a limitação do momento flector nestes três pontos garante que o respectivo valor admissível não é excedido em nenhum ponto do tramo.

As seguintes restrições destinam-se a impedir que alguma barra adopte um valor demasiado pequeno ou negativo

$$a_i \geq \frac{L_i}{100} \quad (7.18a)$$

$$b_i \geq \frac{L_i}{100} \quad (7.18b)$$

Para contemplar a complementaridade entre o comportamento elástico e o comportamento plástico (ver Figura 7.1) deve ser considerada em cada rótula plástica uma restrição que garanta a verificação de pelo menos uma das seguintes condições

$$M - F = 0 \quad (7.19a)$$

$$M + F = 0 \quad (7.19b)$$

$$d_d - d_e = 0 \quad (7.19c)$$

Em (7.19) M é o momento plástico (7.1), F é um dos momentos F_2 , F_4 ou F_8 (ver Figura 7.4, e (7.11)), d_e é a rotação à esquerda da rótula (ver Figura 7.2), d_d é a rotação à direita e $(d_d - d_e)$ é a rotação relativa na rótula plástica. A restrição que garante a verificação de pelo menos uma das condições (7.19) é a seguinte

$$(M - F)(M + F)(d_d - d_e) = 0 \quad (7.20)$$

Esta restrição contempla a possibilidade de F ser positivo ou negativo e, atendendo a (7.17), garante uma complementaridade entre as seguintes situações

$$\begin{array}{l} \text{Comportamento elástico} \\ \text{Comportamento plástico} \end{array} \left\{ \begin{array}{l} -M \leq F \leq M \\ e \\ d_d - d_e = 0 \\ F = -M \text{ ou } F = M \\ e \\ d_d - d_e \neq 0 \end{array} \right.$$

A seguinte restrição é equivalente a (7.20) e apresenta as características necessárias para ser interpretada pelo programa NEWTOP (ver secção 4.2)

$$M^2 d_d - M^2 d_e - F^2 d_d + F^2 d_e = 0 \quad (7.21)$$

A formulação apresentada nesta secção é comum à análise e à optimização de vigas contínuas com comportamento elastoplástico. Nas secções que se seguem são apresentados os aspectos específicos de cada uma das formulações e são apresentados alguns exemplos numéricos.

7.2 - ANÁLISE DE ESTRUTURAS COM COMPORTAMENTO ELASTOPLÁSTICO

A análise de uma estrutura com as características e restrições indicadas na secção 7.1 pode adoptar uma das seguintes formas:

Problema P1 - calcular o máximo factor de carga (Z) e o comportamento da estrutura (e.g., deslocamentos, esforços) para esse valor de Z .

Problema P2 - calcular o comportamento da estrutura para um valor fixo de Z .

O problema P2 será mais adiante apresentado como um caso particular do problema P1. O cálculo do máximo factor de carga é um problema cujo objectivo é o seguinte

$$\text{Maximizar } Z \quad (7.22)$$

Para que o programa NEWTOP possa ser utilizado na forma *standard*, deve ser considerada a seguinte formulação equivalente a (7.22)

$$\text{Min. } f(\tilde{x}) = -Z \quad (7.23)$$

As grandezas que neste tipo de problema são previamente fixadas são as seguintes (ver a Simbologia)

$$\left(\underline{L}, \underline{E}, \underline{G}, \underline{S}, \underline{e}, \underline{B}, \underline{H}, \underline{I}, \underline{M}, \underline{p}, \underline{Q} \right) \quad (7.24)$$

Os valores destas variáveis são substituídos pelo programa NEWTOP em todas as funções que constituem o programa matemático (ver secção 5.3). No problema P1 as variáveis de projecto são as seguintes

$$\left(\underline{Z}, \underline{a}, \underline{b}, \underline{d}, \underline{D}, \underline{F}, \underline{R} \right) \quad (7.25)$$

No Quadro 7.1 é indicado o número de variáveis de projecto no caso de uma viga contínua com n tramos e m nós (ver o exemplo da Figura 7.3).

Quadro 7.1 - Problema P1 - número de variáveis de projecto.

Variável de projecto	Quantidade
Z	1
a	n
b	n
d	$8 n$
D	$2 m$
F	$8 n$
R	$2 m$
Total	$1 + 18 n + 4 m$

No Quadro 7.2 encontra-se a identificação e a contagem das restrições, de acordo com a formulação indicada na secção 7.1.

Quadro 7.2 - Problema P1 - número de restrições.

Tipo de restrição		Quantidade	Total
Desigualdade	(7.17)	$6n$	$14n$
	(7.18)	$2n$	
	(7.2)	$6n$	
Igualdade	(7.3)	$8n$	$18n + 4m$
	(7.6)	$2m$	
	(7.21)	$3n$	
	(7.12)	n	
	(7.7)	$3n$	
	(7.8)	$2n$	
	(7.11)	n	
	(7.15) ou (7.16)	$2m$	

Uma vez que existem $18n + 4m$ restrições igualdade, é possível considerar Z como variável independente e considerar as restantes variáveis do Quadro 7.1 como variáveis dependentes. As $18n + 4m$ restrições igualdade indicadas no Quadro 7.2 constituem, depois de fixado o valor de Z , um sistema de equações não lineares com $18n + 4m$ incógnitas. A resolução deste sistema de equações com o objectivo de calcular o valor inicial das variáveis dependentes é contudo desaconselhado, porque as equações (7.21) só têm um significado físico correcto se forem acompanhadas das restrições desigualdade (7.17) e durante a resolução do referido sistema de equações estas restrições não são consideradas. Por este motivo, sugere-se que o cálculo do valor inicial das variáveis dependentes não seja efectuado pelo programa NEWTOP e que seja fornecida uma solução inicial, eventualmente inadmissível, em que nenhuma restrição desigualdade se encontre activa. De acordo com a experiência acumulada na resolução de alguns problemas, é aconselhada uma solução inicial à qual corresponda uma rotação plástica não nula em todas as rótulas. Apesar de esta solução inicial ser na generalidade dos casos inadmissível, verifica-se ser assim mais provável a obtenção do máximo factor de carga Z .

Com o objectivo de clarificar as características da formulação proposta e validar os resultados obtidos é efectuada uma comparação com o problema de maximização do factor de carga indicado na Figura 7.6 [Tin93].

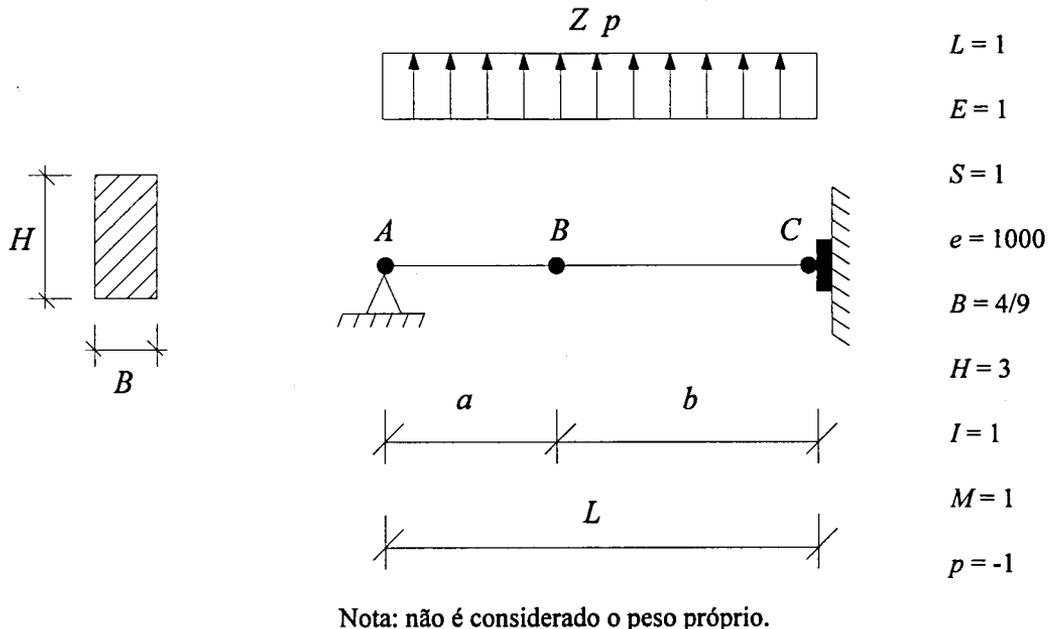


Figura 7.6 - Problema de maximização do factor de carga Z [Tin93].

Em [Tin93] é indicada como solução exacta o seguinte valor de Z

$$Z = 11.6569 \quad (7.26)$$

Com a formulação atrás indicada e recorrendo ao programa NEWTOP para a resolução do correspondente programa matemático foi confirmada a solução (7.26).

Devido ao facto de o volume de informação associado à resolução deste problema com o programa NEWTOP não ser muito elevado, são em seguida apresentados os dados e os resultados mais significativos. Na Listagem 7.1 encontra-se o programa matemático correspondente à maximização do factor de carga da estrutura representada na Figura 7.6. A função objectivo corresponde a (7.23) e as restrições encontram-se enumeradas no Quadro 7.2. As restrições igualdade que podem ser directamente substituídas no programa matemático encontram-se na Listagem 7.2 (ver também a secção 5.3). Na Listagem 7.3 são apresentados os resultados obtidos com o programa NEWTOP. A folga nula nas restrições $M_1_M_min$ e $M_1_R_max$ indica a formação de rótulas plásticas nos pontos B e C da viga representada na

Figura 7.6. O valor da variável a que define a posição da rótula B só é conhecido após a resolução do programa matemático.

Listagem 7.1 - Maximização do factor de carga de uma viga - programa matemático.

```

### Main title of the nonlinear program
Maximizacao do factor de carga de uma viga

Min.
    - Z ;

s.t.i.c.

M_1_L_min: - F_1_2 - M1 < 0 ;
M_1_L_max: + F_1_2 - M1 < 0 ;
M_1_M_min: - F_1_4 - M1 < 0 ;
M_1_M_max: + F_1_4 - M1 < 0 ;
M_1_R_min: - F_1_8 - M1 < 0 ;
M_1_R_max: + F_1_8 - M1 < 0 ;

a1_min: - a1 + 0.01 * L1 < 0 ;
b1_min: - b1 + 0.01 * L1 < 0 ;

Rot_1_L_min: - e1 + D_1_2 - d_1_2 < 0 ;
Rot_1_L_max: - e1 - D_1_2 + d_1_2 < 0 ;
Rot_1_M_min: - e1 + d_1_4 - d_1_6 < 0 ;
Rot_1_M_max: - e1 - d_1_4 + d_1_6 < 0 ;
Rot_1_R_min: - e1 + d_1_8 - D_2_2 < 0 ;
Rot_1_R_max: - e1 - d_1_8 + D_2_2 < 0 ;

s.t.e.c.

Equil_a1_1: + 1.2000000000e+01 * a1 ^ -3 * E1 * I1 * d_1_1
            + -6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_2
            + -1.2000000000e+01 * a1 ^ -3 * E1 * I1 * d_1_3
            + -6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_4
            + -5.0000000000e-01 * Z * a1 ^ 1 * p1
            - -5.0000000000e-01 * a1 ^ 1 * G1 * B1 * H1 - F_1_1 = 0 ;

Equil_a1_2: + -6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_1
            + 4.0000000000e+00 * a1 ^ -1 * E1 * I1 * d_1_2
            + 6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_3
            + 2.0000000000e+00 * a1 ^ -1 * E1 * I1 * d_1_4
            + 8.3333333333e-02 * Z * a1 ^ 2 * p1
            - 8.3333333333e-02 * a1 ^ 2 * G1 * B1 * H1 - F_1_2 = 0 ;

Equil_a1_3: + -1.2000000000e+01 * a1 ^ -3 * E1 * I1 * d_1_1
            + 6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_2
            + 1.2000000000e+01 * a1 ^ -3 * E1 * I1 * d_1_3
            + 6.0000000000e+00 * a1 ^ -2 * E1 * I1 * d_1_4
            + -5.0000000000e-01 * Z * a1 ^ 1 * p1
            - -5.0000000000e-01 * a1 ^ 1 * G1 * B1 * H1 - F_1_3 = 0 ;

```

$$\begin{aligned}
 \text{Equil_a1_4:} & + -6.0000000000e+00 * a1 \wedge -2 * E1 * I1 * d_{1_1} \\
 & + 2.0000000000e+00 * a1 \wedge -1 * E1 * I1 * d_{1_2} \\
 & + 6.0000000000e+00 * a1 \wedge -2 * E1 * I1 * d_{1_3} \\
 & + 4.0000000000e+00 * a1 \wedge -1 * E1 * I1 * d_{1_4} \\
 & + -8.3333333333e-02 * Z * a1 \wedge 2 * p1 \\
 & - -8.3333333333e-02 * a1 \wedge 2 * G1 * B1 * H1 - F_{1_4} = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Equil_b1_1:} & + 1.2000000000e+01 * b1 \wedge -3 * E1 * I1 * d_{1_5} \\
 & + -6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_6} \\
 & + -1.2000000000e+01 * b1 \wedge -3 * E1 * I1 * d_{1_7} \\
 & + -6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_8} \\
 & + -5.0000000000e-01 * Z * b1 \wedge 1 * p1 \\
 & - -5.0000000000e-01 * b1 \wedge 1 * G1 * B1 * H1 - F_{1_5} = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Equil_b1_2:} & + -6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_5} \\
 & + 4.0000000000e+00 * b1 \wedge -1 * E1 * I1 * d_{1_6} \\
 & + 6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_7} \\
 & + 2.0000000000e+00 * b1 \wedge -1 * E1 * I1 * d_{1_8} \\
 & + 8.3333333333e-02 * Z * b1 \wedge 2 * p1 \\
 & - 8.3333333333e-02 * b1 \wedge 2 * G1 * B1 * H1 - F_{1_6} = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Equil_b1_3:} & + -1.2000000000e+01 * b1 \wedge -3 * E1 * I1 * d_{1_5} \\
 & + 6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_6} \\
 & + 1.2000000000e+01 * b1 \wedge -3 * E1 * I1 * d_{1_7} \\
 & + 6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_8} \\
 & + -5.0000000000e-01 * Z * b1 \wedge 1 * p1 \\
 & - -5.0000000000e-01 * b1 \wedge 1 * G1 * B1 * H1 - F_{1_7} = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Equil_b1_4:} & + -6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_5} \\
 & + 2.0000000000e+00 * b1 \wedge -1 * E1 * I1 * d_{1_6} \\
 & + 6.0000000000e+00 * b1 \wedge -2 * E1 * I1 * d_{1_7} \\
 & + 4.0000000000e+00 * b1 \wedge -1 * E1 * I1 * d_{1_8} \\
 & + -8.3333333333e-02 * Z * b1 \wedge 2 * p1 \\
 & - -8.3333333333e-02 * b1 \wedge 2 * G1 * B1 * H1 - F_{1_8} = 0 ;
 \end{aligned}$$

$$\text{Equil_n1_1:} + Q_{1_1} + R_{1_1} - F_{1_1} = 0 ;$$

$$\text{Equil_n1_2:} + Q_{1_2} + R_{1_2} - F_{1_2} = 0 ;$$

$$\text{Equil_n2_1:} + Q_{2_1} + R_{2_1} - F_{1_7} = 0 ;$$

$$\text{Equil_n2_2:} + Q_{2_2} + R_{2_2} - F_{1_8} = 0 ;$$

$$\begin{aligned}
 \text{Compl_1_L:} & + M1 \wedge 2 * d_{1_2} - M1 \wedge 2 * D_{1_2} \\
 & - d_{1_2} * F_{1_2} \wedge 2 + D_{1_2} * F_{1_2} \wedge 2 = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Compl_1_M:} & + M1 \wedge 2 * d_{1_6} - M1 \wedge 2 * d_{1_4} \\
 & - d_{1_6} * F_{1_4} \wedge 2 + d_{1_4} * F_{1_4} \wedge 2 = 0 ;
 \end{aligned}$$

$$\begin{aligned}
 \text{Compl_1_R:} & + M1 \wedge 2 * D_{2_2} - M1 \wedge 2 * d_{1_8} \\
 & - D_{2_2} * F_{1_8} \wedge 2 + d_{1_8} * F_{1_8} \wedge 2 = 0 ;
 \end{aligned}$$

$$\text{L1=a1+b1:} \quad L1 - a1 - b1 = 0 ;$$

END_OF_FILE

Listagem 7.2 - Maximização do factor de carga de uma viga - variáveis que vão ser substituídas no programa matemático.

```

L1 = 1.0
E1 = 1.0
G1 = 0.0
S1 = 1.0
e1 = 1000.0
B1 = 0.4444444444444444
H1 = 3.0
I1 = 1.0
M1 = 1.0
p1 = -1.0

Q_1_1 = 0.0
Q_1_2 = 0.0
Q_2_1 = 0.0
Q_2_2 = 0.0

d_1_1 = 1.0 * D_1_1
d_1_5 = 1.0 * d_1_3
d_1_7 = 1.0 * D_2_1
F_1_3 = 0.0
F_1_5 = 0.0
F_1_6 = -1.0 * F_1_4

D_1_1 = 0.0
R_1_2 = 0.0

D_2_1 = 0.0
D_2_2 = 0.0

```

Listagem 7.3 - Maximização do factor de carga de uma viga - resultados.

*** Parametros gerais:

```

Numero de iteracoes -----> 16
Valor da funcao objectivo -----> -1.16568542e+01
Erro -----> 7.63027033e-09

```

*** Variaveis de projecto:

```

1      1      11.65685424949236900000 -> Z
2      2      1.00000000000000000000 -> L1
3      3      0.41421356237309492000 -> a1
4      4      0.58578643762690508000 -> b1
5      5      1.00000000000000000000 -> E1
6      6      0.00000000000000000000 -> G1
7      7      1.00000000000000000000 -> S1
8      8      1000.000000000000000000 -> e1
9      9      0.444444444443999997000 -> B1
10     10     3.00000000000000000000 -> H1
11     11     1.00000000000000000000 -> I1
12     12     1.00000000000000000000 -> M1
13     13     -1.00000000000000000000 -> p1
14     14     0.00000000000000000000 -> d_1_1

```

15	15	0.38919067808613428000	-> d_1_2
16	16	-0.11831483839906291000	-> d_1_3
17	17	0.11304830317101416000	-> d_1_4
18	18	-0.11831483839906291000	-> d_1_5
19	19	-0.00671391707078631580	-> d_1_6
20	20	0.00000000000000000000	-> d_1_7
21	21	-0.20197606294564019000	-> d_1_8
22	22	0.00000000000000000000	-> D_1_1
23	23	0.38919067808613428000	-> D_1_2
24	24	0.00000000000000000000	-> D_2_1
25	25	0.00000000000000000000	-> D_2_2
26	26	4.82842712474619070000	-> F_1_1
27	27	0.00000000009000000000	-> F_1_2
28	28	0.00000000000000000000	-> F_1_3
29	29	-0.99999999999999989000	-> F_1_4
30	30	0.00000000000000000000	-> F_1_5
31	31	0.99999999999999989000	-> F_1_6
32	32	6.82842712474618810000	-> F_1_7
33	33	0.99999999999999989000	-> F_1_8
34	34	0.00000000000000000000	-> Q_1_1
35	35	0.00000000000000000000	-> Q_1_2
36	36	0.00000000000000000000	-> Q_2_1
37	37	0.00000000000000000000	-> Q_2_2
38	38	4.82842712474619070000	-> R_1_1
39	39	0.00000000000000000000	-> R_1_2
40	40	6.82842712474618810000	-> R_2_1
41	41	0.99999999999999989000	-> R_2_2

*** Folgas nas restricoes desigualdade:

42	1	0.99999999999999989000	-> M_1_L_min
43	2	0.99999999999999967000	-> M_1_L_max
44	3	0.00000000440054965830	-> M_1_M_min
45	4	2.00000000000000000000	-> M_1_M_max
46	5	2.00000000000000040000	-> M_1_R_min
47	6	0.0000000036199383764	-> M_1_R_max
48	7	0.40421356237309497000	-> a1_min
49	8	0.57578643762690518000	-> b1_min
50	9	1000.00000000000000000000	-> Rot_1_L_min
51	10	1000.00000000000000000000	-> Rot_1_L_max
52	11	999.88023777975832000000	-> Rot_1_M_min
53	12	1000.11976222024160000000	-> Rot_1_M_max
54	13	1000.20197606294520000000	-> Rot_1_R_min
55	14	999.79802393705438000000	-> Rot_1_R_max

*** Multiplicadores de Lagrange das restricoes desigualdade:

56	1	0.00000000108465228909	-> M_1_L_min
57	2	0.00000000025607375987	-> M_1_L_max
58	3	0.00070243560051960871	-> M_1_M_min
59	4	-0.00000000031177724583	-> M_1_M_max
60	5	-0.00000000025607375987	-> M_1_R_min
61	6	0.00129704062683631840	-> M_1_R_max
62	7	-0.00000000090894068338	-> a1_min
63	8	-0.00000000090894068338	-> b1_min
64	9	0.00000000000500000000	-> Rot_1_L_min
65	10	-0.00000000000500000000	-> Rot_1_L_max
66	11	0.0000000000050010002	-> Rot_1_M_min
67	12	-0.0000000000049990002	-> Rot_1_M_max
68	13	-0.0000000000049995000	-> Rot_1_R_min
69	14	-0.0000000000050005000	-> Rot_1_R_max

*** Multiplicadores de Lagrange das restricoes igualdade:

70	1	-0.00000000000561284618	-> Equil_a1_1
71	2	-4.82842712474619340000	-> Equil_a1_2
72	3	2.00000000000000000000	-> Equil_a1_3
73	4	-4.82842712474619070000	-> Equil_a1_4
74	5	1.99999999999999960000	-> Equil_b1_1
75	6	3.41421356237309050000	-> Equil_b1_2
76	7	-0.00000000000561283663	-> Equil_b1_3
77	8	3.41421356237309450000	-> Equil_b1_4
78	9	0.00000000060051531311	-> Equil_n1_1
79	10	4.82842712474619340000	-> Equil_n1_2
80	11	0.0000000040971350224	-> Equil_n2_1
81	12	0.0000000111111110425	-> Equil_n2_2
82	13	-0.00000003721614637824	-> Compl_1_L
83	14	34.40959192335467500000	-> Compl_1_M
84	15	-8.44881436077878330000	-> Compl_1_R
85	16	23.31370849898476600000	-> L1=a1+b1

A resolução do problema P2, definido no início desta secção, pode ser efectuada com a formulação destinada à resolução do problema P1, com a excepção de ser necessário acrescentar a seguinte restrição, em que \bar{Z} representa um valor fixo.

$$Z = \bar{Z} \quad (7.27)$$

Com esta restrição adicional o programa matemático passa a apresentar um número de restrições igualdade igual ao número de variáveis. Nestas circunstâncias continua a ser possível obter uma solução admissível com o programa NEWTOP, desde que \bar{Z} não seja maior do que o máximo factor de carga. Na generalidade dos casos, a solução do problema P2 é a única solução admissível do correspondente programa matemático.

7.3 - OPTIMIZAÇÃO DE ESTRUTURAS COM COMPORTAMENTO ELASTOPLÁSTICO

A optimização de uma estrutura com as características e restrições indicadas na secção 7.1 pode adoptar diversas formas que diferem essencialmente na selecção das variáveis que são previamente fixadas e das que podem ser modificadas pelo algoritmo de optimização. Nesta secção é considerada a seguinte formulação:

Problema P3 - minimizar o custo de uma viga contínua com diversos tramos de comprimento fixo e com comportamento elastoplástico. São consideradas como variáveis as alturas das secções e como grandezas fixas as larguras das secções, as acções e as características dos materiais.

O objectivo deste problema é o seguinte

$$\text{Min. } f(\underline{x}) = \sum_{i=1}^n (c_i L_i B_i H_i) \quad (7.28)$$

Se se pretender minimizar o peso da estrutura, a c_i deve ser atribuído o valor do peso específico do material em vez do respectivo custo por unidade de volume.

No problema P3 as seguintes grandezas são consideradas fixas, sendo pré-substituídas pelo programa NEWTOP (ver secção 5.3).

$$\left(Z, \underline{L}, \underline{E}, \underline{G}, \underline{c}, \underline{S}, \underline{e}, \underline{B}, \underline{p}, \underline{Q} \right) \quad (7.29)$$

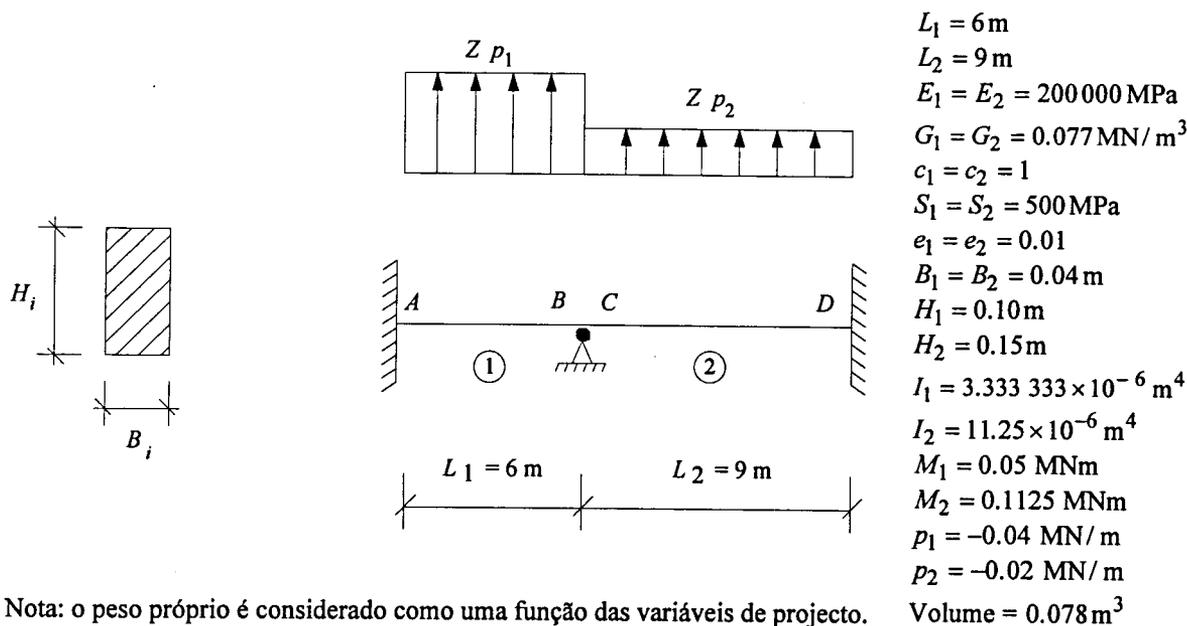
As variáveis de projecto são as seguintes

$$\left(\underline{a}, \underline{b}, \underline{H}, \underline{I}, \underline{M}, \underline{d}, \underline{D}, \underline{F}, \underline{R} \right) \quad (7.30)$$

No problema P3 existem portanto $21n + 4m$ variáveis de projecto (ver Quadro 7.1). À formulação do problema P1 (ver Quadro 7.2), devem ser acrescentadas as restrições igualdade (7.13) e (7.14). No problema P3 existem assim $14n$ restrições desigualdade e $20n + 4m$ restrições igualdade. Subtraindo o número de restrições igualdade ao número de variáveis de projecto, conclui-se que o número de variáveis independentes é igual a n . No problema P3 as variáveis independentes são as alturas das secções (\underline{H}) , porque se estas forem fixadas, o problema de optimização é transformado num problema de análise (ver problema P2). Pelos motivos indicados na secção 7.2, sugere-se que o valor inicial das variáveis dependentes não seja calculado pelo programa NEWTOP e que a solução inicial apresente rotações plásticas moderadas em todas as rótulas. A solução inicial pode eventualmente ser inadmissível e deve implicar a existência de folgas não nulas em todas as restrições desigualdade.

Com o objectivo de validar a formulação proposta e avaliar a aplicabilidade do programa NEWTOP à optimização de estruturas com comportamento não linear, é considerada a viga contínua que se encontra representada na Figura 7.7. Antes de proceder à respectiva optimização foi efectuada uma análise com os dados indicados na Figura 7.7 e com a formulação correspondente ao problema P1 (ver secção 7.2). A maximização do factor de

carga Z com o programa NEWTOP conduziu aos resultados indicados na Figura 7.8. Estes resultados foram confirmados com um programa de análise linear de estruturas [Aze92], tendo sido utilizados elementos de barra prismática.



Nota: o peso próprio é considerado como uma função das variáveis de projecto.

Figura 7.7 - Viga contínua com comportamento elastoplástico.

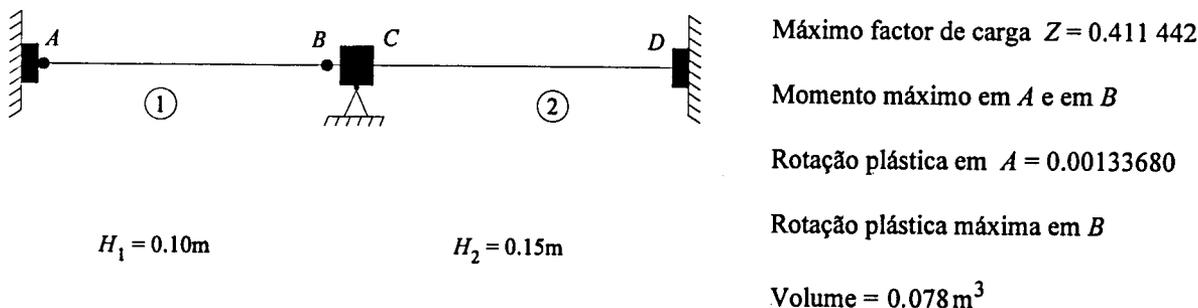


Figura 7.8 - Resultados da análise da viga contínua representada na Figura 7.7.

A viga contínua representada na Figura 7.7 foi em seguida otimizada nas condições correspondentes ao problema P3. Com o objectivo de comparar o volume da solução analisada com o volume da solução óptima, foi considerado o factor de carga indicado na

Figura 7.8. Com $Z = 0.411\ 442$ e com H_1 e H_2 como variáveis independentes foi obtida a solução óptima indicada na Figura 7.9.

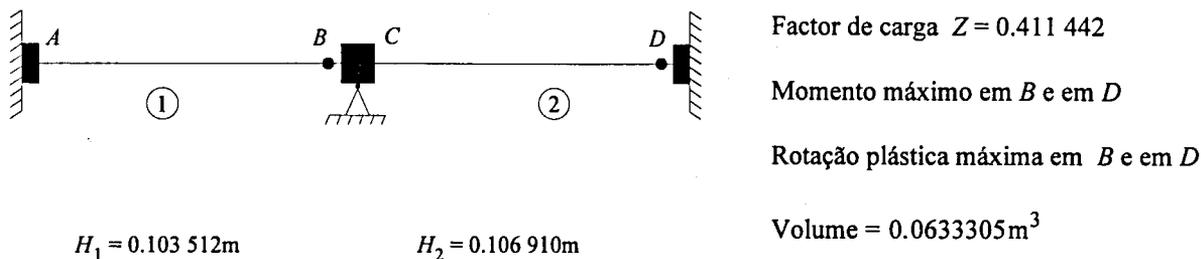


Figura 7.9 - Resultados da otimização da viga contínua representada na Figura 7.7.

O volume correspondente à solução inicial analisada (ver Figura 7.7) é cerca de 23% superior ao volume correspondente à solução óptima (ver Figura 7.9).

Os deslocamentos e os esforços correspondentes à solução óptima indicada na Figura 7.9 foram também confirmados com um programa de análise linear de estruturas [Aze92]. Na Figura 7.10 encontra-se representada a correspondente deformada e na Figura 7.11 o diagrama de momentos flectores.

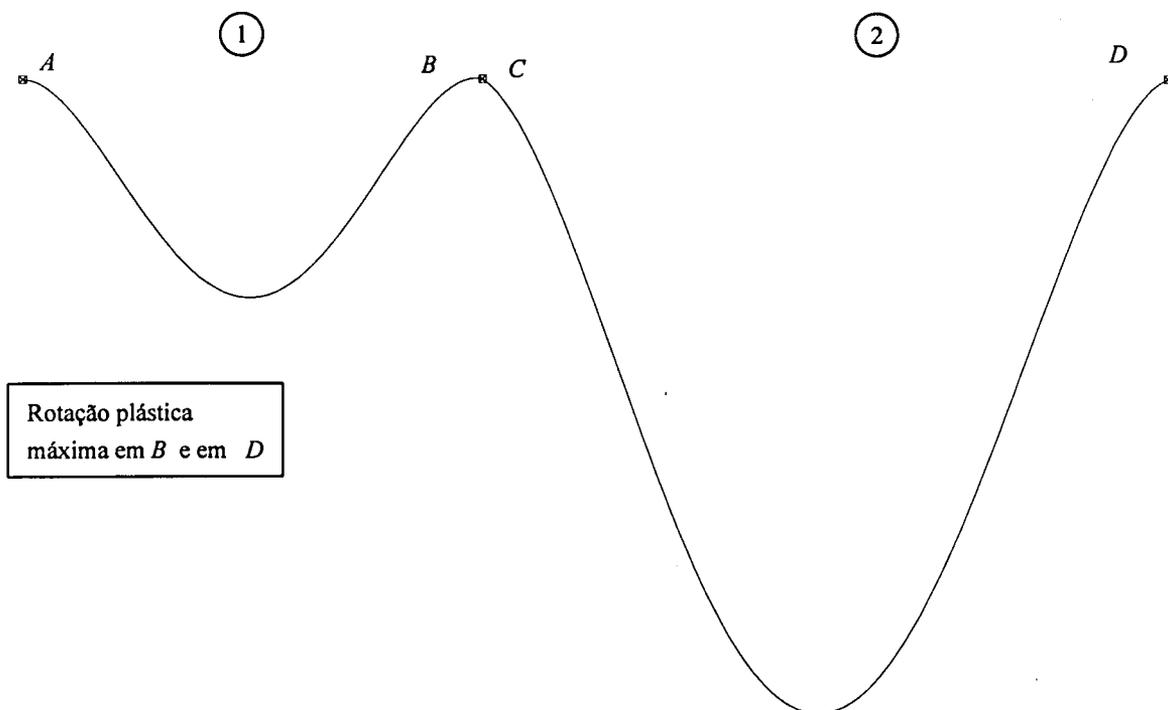


Figura 7.10 - Deformada correspondente à solução óptima indicada na Figura 7.9.

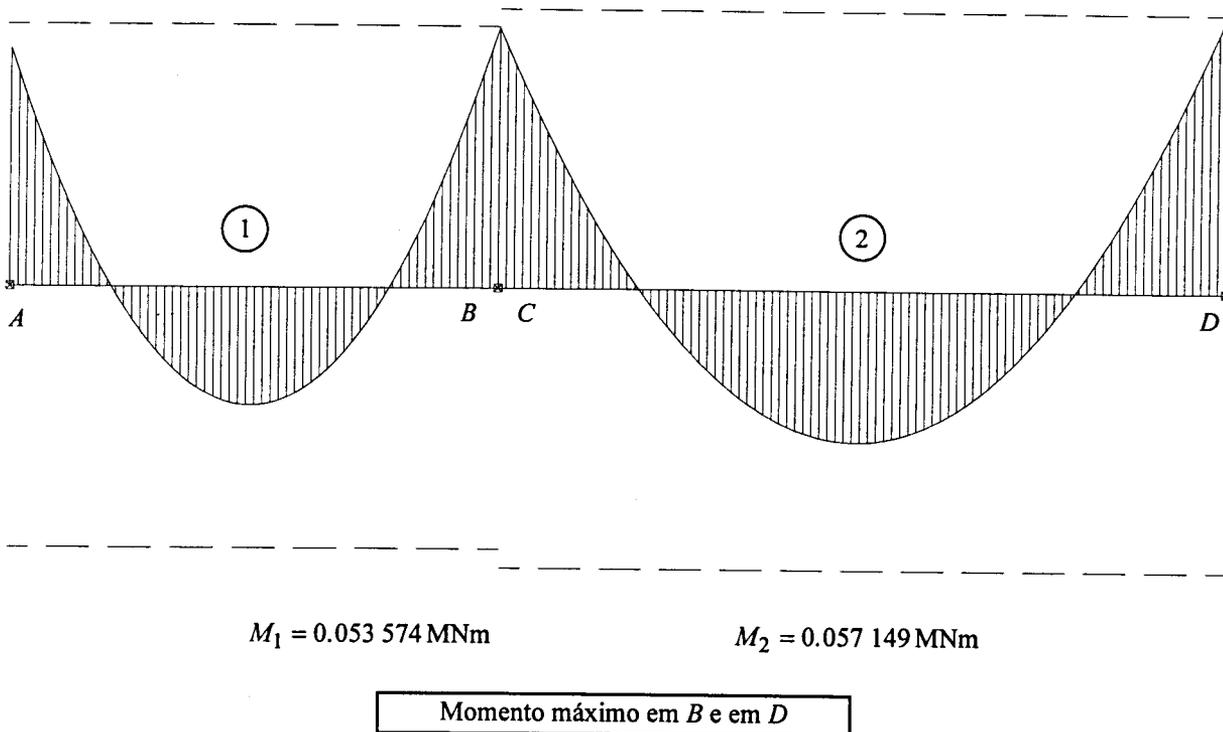


Figura 7.11 - Diagrama de momentos flectores correspondente à solução óptima indicada na Figura 7.9.

7.4 - OBSERVAÇÕES FINAIS

A formulação proposta neste capítulo revela-se já suficientemente versátil para ser aplicada a diversos tipos de problemas de optimização. O facto de praticamente todas as grandezas associadas à descrição e ao comportamento da estrutura estarem presentes como variáveis na formulação torna muito simples a abordagem de outros tipos de problemas (e.g., optimização da posição dos apoios, optimização de materiais). Verifica-se também que todas as considerações atrás referidas são susceptíveis de serem generalizadas a outros tipos de estruturas reticuladas (e.g., pórticos). Os resultados da análise efectuada com base na programação matemática, bem como os correspondentes à solução óptima devem ser sempre verificados recorrendo a uma análise não linear incremental [Aze85]. Esta necessidade é devida ao facto de a formulação proposta apenas contemplar um nível de carregamento, ignorando assim o efeito da história da aplicação das acções e a existência de eventuais descargas locais [Iro86]. Admite-se no entanto que na generalidade dos casos uma solução óptima obtida com a hipótese de existência de um único nível de carregamento é suficientemente rigorosa.

CAPÍTULO 8

CONCLUSÕES

8.1 - OBSERVAÇÕES FINAIS

A investigação no domínio da optimização de estruturas tem tido como principal objectivo a resolução de problemas com comportamento linear com um número moderado de variáveis de projecto independentes e com um número de graus de liberdade relativamente elevado, de modo a tornar possível a abordagem de casos com interesse prático. Para se conseguir esse objectivo, é em geral necessário considerar as variáveis que caracterizam o comportamento da estrutura de um modo implícito (análise de sensibilidades) e, devido às exigências proibitivas associadas ao cálculo de segundas derivadas, é habitual recorrer a métodos que apenas necessitam de primeiras derivadas. Com esta formulação têm sido resolvidos com sucesso alguns problemas de considerável complexidade [Fle86] [Esp87], continuando no entanto a existir uma grande dificuldade na abordagem de problemas que apresentem um número de variáveis de projecto independentes superior a algumas centenas [Van93] [Ram94]. A optimização de estruturas com comportamento não linear recorrendo a formulações baseadas na análise de sensibilidades só recentemente foi iniciada [Aro87a], permanecendo limitada à resolução de problemas de reduzida complexidade.

Como alternativa à abordagem da optimização de estruturas por intermédio da programação matemática e análise de sensibilidades, alguns autores propõem métodos dependentes do significado físico do problema e que se baseiam na verificação de um critério de optimalidade [Ber87]. Com estes métodos é possível resolver problemas de optimização com muitas centenas de variáveis, mas o respectivo campo de aplicação permanece muito limitado [Ram94].

O principal objectivo do presente trabalho consistiu na abordagem da optimização de estruturas de um modo mais genérico e mais versátil, mantendo uma eficiência aceitável, que permita uma resolução em tempo útil de problemas com considerável complexidade. Uma vez

que a robustez de um método de optimização é uma característica decisiva do ponto de vista de garantia da qualidade da solução, foi desenvolvido um método baseado em aproximações de segunda ordem. Pelo facto de serem utilizadas aproximações de grau mais elevado do que é habitual, torna-se possível na generalidade dos casos a obtenção com maior fiabilidade de soluções rigorosas. Apesar do aumento de complexidade que resulta da necessidade de calcular as primeiras e segundas derivadas de todas as funções presentes no programa matemático, foi possível resolver problemas com um número de variáveis de projecto independentes da ordem dos milhares e com um número de restrições da ordem da dezena de milhar. Da comparação com algumas soluções obtidas com outros métodos e por outros autores, foi possível concluir que os resultados publicados neste trabalho apresentam uma elevada precisão e que o método proposto se revela menos susceptível de convergir para mínimos locais.

Tendo como principal objectivo a possibilidade de formular os problemas de optimização de estruturas com uma elevada versatilidade, foi necessário considerar as variáveis associadas ao comportamento da estrutura como variáveis de projecto. Esta formulação apresenta a vantagem de ser evitada a análise de sensibilidades, mas conduz a um aumento significativo do número de variáveis, principalmente quando as estruturas são solicitadas por múltiplos casos de carga. Este inconveniente tem uma progressiva tendência para se tornar secundário, devido ao constante aumento da *performance* e da capacidade dos meios de cálculo automático. A consideração das variáveis associadas ao comportamento da estrutura como variáveis de projecto é fundamental quando o respectivo comportamento é não linear. Deste modo é evitado o recurso a uma análise de sensibilidades mais complexa, bem como a necessidade de efectuar sucessivas reanálises da estrutura recorrendo a métodos incrementais.

Com base na situação actual de desenvolvimento do método proposto neste trabalho, é possível concluir que a optimização de estruturas reticuladas (articuladas ou contínuas) e eventualmente com um comportamento não linear pode ser efectuada de um modo robusto e eficiente. A sua extensão a outros tipos de problemas é facilitada pelo facto de a preparação dos correspondentes programas matemáticos apresentar uma grande versatilidade, que resulta da possibilidade de interpretação automática pelo próprio código computacional.

O método de optimização desenvolvido no âmbito do presente trabalho apresenta uma significativa dependência da qualidade da solução inicial. Se o utilizador não pré-dimensionar a estrutura e não atender ao significado físico das variáveis do problema, o sucesso da aplicação do método de optimização pode ser comprometido. Se o código computacional for utilizado nas condições sugeridas, qualquer projectista de estruturas será certamente capaz de

utilizar as técnicas de optimização propostas com o objectivo de conseguir soluções mais económicas e igualmente seguras.

8.2 - SUGESTÕES PARA FUTUROS DESENVOLVIMENTOS

O método desenvolvido no âmbito do presente trabalho pode ser utilizado na resolução de um elevado número de classes de problemas de optimização. Sempre que é desenvolvida a formulação de um novo tipo de problema, deve ser também preparado um programa de computador destinado à geração das funções que constituem o programa matemático. Estes programas de computador devem apresentar uma elevada versatilidade para que qualquer restrição com interesse prático possa ser incluída no programa matemático. Tendo em vista um futuro desenvolvimento da optimização de estruturas com base na utilização do programa NEWTOP como caixa negra, sugere-se a preparação de programas de computador destinados à geração de programas matemáticos nos seguintes domínios:

- optimização da forma de estruturas reticuladas;
- optimização de estruturas pré-esforçadas;
- optimização de estruturas com restrições relativas a valores próprios (frequências próprias ou instabilidade);
- optimização de estruturas sujeitas a grandes deformações;
- optimização de fundações e estruturas de suporte de terras;
- optimização de espessuras em meios contínuos;
- optimização da forma em meios contínuos;
- optimização de materiais.

Esta lista não é exaustiva e pode ser ampliada com problemas que correspondam a combinações das características atrás indicadas. Algumas das formulações referidas foram já testadas com sucesso em casos simples.

O programa NEWTOP apresenta-se já numa fase avançada de desenvolvimento, devendo no entanto ser melhorados os seguintes aspectos:

- aumento da eficiência e da economia de memória durante a resolução do sistema de equações $\tilde{H} \Delta \tilde{X} + \nabla L = 0$, tendo em consideração a elevada esparsidade da matriz Hessiana;
- aumento da eficiência e da robustez da fase de pesquisa unidimensional;
- aumento da versatilidade do módulo de interpretação e derivação de funções, tendo em vista a manipulação de outros tipos de expressões.

Devido ao facto de o programa NEWTOP apresentar já um elevado número de módulos, afigura-se como fundamental que, antes de se proceder ao aumento da sua funcionalidade, a escrita do código computacional passe a ser efectuada tirando partido de técnicas de programação orientada para objectos.

REFERÊNCIAS BIBLIOGRÁFICAS

- [Aba69] Abadie, J.; Carpentier, J. - Generalization of the Wolfe Reduced Gradient Method to the Case of Nonlinear Constraints, Optimization, R. Fletcher (ed.), Academic Press, New York, pp. 37-47, 1969.
- [Ada80] Adão da Fonseca, A. M. - Plastic Limit Analysis and Synthesis of Plates and Shells by Mathematical Programming, Tese de Doutorado (PhD), Universidade de Londres, 1980.
- [Aro74] Arora, J. S. - On Improving Efficiency of an Algorithm for Structural Optimization and a User's Manual for Program Trussopt 3, Tech. Rep. N° 12, Materials Engineering Division, The University of Iowa, Iowa City, June 1974.
- [Aro87a] Arora, J. S.; Wu, C. C. - Design Sensitivity Analysis and Optimization of Nonlinear Structures- Computer Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer - Verlag, 1987.
- [Aro87b] Arora, J. S. - Issues of Generality, Reliability and Efficiency in Optimum Design (panel discussion) - Computer Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer-Verlag, 1987.
- [Aro89] Arora, J. S. - Introduction to Optimum Design, McGraw-Hill, 1989.
- [Aro91] Arora, J. S.; Chahande, A. I.; Paeng, J. K. - Multiplier Methods for Engineering Optimization, International Journal for Numerical Methods in Engineering, Vol. 32, pp. 1485-1525, 1991.
- [Aze84] Azeredo, M. - Análise de Estruturas, Novos Conceitos e Bases para a sua Aplicação, Dissertação para Doutorado, Faculdade de Engenharia da Universidade do Porto, 1984.
- [Aze85] Azevedo, A. F. M. - Análise Não Linear de Estruturas Planas de Betão Armado pelo Método dos Elementos Finitos - Tese de Mestrado, Faculdade de Engenharia da Universidade do Porto, 1985.

- [Aze89] Azevedo, A. F. M.; Adão da Fonseca, A. M. - Optimização de Estruturas Reticuladas por um Método de Segunda Ordem, X Congresso Ibero-Latino-Americano sobre Métodos Computacionais em Engenharia, Porto, 1989.
- [Aze90] Azevedo, A. F. M.; Barros, J. A. O. - Análise Comparativa de Métodos Directos e Iterativos na Resolução de Grandes Sistemas de Equações Lineares, 2^{as} Jornadas Portuguesas de Engenharia de Estruturas, Laboratório Nacional de Engenharia Civil, Lisboa, Novembro, 1990.
- [Aze92] Azevedo, A. F. M.; Barros, J. A. O. - Manual de Utilização do Programa FEMIX-versão 2.1, Faculdade de Engenharia da Universidade do Porto, 1992.
- [Bar89] Barthelemy, B.; Haftka, R. T.; Madapur, U.; Sankaranarayanan, S. - Integrated Structural Analysis and Design using 3D Finite Elements, 30th Structures, Structural Dynamics and Materials Conference, Mobile, Alabama, pp.1305-1310, 1989.
- [Bec87] Beckers, P. - CAD Technology in Optimal Design, Computer Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer-Verlag, 1987.
- [Bec91] Beckers, P. - Recent Developments in Shape Sensitivity Analysis: the Physical Approach, Engineering Optimization, Vol.18, pp. 67-78, 1991.
- [Bel82] Belegundu, A. D. - A Study of Mathematical Programming Methods for Structural Optimization, PhD thesis, University of Iowa, 1982.
- [Bel85] Belegundu, A. D.; Arora J. S. - A Study of Mathematical Programming Methods for Structural Optimization: Part I; Part II, International Journal for Numerical Methods in Engineering, Vol. 21, pp. 1583-1623, 1985.
- [Ber87] Berke, L.; Khot, N. - Structural Optimization Using Optimality Criteria - Computer Aided Optimal Design: Structural and Mechanical Systems, Edited by C. A. Mota Soares, Springer-Verlag, 1987.
- [Bla93] Blaauwendraad, J.; Schneider, J. - Structural Analysis: Elastic or Plastic? - Structural Engineering International, Vol. 3, Nº 3, pp. 165-168, August 1993.
- [Ble92] Bletzinger, K. U.; Ramm, E. - Structural Optimization as Tool for Shape Design, Numerical Methods in Engineering 92, Ch. Hirsch *et al.* (eds.), Elsevier Science Publishers B. V., 1992.

[Boc92] Bochenek, B.; Rasmussen, J. - On the Implementation of Equality Constraints into the Method of Moving Asymptotes, Numerical Methods in Engineering 92, Ch. Hirsch *et al.* (eds.), Elsevier Science Publishers B.V., pp. 811-818, 1992.

[Box69] Box, M. J.; Davies, D.; Swann, W. H. - Nonlinear Optimization Techniques, ICI Monograph N° 5, Oliver & Boyd, 1969.

[Bur93] Burden, R. L.; Faires, J. D. - Numerical Analysis, Fifth Edition, PWS Publishing Company, Boston, 1993.

[Dan63] Dantzig, G. B. - Linear Programming and Extensions, Princeton University Press, Princeton, N. J., 1963.

[Esp87] Esping, B. J. D.; Holm, D. - A CAD Approach to Structural Optimization - Computer-Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer-Verlag, 1987.

[Fia68] Fiacco, A. V.; McCormick, G. P. - Nonlinear Programming: Sequential Unconstrained Minimization Techniques, John Wiley and Sons, New York, 1968.

[Fle64] Fletcher, R.; Reeves, C. M. - Function Minimization by Conjugate Gradients, Br. Computer J., Vol. 7, N° 2, pp. 149-154, 1964.

[Fle82] Fletcher, R. - Methods for Nonlinear Constraints, Nonlinear Optimization 1981, NATO Conference Series, M. J. D. Powell (ed.), Academic Press, pp. 185-211, 1982.

[Fle86] Fleury, C.; Braibant, V. - Structural Optimization: A New Dual Method Using Mixed Variables, International Journal for Numerical Methods in Engineering, Vol. 23, pp. 409-428, 1986.

[Fle87] Fleury, C. - Computer Aided Optimal Design of Elastic Structures- Computer Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer-Verlag, 1987.

[Fle88] Fleury, C. - A Convex Linearization Method Using Second Order Information, FEMCAD 88 (proceedings), IITT International, Paris, 1988.

[Fle89] Fleury, C. - First and Second Order Convex Approximation Strategies in Structural Optimization, Structural Optimization, 1, pp. 3-10, Springer-Verlag, 1989.

- [Fri70] Fried, I. - A Gradient Computational Procedure for the Solution of Large Problems Arising from the Finite Element Discretization Method, *International Journal for Numerical Methods in Engineering*, Vol. 2, pp. 477-494, 1970.
- [Gal93] Galante, M.; Cerrolaza, M. - Un Algoritmo Genetico para Optimizar Estructuras Esbeltas Reales, *II Congresso de Metodos Numericos en Ingenieria*, F. Navarrina y M. Casteleiro (eds.), SEMNI, 1993.
- [Gil81] Gill, P. E.; Murray, W.; Wright, M. H. - *Practical Optimization*, Academic Press, 1981.
- [Gil91] Gilbert, J. C. - Nonlinear Optimization and Large Scale Problems, *Engineering Optimization*, Vol. 18, pp. 5-21, 1991.
- [Haf85] Haftka, R. T. - Simultaneous Analysis and Design, *AIAA Journal*, Vol.23, N° 7, pp.1099-1103, 1985.
- [Haf87] Haftka, R. T.; Kamat, M. P. - *Finite Elements in Optimal Structural Design - Computer Aided Optimal Design: Structural and Mechanical Systems*, C. A. Mota Soares (ed.), Springer-Verlag, 1987.
- [Haj92] Hajela, P. - Genetic Algorithms in Automated Structural Synthesis - Optimization and Artificial Intelligence in Civil and Structural Engineering, B. H. V. Topping (ed.), Volume I, Kluwer Academic Publishers, 1992.
- [Hau79] Haug, E. J.; Arora, J. S. - *Applied Optimal Design: Mechanical and Structural Systems*, Wiley - Interscience, New York, 1979.
- [Her93] Hernandez, S. - From CAD to CHAOSD: Structural Design and Optimization Techniques - Advanced Techniques in the Optimum Design of Structures, S. Hernandez (ed.), Computational Mechanics Publications, 1993.
- [Hes52] Hestenes, M. R.; Stiefel, E. L. - Methods of Conjugate Gradients for Solving Linear Systems, *J. Res. Nat. Bur. Standards, Section B*, 49, pp. 409-436, 1952.
- [Hes69] Hestenes, M. R. - Multiplier and Gradient Methods, *J. Optimization Theory Applic.*, 4, pp. 303-320, 1969.
- [Hil90] Hillier, F. S.; Lieberman, G. J. - *Introduction to Operations Research*, Fifth Edition, McGraw-Hill, 1990.

- [Hin91] Hinton, E.; Rao, N. V. R.; Ozakça, M. - An Integrated Approach to Structural Shape Optimization of Linearly Elastic Structures, Part I: General Methodology, Computing Systems in Engineering, Vol. 2, N° 1, 1991.
- [Iro86] Irons, B.; Ahmad, S. - Techniques of Finite Elements - Ellis Horwood Limited, 1986.
- [Kao94] Kao, C.; Chen, S. P. - A Sequential Quadratic Programming Algorithm Utilizing QR Matrix Factorization, Engineering Optimization, Vol. 22, pp. 283-296, 1994.
- [Kar39] Karush, W. - Minima of Functions of Several Variables with Inequalities as Side Conditions, M. S. Thesis, Department of Mathematics, University of Chicago, 1939.
- [Kar93] Karihaloo, B. L. - Most Efficient NLP Techniques in Optimum Structural Frame Design, Eng. Opt., Vol. 20, pp. 261-272, 1993.
- [Kel60] Kelley, J. E. - The Cutting Plane Method for Solving Convex Programs, J. SIAM, Vol. 8, pp. 702-712, 1960.
- [Ker88] Kernighan, B. W.; Ritchie, D. M. - The C Programming Language - ANSI C, Second Edition, Prentice Hall, 1988.
- [Kir81] Kirsch, U. - Optimum Structural Design, Concepts Methods and Applications, McGraw-Hill, 1981.
- [Kod93] Kodiyalam, S. - Sensitivity Analysis in Static and Dynamic Problems - Advanced Techniques in the Optimum Design of Structures, S. Hernandez (ed.), Computational Mechanics Publications, 1993.
- [Kuh51] Kuhn, H. W.; Tucker, A. W. - Nonlinear Programming, Jerzy Neyman (ed.), Proceedings of the Second Berkeley Symposium, University of California Press, Berkeley, pp. 481-492, 1951.
- [Loo82] Lootsma, F. A. - Performance Evaluation of Nonlinear Optimization Methods via Multi-Criteria Decision Analysis and via Linear Model Analysis, Nonlinear Optimization 1981, NATO Conference Series, M. J. D. Powell (ed.), Academic Press, pp. 419-453, 1982.
- [Lue69] Luenberger, D. G. - Hyperbolic Pairs in the Method of Conjugate Gradients, SIAM J. Appl. Math., Vol. 17, N° 6, November 1969.
- [Lue84] Luenberger, D. G. - Linear and Nonlinear Programming, 2nd Edition, Addison-Wesley, 1984.

- [Maj72] Majid, K. I. - Non-Linear Structures - Butterworth & Co., London, 1972.
- [McC67] McCormick, G. P. - Second Order Conditions for Constrained Optima, SIAM J. Applied Mathematics, Vol. 15, pp. 641-652, 1967.
- [Mot88] Mota Soares, C. M.; Barbosa, J. I.; Mota Soares, C. A.; Pinto, P. - Optimal Design of Axisymmetric Shell Structures, FEMCAD 88 (proceedings), IITT International, Paris, 1988.
- [NAG91] NAG Fortran Library Concise Reference, Mark 15, The Numerical Algorithms Group Limited, Oxford, United Kingdom, 1991.
- [Nea77] Neal, B. G. - The Plastic Methods of Structural Analysis - Third (S. I.) Edition, Chapman and Hall, London, 1977.
- [Nye90] Nye, A. - Xlib Programming Manual, Volume One, O'Reilly & Associates, 1990.
- [Pow64] Powell, M. J. D. - An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives, Computer J., Vol. 7, Nº 4, pp. 303-307, 1964.
- [Pow69] Powell, M. J. D. - A Method for Nonlinear Constraints in Minimization Problems, R. Fletcher (ed.), Optimization, Academic Press, New York, 1969.
- [Pow77] Powell, M. J. D. - A Fast Algorithm for Nonlinearly Constrained Optimization Calculations, DAM PTP77/NA2, University of Cambridge, England, 1977.
- [Pow81] Powell, M. J. D. - Optimization Algorithms in 1979, Comittee on Algorithms Newsletter, Nº 5, Mathematical Programming Society, pp. 2-16, February 1981.
- [Pre92] Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. - Numerical Recipes in C - The Art of Scientific Computing, Second Edition, Cambridge University Press, 1992.
- [Ram94] Ramm, E.; Maute, K. - Topology Optimization - A General Tool in Structural Design, Euro C - Computational Modelling of Concrete Structures (Proceedings), Innsbruck, Pineridge Press, 1994.
- [Ros61] Rosen, J. B. - The Gradient Projection Method for Nonlinear Programming, Part II, Nonlinear Constraints, J. Soc. Ind. Appl. Math., 9, pp. 514-532, 1961.
- [San89] Santos, R. M. P.; Azevedo, A. F. M. - Visualização de Dados e Resultados da Aplicação do Método dos Elementos Finitos a Estruturas Tridimensionais, X Congresso Ibero-Latino-Americano sobre Métodos Computacionais em Engenharia, Porto, 1989.

- [Sch60] Schmit, L. A. - Structural Design by Systematic Synthesis, Proceedings of the Second ASCE Conference on Electronic Computations, Pittsburgh, pp. 105-122, 1960.
- [Ser88] Serra Neves, A. A. - Análise de Placas e Cascas em Regime Elastoplástico Aplicando a Programação Matemática - Dissertação para Doutoramento, Faculdade de Engenharia da Universidade do Porto, 1988.
- [Sva85] Svanberg, K. - Method of Moving Asymptotes - A New Method for Structural Optimization, TRITA-MAT, 1985.
- [Sva87] Svanberg, K. - The Method of Moving Asymptotes - A New Method for Structural Optimization, International Journal for Numerical Methods in Engineering, Vol. 24, pp. 359-373, 1987.
- [Sve68] Sved, G.; Ginos, Z. - Structural Optimization under Multiple Loading, Int. J. Mech. Sci., 10, pp. 803-805, 1968.
- [Tin93] Tin-Loi, F. - Plastic Limit Analysis, Mathematical Programming and GAMS - Eng. Opt., Vol. 20, pp. 273-286, 1993.
- [Top84] Topping, B. H. V.; Robinson, D. J. - Selecting Nonlinear Optimization Techniques for Structural Design, Eng. Comput., Vol. 1, pp. 252-262, September 1984.
- [Val86] Valadares Tavares, L.; Nunes Correia, F. - Optimização Linear e Não Linear - Conceitos, Métodos e Algoritmos, Fundação Calouste Gulbenkian, 1986.
- [Van84] Vanderplaats, G. N. - Numerical Optimization Techniques for Engineering Design: with Applications, McGraw-Hill, 1984.
- [Van87] Vanderplaats, G. N. - Numerical Optimization Techniques - Computer Aided Optimal Design: Structural and Mechanical Systems, C. A. Mota Soares (ed.), Springer-Verlag, 1987.
- [Van89a] Vanderplaats, G. N.; Hansen, S. R. - DOT Users Manual, VMA Engineering, Goleta, CA, 1989.
- [Van89b] Vanderplaats, G. N.; Miura, H.; Nagendra, G.; Wallerstein, D. - Optimization of Large Scale Structures using MSC/NASTRAN, Computer Aided Optimum Design of Structures: Applications, C. A. Brebbia and S. Hernandez (eds.), Computational Mechanics Publications, pp. 51-68, 1989.

- [Van93] Vanderplaats, G. N. - Development of Efficient Structural Design Software - Advanced Techniques in the Optimum Design of Structures, S. Hernandez (ed.), Computational Mechanics Publications, 1993.
- [Wol63] Wolfe, P. - Methods of Nonlinear Programming, Recent Advances in Mathematical Programming, R. L. Graves and P. Wolfe (eds.), McGraw-Hill, New York, pp. 76-77, 1963.
- [WuA87] Wu, C. C.; Arora, J. S. - Simultaneous Analysis and Design Optimization of Nonlinear Response - Engineering with Computers, Vol. 2, pp. 53-63, Springer-Verlag, 1987.
- [Zou60] Zoutendijk, G. - Methods of Feasible Directions, Elsevier, Amsterdam, 1960.



FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

BIBLIOTECA



000005804

