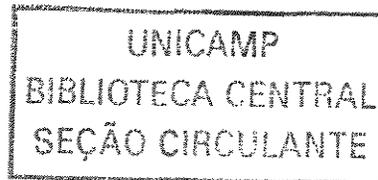


**Universidade Estadual de Campinas  
Faculdade de Engenharia Civil**

# **OTIMIZAÇÃO DE TRELIÇAS PLANAS**

***Carlos Frederico Macêdo Cortês***



**Campinas – 2002**

Universidade Estadual de Campinas  
Faculdade de Engenharia Civil

# Otimização de Treliças Planas

Carlos Frederico Macêdo Cortês

Orientador: Francisco Antonio Menezes

Co-orientador: Renato Soliani

Dissertação de Mestrado apresentada à Comissão de Pós-graduação da Faculdade Engenharia Civil da Universidade Estadual de Campinas, como parte dos requisitos para obtenção do título de Mestre em Engenharia Civil, na área de concentração de Estruturas.

Atesto que esta é a versão definitiva da dissertação/tese.

3, 9, 2

Prof. Dr. *Francisco Antonio Menezes*  
Matrícula: 03942-0

Campinas/SP – 2002

UNIDADE Be  
Nº CHAMADA T/UNICAMP  
C818o  
V \_\_\_\_\_ EX \_\_\_\_\_  
TOMBO BCI 50963  
PROC 16-837/02  
C \_\_\_\_\_ DX \_\_\_\_\_  
PREÇO R\$ 11,00  
DATA 26/09/02  
Nº CPD \_\_\_\_\_

CM00173390-5

318 ID 259095

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

C818o Cortês, Carlos Frederico Macêdo  
Otimização de treliças planas / Carlos Frederico  
Macêdo Cortês.--Campinas, SP: [s.n.], 2002.

Orientadores: Francisco Antonio Menezes e Renato  
Soliani.

Dissertação (mestrado) - Universidade Estadual de  
Campinas, Faculdade de Engenharia Civil.

1. Otimização estrutural. 2. Treliças (Construção  
civil). 3. Otimização matemática. 4. Programação  
(Matemática). 5. Programação não-linear. 6.  
Programação inteira. I. Menezes, Francisco Antonio. II.  
Soliani, Renato. III. Universidade Estadual de Campinas.  
Faculdade de Engenharia Civil. IV. Título.

Matemática  
Prof. Dr.  
Mestrado

**Universidade Estadual de Campinas  
Faculdade de Engenharia Civil**

# **Otimização de Treliças Planas**

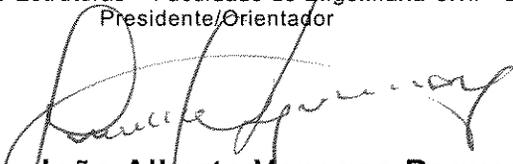
***Carlos Frederico Macêdo Cortês***

Dissertação de Mestrado qualificada pela Banca Examinadora constituída por:



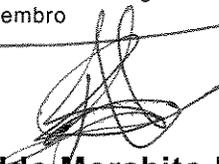
**Prof. Dr. Francisco Antonio Menezes**

Departamento de Estruturas – Faculdade de Engenharia Civil - UNICAMP  
Presidente/Orientador



**Prof. Dr. João Alberto Venegas Requena**

Departamento de Estruturas – Faculdade de Engenharia Civil - UNICAMP  
Membro



**Prof. Dr. Reinaldo Morabito Neto**

Departamento de Engenharia de Produção – UFScar  
Membro

Campinas, 2002

## Dedicatória

---

A Wânia, minha esposa, que foi o principal incentivo na conquista deste objetivo tão difícil... Somente Deus conhece o amor que tenho por ti...

## **Agradecimentos**

Aos professores da FEC/UNICAMP, que direta ou indiretamente contribuíram para o meu aprendizado. Em especial, a Francisco Antonio Menezes, Renato Soliani, respectivamente meu orientador e co-orientador, e a João Alberto Venegas Requena, que deram importante contribuição a esse trabalho.

A Elias Nicolas, Everaldo Bonaldo, Jeferson Cassiano, Leandro Vanalli, Newton dos Santos, Rafael de Souza, Rodrigo Paccola e Sérgio Soares, pela amizade e companheirismo.

A Wagner Paiva de Argolo, amigo da Graduação, com quem troquei informações e material teórico imprescindíveis ao desenvolvimento de nossas teses.

A Cícero, Marinilza e Elisângela Santos, que me acolheram com muito carinho nos momentos em que mais precisei.

Em especial, a meu pai, *José Napion Cortês*, pelo apoio, amor e incentivo, que mesmo à distância, nunca me faltou.

E a *Renan e Nenza Rocha*, meus sogros, pelo carinho, paciência e compreensão na conclusão desta jornada.

A DEUS, que tudo concretiza.

“... deixar de conseguir não tem importância, o lamentável é desistir de tentá-lo.”

(Carlos Drummond de Andrade –  
1902-1987)

# Índice

---

|  |      |
|--|------|
| Índice de Figuras                          | xv   |
| Índice de Tabelas                          | xvii |
| Resumo                                     | xix  |
| Abstract                                   | xxi  |
| <b>1. Introdução</b>                       |      |
| 1.1. Considerações Iniciais                | 1    |
| 1.2. Objetivos                             | 4    |
| 1.2.1. Por que Utilizar o GAMS?            | 5    |
| 1.3. Organização                           | 6    |
| <b>2. Otimização</b>                       |      |
| 2.1. Definições e Conceitos                | 9    |
| 2.2. Subproblemas em Otimização Estrutural | 11   |
| 2.3. Histórico                             | 12   |
| 2.4. Formulação do Problema de Otimização  | 22   |
| 2.4.1. Outras Restrições                   | 26   |

|   |    |
|---|----|
| <b>3. Programação Matemática – Técnicas de Otimização</b> |    |
| 3.1. Introdução   | 31 |
| 3.2. Programação Linear                                   | 32 |
| 3.3. Programação Não-linear                               | 35 |
| 3.3.1. Problemas de Projeto Ótimo sem Restrições          | 37 |
| 3.3.1.1. Método da Máxima Descida (Steepest Descent)      | 38 |
| 3.3.1.2. Método do Gradiente Conjugado                    | 39 |
| 3.3.1.3. Método de Newton                                 | 42 |
| 3.3.1.4. Método Quasi-Newton                              | 44 |
| 3.3.2. Problemas de Projeto Ótimo com Restrições          | 45 |
| 3.3.2.1. Método das Penalidades                           | 47 |
| 3.3.2.2. Método do Lagrangeano Aumentado                  | 49 |
| <br>  |    |
| <b>4. Otimizadores do GAMS</b>                            |    |
| 4.1. <i>MINOS</i>   | 56 |
| 4.1.1. Descrição Geral do Otimizador <i>MINOS</i>         | 56 |
| 4.1.1.1. Programação Linear                               | 58 |
| 4.1.1.2. Problemas com a Função Objetivo Não-Linear       | 59 |
| 4.1.1.3. Problemas com Restrições Não-Lineares            | 59 |
| 4.2. <i>DICOPT</i>  | 60 |
| 4.1.1. Descrição Geral do Otimizador <i>DICOPT</i>        | 62 |
| 4.3. Limitações   | 66 |

|   |            |
|---|------------|
| <b>5. Descrição do Programa e Modelos Gerados</b>   |            |
| <b>5.1. Programa pTRUSS</b>                         | <b>71</b>  |
| <b>5.2. Processamento e Impressão da Saída .GMS</b> | <b>74</b>  |
| <b>6. Exemplos e Discussão dos Resultados</b>       | <b>79</b>  |
| <b>6.1. Treliça de 6 Nós e 10 Barras</b>            | <b>81</b>  |
| <b>6.1.1. Resultados da Análise Contínua</b>        | <b>82</b>  |
| <b>6.1.2. Resultados da Análise Discreta</b>        | <b>83</b>  |
| <b>6.2. Treliça de 11 Nós e 18 Barras</b>           | <b>85</b>  |
| <b>6.2.1. Resultados da Análise Contínua</b>        | <b>86</b>  |
| <b>6.2.2. Resultados da Análise Discreta</b>        | <b>87</b>  |
| <b>6.3. Treliça de 11 Nós e 23 Barras</b>           | <b>89</b>  |
| <b>6.3.1. Resultados da Análise Contínua</b>        | <b>89</b>  |
| <b>6.3.2. Resultados da Análise Discreta</b>        | <b>90</b>  |
| <b>6.4. Torre Plana de 22 Nós e 47 Barras</b>       | <b>92</b>  |
| <b>6.4.1. Resultados da Análise</b>                 | <b>93</b>  |
| <b>6.5. Treliça de 22 Nós e 47 Barras</b>           | <b>97</b>  |
| <b>6.5.1. Resultados da Análise</b>                 | <b>97</b>  |
| <b>7. Conclusões</b>                                | <b>101</b> |
| <b>8. Referências Bibliográficas</b>                | <b>107</b> |

## **Anexos**

|  |            |
|--|------------|
| <b>AN-1. Tabelas de Perfis Disponíveis</b>         | <b>115</b> |
| <b>AN-2. GAMS – Uma Noção</b>                      |            |
| <b>AN-2.1. Introdução</b>                          | <b>121</b> |
| <b>AN-2.2. Tutorial GAMS</b>                       | <b>122</b> |
| <b>AN-2.2.1. Estrutura</b>                         | <b>123</b> |
| <b>AN-2.2.2. Conjuntos</b>                         | <b>125</b> |
| <b>AN-2.2.3. Entrada de Dados</b>                  | <b>125</b> |
| <b>AN-2.2.3.1. Lista</b>                           | <b>126</b> |
| <b>AN-2.2.3.2. Tabelas</b>                         | <b>127</b> |
| <b>AN-2.2.3.3. Atribuição Direta</b>               | <b>127</b> |
| <b>AN-2.2.4. Variáveis</b>                         | <b>128</b> |
| <b>AN-2.2.5. Equações</b>                          | <b>129</b> |
| <b>Função Objetivo</b>                             | <b>131</b> |
| <b>AN-2.2.6. Conjunto de Equações para Solução</b> | <b>131</b> |
| <b>AN-2.2.7. Saída e Exibição de Resultados</b>    | <b>133</b> |
| <b>AN-3. Norma AISC-ASD/1989</b>                   |            |
| <b>AN-3.1. Classificação das Seções</b>            | <b>135</b> |
| <b>AN-3.2. Cálculo das Tensões</b>                 | <b>140</b> |
| <b>AN-3.3. Cálculo das Tensões Admissíveis</b>     | <b>141</b> |
| <b>AN-3.3.1. Tensão Admissível de Tração</b>       | <b>141</b> |
| <b>AN-3.3.2. Tensão Admissível de Compressão</b>   | <b>141</b> |
| <b>AN-3.4. Cálculo das Tensões Admissíveis</b>     | <b>144</b> |

## Índice de Figuras

|   |    |
|---|----|
| <b>Figura 1.1</b> – Comparação entre as etapas do Projeto Convencional e do Projeto Ótimo | 2  |
| <b>Figura 2.1</b> – Esquema generalizado de alinhamentos de nós                           | 28 |
| <b>Figura 3.1</b> – Algoritmo do Método Simplex   | 34 |
| <b>Figura 3.2</b> – Algoritmo geral para um método da Programação Não-linear              | 36 |
| <b>Figura 3.3</b> – Algoritmo do Método do Gradiente                                      | 38 |
| <b>Figura 3.4</b> – Ilustração do Método das Penalidades                                  | 48 |
| <b>Figura 5.1</b> – Exemplo de arquivo de entrada de dados                                | 72 |
| <b>Figura 5.2</b> – Exemplo alternativo para o bloco 4 de entrada de dados                | 73 |
| <b>Figura 6.1</b> – Treliça plana de 6 nós e 10 barras                                    | 82 |
| <b>Figura 6.2</b> – Treliça plana de 11 nós e 18 barras                                   | 86 |

|   |            |
|---|------------|
| <b>Figura 6.3 – Treliça plana de 11 nós e 23 barras</b>       | <b>90</b>  |
| <b>Figura 6.4 – Torre plana de 22 nós e 47 barras</b>         | <b>94</b>  |
| <b>Figura 6.5 – Treliça plana de 28 nós e 64 barras</b>       | <b>98</b>  |
| <b>Figura AN-3.1 – Definição das propriedades geométricas</b> | <b>136</b> |

## Índice de Tabelas

|  |    |
|--|----|
| <b>Tabela 6.1</b> – Comparação de resultados da análise contínua da treliça de 10 barras _____                               | 82 |
| <b>Tabela 6.2</b> – Perfis disponíveis por grupo de barras usados no processamento <i>GAMS</i> da treliça de 10 barras _____ | 84 |
| <b>Tabela 6.3</b> – Comparação de resultados da análise discreta da treliça de 10 barras _____                               | 84 |
| <b>Tabela 6.4</b> – Comparação de resultados da análise contínua da treliça de 18 barras _____                               | 87 |
| <b>Tabela 6.5</b> – Perfis disponíveis por grupo de barras usados no processamento <i>GAMS</i> da treliça de 18 barras _____ | 87 |
| <b>Tabela 6.6</b> – Comparação de resultados da análise discreta da treliça de 18 barras _____                               | 88 |
| <b>Tabela 6.7</b> – Comparação de resultados da análise contínua da treliça de 23 barras _____                               | 90 |
| <b>Tabela 6.8</b> – Perfis disponíveis por grupo de barras usados no processamento <i>GAMS</i> da treliça de 23 barras _____ | 91 |
| <b>Tabela 6.9</b> – Comparação de resultados da análise discreta da treliça de 23 barras _____                               | 91 |

|   |     |
|---|-----|
| <b>Tabela 6.10</b> – Perfis disponíveis por grupo de barras usados no processamento <i>GAMS</i> da treliça de 47 barras _____                             | 95  |
| <b>Tabela 6.11</b> – Comparação de resultados da análise discreta da treliça de 47 barras _____   | 96  |
| <b>Tabela 6.12</b> – Perfis disponíveis por grupo de barras usados no processamento <i>GAMS</i> da treliça de 64 barras _____                             | 99  |
| <b>Tabela 6.13</b> – Comparação de resultados da análise discreta da treliça de 64 barras _____   | 100 |
| <br>  |     |
| <b>Tabela AN-2.1</b> – Domínios de variáveis em <i>GAMS</i> _____   | 128 |
| <b>Tabela AN-2.2</b> – Procedimentos de resolução em <i>GAMS</i> _____  | 132 |
| <b>Tabela AN-3.1.a</b> – Classificação dos perfis <i>I</i> baseado nos limites da razão largura-espessura _____   | 137 |
| <b>Tabela AN-3.1.b</b> – Classificação dos perfis <i>Box</i> baseado nos limites da razão largura-espessura _____   | 138 |
| <b>Tabela AN-3.1.c</b> – Classificação dos perfis <i>T</i> baseado nos limites da razão largura-espessura _____   | 138 |
| <b>Tabela AN-3.1.d</b> – Classificação dos perfis <i>C</i> (canal) baseado nos limites da razão largura-espessura _____                                   | 139 |
| <b>Tabela AN-3.1.e</b> – Classificação dos perfis Cantoneira, simples e duplos (com abas paralelas), baseado nos limites da razão largura-espessura _____ | 139 |
| <b>Tabela AN-3.1.f</b> – Classificação dos perfis Tubulares baseado nos limites da razão largura-espessura _____  | 140 |

## Resumo

---

Diminuir os custos com a construção de estruturas tornou-se ainda mais importante com a globalização da economia. No caso de treliças, uma estrutura leve, de fácil e rápida execução, o menor custo será representado pelo menor peso que a treliça poderá ter. O presente trabalho trata da otimização de peso de treliças planas com geometria e topologia fixadas.

O processo de otimização proposto consiste em submeter uma configuração inicial de treliça plana a um programa que primeiramente sujeitará a função objetivo a um otimizador de caráter contínuo e não-linear, restritas por funções que representam tensões admissíveis das barras (calculadas segundo a norma norte-americana de tensões admissíveis AISC/ASD 1989), limites para os deslocamentos nodais, áreas das seções transversais, além de equações de equilíbrio estático; com a finalidade de minimizar a área da seção transversal, mantendo o *layout* sugerido inicialmente. De posse desses valores ótimos, submetem-se novamente as mesmas equações a um outro otimizador, de caráter discreto e não-linear, acrescentando o conjunto de restrições com equações para a escolha de perfis disponíveis no mercado. Foi escrito o programa pTRUSS, em linguagem Pascal, que prepara um arquivo com os comandos específicos da linguagem interpretada GAMS. Esse arquivo

gerado pelo programa pTRUSS é submetido ao software de otimização GAMS, que resolve o problema em estudo. Permitem-se dois tipos de análise: análise contínua e análise discreta. No primeiro caso o GAMS utiliza rotinas internas do MINOS 5.1 e no segundo o GAMS utiliza o otimizador *DICOPT*.

Apresentam-se alguns exemplos encontrados na literatura para efeito de comparação de resultados. Compararam-se também os resultados obtidos pela proposta da dissertação com valores calculados com os programas SAP2000 e AutoMetal.

## **Abstract**

---

*Due to the globalization of the economy, cost reduction has become more important. Concerning to the truss, a light, easy and quick build structure, the minimum cost will be represented by minimum weight that the truss could have. In this his work plane truss optimization with fixed geometry and fixed topology is studied.*

*The proposed optimization process consists in applying to an initial configuration plane truss a code where, firstly, the Weight Objective Function will be subjected to a continuum and nonlinear solver, constrained by functions which represent the allowable stress members (according to the AISC-ASD/1989 code), displacement nodal limits and cross section area limits, as well as static equilibrium equations; focusing on minimizing the cross section area and keeping the initial configuration proposed. The optimal values obtained by continuum analysis will be used to submit again the same equations to another solver, now discrete and nonlinear. New equations of restraint that permit to choose available commercial sections were increased on program. Using Pascal language, it was made a program named pTRUSS that prepares a file with specific commands of language interpreted by GAMS. This file created by the*

*pTRUSS* program is submitted to the optimization software GAMS, that solves the problem in analysis. Two types of analyses are permitted: continuum analysis and discret analyses. In the former, the software GAMS uses internal routines of MINOS 5.1 and in the latter, GAMS uses the solver DICOPT.

Examples from the literature are presented in order to compare the results. Commercial softwares as SAP2000 and AutoMetal were used to validate the results obtained by the software GAMS with the values calculated.

# **1. Introdução**

---

## **1.1. Considerações Iniciais**

Com a globalização da economia, seus produtos e serviços, procura-se crescentemente buscar melhor qualidade a menores custos, tornando-se muito importante o projeto de estruturas civis que satisfaçam os requisitos econômicos e que cumpram com segurança as finalidades para as quais foram desenvolvidas.

Para se entender a importância da otimização de estruturas é preciso ressaltar as diferenças básicas entre um projeto convencional e um projeto ótimo (**Fig. 1.1**). O dimensionamento convencional de estruturas visa obter uma configuração de projeto aceitável e adequada aos requisitos funcionais, normalmente regidos por uma regulamentação, se caracterizando por sofrer influência direta do projetista, através de um processo de tentativa e erro, dependendo assim de sua habilidade, experiência e intuição. Tal fato, além de conduzir a uma infinidade de soluções possíveis, pode levar a resultados insatisfatórios, primeiro

porque está sujeito a falhas humanas e depois, não existem garantias de que a solução obtida seja a melhor do ponto de vista econômico; desta forma, a solução ótima só seria obtida eventualmente.

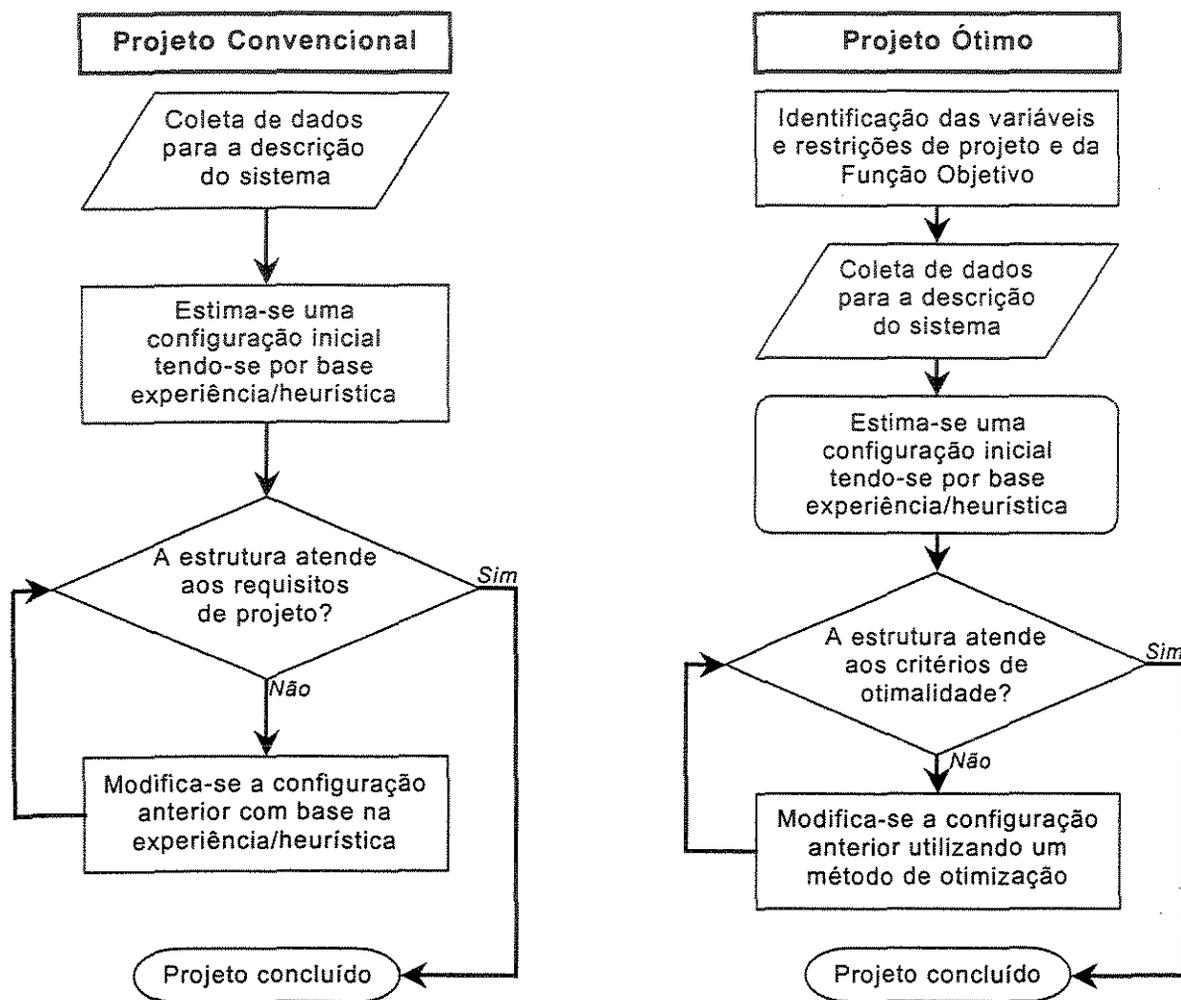


Figura 1.1 – Comparação entre as etapas do Projeto Convencional e do Projeto Ótimo

Já o emprego de uma técnica de otimização possibilita um melhor entendimento do processo de dimensionamento, pois transforma a estrutura do aspecto físico para o aspecto matemático, através de um processo de modelagem matemática da estrutura considerada e todas as suas características. Essa modelagem exige uma função chamada de Função Objetivo, sujeita a diversas restrições, que limitam a forma e a

configuração topológica da estrutura, as formulações para a análise estrutural e as limitações impostas por Normas Técnicas ou pela resistência dos materiais empregados na estrutura. É a Função Objetivo e as equações de restrição que estarão sendo submetidas a algoritmos sistemáticos de buscas e que retornará a solução ótima. Em outras palavras, a otimização estrutural nada mais é do que a união de uma área de engenharia com uma da matemática [HAFTKA e GÜRDAL, 1992].

De fato, o projeto estrutural ótimo consiste na escolha dos materiais, da topologia e da geometria a serem utilizados no sistema, e ainda com o dimensionamento de suas diversas partes constituintes, de forma a atender os requisitos de desempenho, economia e segurança. Para tanto, são definidas uma ou mais funções objetivo, sujeitas ou não a restrições. As condições ou restrições de comportamento impostas são relacionadas ao material escolhido, durabilidade, funcionalidade, confiabilidade, eficiência, estética, recursos disponíveis para análise, etc. Entretanto são adotadas algumas simplificações devido à complexidade de se conseguir uma parametrização que considere todos esses fatores de forma adequada. Deste modo, o projeto estrutural ótimo concentra-se na determinação de uma combinação conveniente obtida através de várias análises visando a concepção ou determinação de uma estrutura de melhor desempenho global dentro dos objetivos estabelecidos.

Então o projetista deve ter a capacidade de definir a Função Objetivo, identificar todas as restrições importantes e os níveis ou limites aos quais as restrições devem ser satisfeitas e incluir essas informações no processo de otimização estrutural. Quanto às restrições, em projetos usuais, elas estão associadas às tensões, deslocamentos, cargas críticas de flambagem, entre outras. Deste modo, para se obter bons resultados com este processo, existem inúmeros aspectos importantes da otimização estrutural que o projetista deve estar atento, tais como o tamanho, forma

e topologia, os diversos tipos de carregamentos a que a estrutura pode ser submetida, as possíveis variações nos sistemas de apoios, os diversos tipos de materiais que podem ser utilizados, considerações de estabilidade, análise estática e dinâmica e ponderações das não-linearidades físicas e geométricas.

## 1.2. Objetivos

Para a otimização de uma estrutura treliçada plana a Função Objetivo pode ser estabelecida como o mínimo da função peso da estrutura, já que se considera que na maioria dos casos, para estruturas metálicas, o peso é proporcional ao custo da treliça quanto às barras, elementos de ligação e custos de montagem. Ressalva-se entretanto que a quantidade de ligações e elementos de barras também pode contribuir para o encarecimento da treliça.

O processo de otimização apresentado neste trabalho consiste em modelar treliças planas na linguagem *GAMS (General Algebraic Modelling System)* e submeter esses modelos a duas formas diferentes de otimização, no que diz respeito às características do espaço solução. Primeiramente o modelo será resolvido com o uso de um otimizador contínuo e não-linear, o *MINOS*. Esta etapa trará resultados que servirão para determinar o vetor de áreas discretas (em função de disponibilidades comerciais) que será utilizado ao submeter o modelo desta vez a um otimizador discreto e não-linear, o *DICOPT*. Desta forma, visa-se testar e mostrar que o procedimento é eficaz para resolver tais problemas.

Esta minimização ocorre em função das áreas da seção transversal das barras, modeladas como variáveis de projeto, e das posições dos nós (obtidas e restringidas a partir do Método dos Deslocamentos), considerando que a estrutura terá sua geometria e topologia fixada; fixou-se também a forma da seção transversal das barras para fins de simplificação dos modelos gerados.

Como os modelos de treliças são grandes, principalmente em virtude do grande número de equações de restrição, houve a necessidade de gerá-lo também de forma sistemática. Para isso, desenvolveu-se um programa em linguagem Pascal para gerar os modelos em GAMS de quaisquer treliças planas, onde o usuário pode optar por gerar um modelo contínuo ou discreto. O programa é denominado pTRUSS.

O programa pTRUSS para a geração dos arquivos de entrada para o programa GAMS está disponível no site [www.carlosfmcortes.hpg.com.br](http://www.carlosfmcortes.hpg.com.br). Estão disponíveis também arquivos com a extensão .DAT, referentes a alguns exemplos, bem como os arquivos .GMS referentes aos respectivos modelos gerados em linguagem GAMS e também os arquivos de saída .LST, gerados pelo programa GAMS.

### **1.2.1. Por que utilizar o GAMS?**

O GAMS é uma linguagem de alto nível desenvolvida para a formulação de modelos matemáticos de Pesquisa Operacional com declarações algébricas bastante concisas. Possui duas características marcantes: trabalha o problema matemático independente do método de

solução; e possui uma estrutura básica para codificar o modelo, consistindo da declaração e definição de conjuntos, dados, variáveis, equações, modelo e saída de resultados [CHATTOPADHYAY, 1999].

A primeira característica é particularmente importante pois permite ao usuário escolher qualquer otimizador, isto é, entre as diversas técnicas de otimização matemática, pode-se resolver o mesmo problema, permitindo-se comparar resultados. A bem da verdade, no *GAMS* se escreve o modelo universal onde seu compilador transforma numa linguagem própria para ser lida pelos otimizadores matemáticos diversos.

Como o *GAMS* foi criado para fins de pesquisa operacional, existe pouca literatura com aplicações na área de Engenharia de Estruturas e otimização estrutural, onde o número de equações definidas é muitas vezes superior ao número de dados computados, tornando o modelo com custo elevado de processamento. Houve o interesse em pesquisar como esta linguagem se comportaria para a modelagem de estruturas, em particular, as treliças planas com as características sugeridas.

### **1.3. Organização**

Como já foi enfatizado, este trabalho versará sobre otimização de treliças metálicas e para tanto, será visto todo o embasamento teórico necessário ao conhecimento do leitor para executar o procedimento proposto, conhecer mais sobre o *GAMS* e os otimizadores que serão

utilizados, e ter uma noção de como funciona as principais técnicas de otimização matemática.

Na seqüência, no capítulo 2 será descrito de forma sucinta a formulação geral para um problema de otimização aplicado a estruturas treliçadas com geometria fixada, suas restrições quanto às tensões, áreas das seções transversais e equações de equilíbrio. Para se ter uma visão mais ampla de um problema mais geral, serão levantados outros aspectos que não foram levados em conta, mas que poderiam entrar no processo de otimização, tais como o alinhamento nodal, custos de fabricação de ligações. Além disso, serão apresentados os conceitos e definições mais importantes para compreender os termos empregados em otimização, bem como os subproblemas possíveis no processo. Fechando este capítulo, tem-se um histórico sobre os principais trabalhos que contribuíram para o desenvolvimento de processos ótimos, especialmente de treliças.

No 3º capítulo serão mostrados os principais métodos de Programação relacionados a otimização de estruturas. Destacam-se a Programação Linear, com seu método Simplex e a Programação Não-linear sem restrições e com ou sem restrições.

Como funcionam os otimizadores do *GAMS* adotados aqui, o *MINOS*, para análise contínua, e o *DICOPT*, para análise discreta, estará descrito no 4º capítulo. Nele ter-se-á um breve histórico do seu desenvolvimento, como funciona o seu algoritmo e como trabalham em parceria com o *GAMS*. Para completar, um tutorial a respeito do *GAMS*, com as informações necessárias para se modelar o problema de otimização para esta linguagem, estará disponível no Apêndice AP-1 deste trabalho.

Para concluir o corpo deste trabalho, os capítulos 5 e 6 trarão respectivamente a descrição do programa em linguagem Pascal, denominado pTRUSS, o qual foi elaborado pelo autor. Esse programa gerará os arquivos de dados, escritos na linguagem numa linguagem especializada, necessária para que os arquivos sejam interpretados pelo software *GAMS*. Serão apresentados alguns exemplos de treliças, retirados de trabalhos de outros autores, para poder assim comparar os resultados obtidos com a proposta da dissertação. Ao final apresentam-se algumas discussões e conclusões, com sugestões para continuidade do trabalho.

Os Anexos mostrarão uma das tabelas de perfis comerciais disponíveis no mercado e utilizados nos ensaios discretos das treliças (AN-1), um tutorial sobre a linguagem de modelagem *GAMS* (AN-2) e uma transcrição simplificada da Norma Técnica adotada, a AISC-ASD (Allowable Stress Design, do American Institute of Steel Construction), como suporte regulamentador das estruturas treliçadas planas (AN-3).

## **2. Otimização**

---

### **2.1. Definições e Conceitos**

Para proporcionar uma melhor compreensão dos termos empregados neste trabalho, bem como na literatura sobre otimização em geral, apresentam-se na seqüência as definições e conceitos pertinentes.

A **Função Objetivo** constitui-se na função a ser minimizada (ou maximizada). É a representação matemática do critério, o qual é influenciado pelas variáveis de projeto e sob o qual o sistema estrutural deverá ser otimizado. A Função Objetivo normalmente é *multidimensional*, ou seja, a função depende de mais de uma variável de projeto; caso contrário, ela será *unidimensional*. Quando essa função apresenta um único ponto de mínimo (ou máximo), ela é dita *unimodal*, como é o caso das funções lineares; quando não, ela é *multimodal*.

O problema de otimização pode ter vários critérios a serem otimizados; neste caso, a Função Objetivo é chamada *multiobjetivo*. Para se obter uma solução mais geral, pode-se considerar na modelagem quesitos como mão-de-obra, relações custo-benefício, transporte de material, tempo de execução e assim por diante.

As **Variáveis de Projeto** são determinadas pelos parâmetros tratados como variáveis no processo do Projeto Ótimo. Estas variáveis podem ser do tipo contínua ou discreta.

As **Restrições de Projeto** podem ser basicamente de dois tipos: *restrições de comportamento e de resistência*, que normalmente são funções que representam, por exemplo, as condições de tensões, deslocamento nodal, flambagem, etc.; e *restrições de funcionalidade*, que atuam diretamente sobre as variáveis de projeto, limitando seus valores.

O **Espaço Solução** é o conjunto, espaço ou região, que compreende as soluções factíveis ou viáveis do problema a ser otimizado.

O **Ponto Ótimo** é definido pelo vetor das variáveis de projeto que otimizam a Função Objetivo satisfazendo as restrições do modelo. O **Valor Ótimo** é o valor da Função Objetivo no ponto ótimo. A **Solução Ótima** é o par ordenado definido pelo ponto ótimo e valor ótimo. Pode-se ter uma solução ótima local ou global, dependendo das condições de otimalidade serem satisfeitas numa vizinhança do ponto ou em toda a região de factibilidade.

## 2.2. Subproblemas em Otimização Estrutural

Na otimização de treliças pode-se considerar que o projeto ótimo pode envolver até três subproblemas complementares diferentes, cada qual com uma metodologia própria. Um deles, a **Otimização Dimensional**, que será enfocada neste trabalho, consiste na determinação da geometria ou das dimensões ótimas da seção transversal dos elementos da treliça. Outro, a **Otimização Geométrica**, define a localização ótima dos pontos nodais no espaço de coordenadas. O terceiro subproblema, de **Otimização Topológica**, consiste na determinação do número de pontos nodais e de elementos da estrutura, bem como a relação de conectividade entre os mesmos; particularmente, caracteriza-se por envolver variáveis booleanas no processo, ao contrário dos dois primeiros, que se utilizam de variáveis reais.

Estes subproblemas estão em geral interligados, sendo desejável a resolução destes simultaneamente, a fim de se conseguir a estrutura ideal e indiscutível. Na prática, é mais comum otimizar apenas o problema dimensional, pois os projetistas sempre lidam com estruturas com configuração já definida. No caso de treliças de uso aeroespacial, onde o peso é um fator preponderante em relação aos demais, trabalha-se com todos os subproblemas [FALCÓN, 1991].

### 2.3. Histórico

A seguir apresenta-se a evolução do estudo de otimização ao longo dos tempos, deste o surgimento da idéia do ótimo e os primeiros trabalhos que a consolidaram. Depois, uma revisão bibliográfica resumida do desenvolvimento da otimização estrutural, procurando-se enfatizar as etapas decisivas neste desenvolvimento, os trabalhos dos precursores desta área e finalizando com os principais trabalhos mais recentes específicos relativos a otimização de treliças metálicas.

A otimização faz parte da índole humana. Desde seu surgimento, o *homo sapiens* vem se dedicando a minimizar esforços e maximizar os retornos de alguma atividade por ele desenvolvida, que no princípio de sua existência limitava-se à própria sobrevivência. Essa preocupação pelo ótimo se encontra registrado nos trabalhos de Euclides (séc. III a.C.), que no seu terceiro livro procura encontrar a maior e a menor distância de um ponto a uma circunferência e no seu quarto livro expõe uma forma de obtenção da área máxima de um paralelogramo, dado seu perímetro. O matemático Héron de Alexandria (séc. III a.C.) afirmou que a luz vai de um ponto a outro pela trajetória mínima. Mais tarde, Fermat sustentou que tal mínimo se referia ao tempo de percurso.

No século XVII surge o cálculo diferencial e integral por necessidade, pois a humanidade começava a entrar na Revolução Industrial, a qual tornava imperativo o uso pleno dos recursos, ou seja, das matérias-primas utilizadas na fabricação de produtos. A indústria, mesmo que primitiva, começa a se preocupar com a maximização dos

lucros e a minimização de custos. Neste sentido contribuíram os matemáticos Newton, Leibniz, Lagrange e Bernoulli.

Entretanto, somente com o início da II Guerra Mundial é que a Otimização surgiu como ramo científico independente, à época chamada de Pesquisa Operacional. Esta nova área da Matemática começou a investigar de forma sistemática e racional os processos envolvidos na realização de uma atividade produtiva, mesmo que de início com finalidades bélicas. A Pesquisa Operacional começa a servir, a partir de suas técnicas quantitativas, de ajuda no processo de tomada de decisão em problemas relacionados ao controle de sistemas. Embutido nesta problemática militar, especificamente da logística de tempo de guerra, é que surge o fato consolidador da Otimização: a criação do método Simplex, que é a forma sistemática de resolução de problemas lineares e que se revelou de uma eficácia extraordinária [LOESCH e HEIN, 1999].

O estudo da Otimização Estrutural em que a configuração dimensional e geométrica é variável de projeto não é novo, tendo sido iniciado por J. Maxwell em 1854 e posteriormente continuado por A. Michell em 1904. Porém os resultados obtidos não permitiam aplicações práticas de interesse.

Na década de sessenta, após um grande período de pouca evolução nesta área, surgiram alguns trabalhos importantes, como o de DORN et al. (1969; em MOURA, 1977). Esses autores desenvolveram um método pelo qual era escolhido um conjunto de pontos no espaço onde poderiam ser situados nós de treliças, e determinaram como esses nós deveriam ser conectados de forma a suportar o carregamento exterior, com a mínima quantidade de material, problema esse resolvido por meio

de Técnicas de Programação Linear. Infelizmente, limitadas pelo número de variáveis de projeto que poderiam ser consideradas, os métodos eram restritos às aplicações práticas para estruturas de pequeno porte, necessariamente treliças isostáticas, com uma única carga atuante e poucas barras.

A partir da década de setenta, com o advento dos computadores e o desenvolvimento de novas técnicas de programação não-linear deu-se ânimo a estes estudos, onde já se consideravam as coordenadas nodais independentes como variáveis de projeto, permitindo o tratamento de problemas de grande porte com maior eficiência, observando-se o alto grau de não-linearidade do subproblema de otimização geométrica, geralmente mais acentuado que no caso da otimização dimensional.

Assim, DOBBS e FELTON (1969) investigaram o efeito das condições de múltiplo carregamento sobre a configuração geométrica de treliças, utilizando o mesmo princípio de Dorn et al., a partir de uma configuração inicial que era modificada através do Método do Gradiente (*Steepest Descent*). Ponderaram também que os elementos estariam sujeitos ao colapso por flambagem local e de Eüler, e, ao contrário do que pensavam seus predecessores, concluíram que o peso mínimo da treliça não requeria necessariamente que as barras estivessem completamente tensionadas.

Seguindo outra linha de modelagem, VANDERPLAATS e MOSES (1973) aboliram a idéia do conjunto de nós possíveis e trabalharam com uma configuração geométrica inicial para a estrutura com limites para o posicionamento dos nós e consideraram, além do múltiplo carregamento, a hiperestaticidade das treliças. Seu modelo era submetido inicialmente a otimização da área das seções transversais e depois sofria a otimização geométrica da estrutura, também utilizando o Método do Gradiente. Mais

tarde, PEDERSEN (1972) desenvolveu um procedimento com sucessivas iterações para realizar a mesma otimização, incrementada por limites críticos móveis. Depois, ALSPAUGH e KUNOO (1974) inovaram e fizeram a análise estrutural de suas treliças pelo Método de Deslocamento de Elementos Finitos e usaram o Método do Gradiente para otimizar simultaneamente dimensão e geometria com a inclusão de restrições de desigualdade para limitar as variáveis de estado (tensão e deformação) e projeto (área da seção e deslocamento nodal).

No Brasil, MOURA (1977) otimizou nos mesmos moldes sugeridos por Vanderplaats e Moses treliças tridimensionais com uma variação do Método do Gradiente adaptado por Fletcher e Reeves em 1964, que apresentava a vantagem de, além da simplicidade de programação, ter uma convergência mais rápida.

IMAI e SCHMIT (1981) apresentaram um método primal-dual, chamado de Método do Multiplicador, em conjunto com expansões das Séries de Taylor de segunda-ordem para funções de deslocamento, na tentativa de garantir uma eficiência computacional razoável e superar os problemas graves de não-linearidade, causados pelo uso de variáveis de projeto de diferentes características e comportamentos (por exemplo, discretas, contínuas, inteiras).

ALLWOOD e CHUNG (1984) otimizaram o peso de treliças utilizando o Método de Newton-Raphson, auxiliado por aproximações de primeira-ordem das equações de restrição de deslocamento nodal e tensão das barras, e concluíram que o método apresenta excelente desempenho com relação a outros métodos, inclusive para estruturas muito grandes e com diversas restrições.

SEPÚLVEDA e CASSIS (1986) desenvolveram um método de otimização envolvendo variáveis contínuas, discretas e mistas. Neste método, a formulação original, caracterizada exclusivamente por problemas de minimização não-lineares e restrições de desigualdade, é resolvida por meio de uma técnica dual aplicada sobre uma seqüência de aproximações de primeira-ordem da Série de Taylor, onde a Função Objetivo é maximizada.

O uso da expansão de primeira-ordem da Série de Taylor conseguiu reduzir o grau de dependência entre as variáveis dimensionais e geométricas, e conseqüentemente aumentou a precisão da aproximação da função que representava as forças atuantes nas barras utilizada no trabalho de HANSEN e VANDERPLAATS (1990). Além disso, desenvolveram um código iterativo envolvendo sub-rotinas que analisavam a estrutura pelo Método dos Elementos Finitos, calculava os gradientes das forças nas barras (em função da área e do posicionamento nodal simultaneamente) e otimizava a estrutura.

Ampliando os objetivos de otimização de treliças, SAKA e ULKER (1991) desenvolveram um algoritmo para treliças geometricamente não-lineares com critérios de otimalidade aproximados, onde demonstraram que poderiam conseguir uma redução adicional no peso global e levaria em conta o real comportamento da estrutura. Isto foi conseguido com o incremento da Matriz de Rigidez Elástico-linear a Matriz de Rigidez Geométrica, usando de um processo iterativo linear.

Um método para minimização com configuração ótima foi desenvolvido por LASSEN (1993), onde sua abordagem consistia em uma técnica de reanálise em combinação com uma sub-otimização para cada grupo de barras da treliça, usando Programação Quadrática Seqüencial.

No modelo foram consideradas ainda a maximização das tensões de Von Mises e a possibilidade de fadiga nos pontos nodais.

STANDER e SNYMAN (1993) desenvolveram o Método de Direções Interiores Factíveis a partir da modificação de uma técnica de Programação Linear, para aplicar a problemas de otimização com restrições não-lineares. Concluíram, em comparação com outros métodos, que a solução convergia com poucas iterações e que possuía boa estabilidade comum aos métodos baseados no Simplex.

Após um levantamento em que concluiu que restrições de estabilidade de problemas de otimização eram tratados como problemas genéricos de autovalor, LEVY (1994) apresentou uma formulação que inclui as condições de estabilidade linear, não-linear e de Eüler, através da derivada primeira do Lagrangeano destas funções, na configuração ótima de quaisquer casos de treliças, utilizando Programação Não-linear. Na comparação com os resultados envolvendo a técnica do autovalor, descobriu que esta abordagem gerava resultados superestimados.

GUTKOWSKY e BAUER (1994) basearam seu processo de otimização na solução simulada de um sistema de equações e nas condições necessárias de Kuhn-Tucker, e incorporaram, além das restrições de comportamento, restrições de sensibilidade em um problema de otimização, com o objetivo de avaliar a performance da solução e a possibilidade da violação dos valores limites das restrições, principalmente no que se refere as diferenças entre o projeto e a estrutura real.

Também considerando a sensibilidade do problema, SCHMIT Jr. e LAI (1994) apresentaram o Método do Gradiente Conjugado Pré-Condicionado como uma alternativa aos métodos clássicos de

decomposição direta. Neste método, dois vetores de variáveis iniciais são propostos e usados para resolver o sistema de equações lineares, associado com a análise estrutural pelo Método dos Elementos Finitos.

CHENG (1995) estudou alguns aspectos da formulação de problemas de topologia de treliças, no que se refere a otimização do peso estrutural sujeito a restrições de tensões e flambagem local. Considerou na sua comparação o fato de que a continuidade das funções de restrições e/ou da função objetivo é um fator importante para a determinação de estruturas matemáticas de problemas. Assim, mostrou que uma função de tensão das barras tem descontinuidade quando a área da seção transversal tende a zero, e que a função de deslocamento é contínua nesta área. Baseado nisso, CHENG mostrou as características do domínio factível para função de tensão e o ótimo global para problemas de otimização com restrições de tensão e/ou flambagem local, e concluiu que estes problemas são de programação matemática com funções descontínuas e que de certa forma poderiam ser resolvidos discretamente.

OBERNDORFER et al. (1996) discutiram duas abordagens estruturais para otimização de treliças; a primeira delas levava em conta a formulação tradicional para minimização do volume da treliça e a outra minimizava uma aproximação da energia baseada no Método dos Trabalhos Virtuais. Ambos os métodos de otimização topológica seleciona um subconjunto ótimo de barras oriundo de um conjunto de todas as possíveis barras definidas sobre uma lista discretizada. Mostraram também vantagens e desvantagens e que restrições existem onde a formulação se torna equivalente, além da importância de se considerar a influência da flambagem sobre o layout ótimo.

BOFFEY e YATES (1997) determinaram a área da seção transversal dos membros da estrutura treliçada otimizada através da programação não-linear discreta. Para isso, utilizaram subproblemas duais do Lagrangeano para a solução, resolvido através de um método desenvolvido a partir do Método Simplex. Estruturas próprias do problema permitiram o uso da tabulação do Simplex que são muito reduzidas em tamanho, tornando a resolução dos subproblemas mais rápida.

BARBIERI e LOMBARDI (1998) trabalharam com a otimização de treliças segundo a minimização do peso sujeito a restrições de alguns deslocamentos e as tensões nas barras, entretanto consideraram a treliça fabricada de um material desconhecido, ou seja, não se conhecia o módulo de elasticidade das barras, portanto estas poderiam apresentar diferentes valores de rigidez, com um limitante para esta variação. Deste modo, desenvolveram um processo para a otimização de tal forma que a solução ótima permanece viável para cada valor que a constante do material possa tomar. Nesse processo uma abordagem probabilística para esta incerteza é utilizada, bem como uma variação do módulo do material com uma distribuição uniforme dentro de um domínio convexo e linear. BARBIERI e LOMBARDI utilizaram um método de dois passos para incluir a incerteza na otimização, onde uma aproximação quadrática diagonal foi usada para a função objetivo e para as restrições.

YAM e LI (1998) desenvolveram uma metodologia que utilizava as relações de Programação Não-linear e dava condições para a determinação da melhor estrutura entre duas com configuração diferentes, entretanto próximas. Esta metodologia fazia comparações entre pares de treliças sugeridas pelo projetista, onde eram fixados os nós de apoio, os nós submetidos a cargas e aqueles impostos por estética; em cada

configurações as áreas eram minimizadas de forma que as barras estivessem submetidas às suas cargas críticas.

LAMBERTI e PAPPALETTERE (2000) apresentaram o efeito da solução numérica de diferentes critérios para definição dos limites móveis, através de aproximações a serem adotadas na resolução pelo Método de Programação Linear Seqüencial, caracterizado por ser um procedimento que consiste na formulação e resolução de uma série de subproblemas linearmente aproximados, onde cada solução intermediária é o ponto de partida para o subproblema subsequente.

GIL e ANDREU (2001) apresentaram um método para a identificação de uma topologia de seções transversais ótimas para estruturas treliçadas planas sob restrições geométricas e de tensões. Tal problema de configuração e formato foi definido pelo desconhecimento das coordenadas nodais e associado com um problema de configuração paramétrico definido pela seção transversal. Este método inclui uma abordagem para juntar os dois problemas e resolver as dificuldades provenientes da combinação das variáveis de otimização de cada natureza diferente.

Somente na década de 1990 é que surgiram no Brasil significativos trabalhos na área de otimização de treliças, sendo destaque o de PRUDENTE (1998), que apresentaram um processo de busca da solução de mínimo peso, utilizando-se perfis comerciais comuns. O processo baseou-se no método do gradiente, modificando-se alguns passos para considerar a não-linearidade das tabelas e perfis usados; além disso, utilizaram funções de penalidade para transformar a solução de problemas restritos em um conjunto de subproblemas não-restritos.

Outro trabalho foi apresentado por KOEHLER e ALVES (1999), onde usaram dois procedimentos no processo de otimização. No primeiro, utilizaram uma adaptação do Método dos Multiplicadores, onde incluíram o coeficiente de Lagrange para transformar o problema de programação matemática não-linear, a fim de minimizar o par ótimo que envolve o vetor das variáveis de projeto e o vetor dos coeficientes supracitados. No segundo procedimento, após transformar o problema de otimização tradicional, resolveram uma seqüência de problemas sem restrição, gerados no procedimento anterior, utilizando o Método Quasi-Newton, onde a inversa da matriz Hessiana, necessária neste método, fora aproximada pelo Método BFGS (Broyden-Fletcher-Goldfarb-Shanno).

Também em 1999, VECCI e LAU utilizaram expansões lineares da Série de Taylor para a construção das aproximações para as tensões normais atuantes nos elementos de barra, onde os deslocamentos foram linearmente aproximados nas variáveis de coordenadas nodais e áreas da seção transversal. Pelo fato de considerarem como restrição o alinhamento de alguns nós, essas aproximações foram derivadas a primeira vez em função das variáveis nodais dependentes e posteriormente, em função das independentes. Outra peculiaridade no trabalho de Vecci e Lau é o fato de utilizarem o Método das Direções Viáveis Modificado para o algoritmo de otimização.

Por fim, GUILHERME et al. (1999) apresentaram uma metodologia de otimização topológica de estruturas, restringida com critérios de flambagem e flexibilidade. Partindo de um conjunto de barras possíveis entre os nós da treliça, esta metodologia seleciona as barras necessárias para garantir a estrutura mais econômica, utilizando o Método de Programação Linear Seqüencial para solucionar o problema de otimização, onde os gradientes da função objetivo e das restrições são obtidos analiticamente ou por diferenças finitas.

## 2.4. Formulação do Problema de Otimização

De maneira geral, a grande maioria dos autores formulam o problema de otimização da forma a seguir. Algumas diferenças ocorrem somente na inclusão ou exclusão de um ou outro fator [SILVA, 1992].

No caso deste trabalho, o objetivo será encontrar o menor valor para o peso da treliça, conforme a Eq. 2.1, cujas variáveis de projeto serão somente as áreas da seção transversal das barras, em virtude de que nesta dissertação a geometria da treliça é fixada.

$$W = \sum_{i=1}^{NE} \rho_i A_i L_i \quad (2.1)$$

onde:  $\rho_i$  : massa específica do elemento  $i$ ;  
 $A_i$  : área do elemento  $i$ ;  
 $L_i$  : comprimento linear do elemento  $i$ ;  
 $NE$  : número de elementos da treliça.

Caso toda a treliça seja do mesmo tipo de material a função expressa na Eq. 2.1 modifica-se para:

$$W = \rho \cdot \sum_{i=1}^{NE} A_i L_i \quad (2.2)$$

Para completar o processo, é necessário restringir o espaço solução da Eq. (2.1). Assim teremos:

**a. Restrição às Tensões Normais:**

$$\bar{\sigma}_i^c \leq \sigma_{ik} \leq \bar{\sigma}_i^t \quad (2.3)$$

onde:  $\bar{\sigma}_i^c$  : tensão admissível à compressão no elemento  $i$ , considerada negativa em estruturas metálicas;

$\bar{\sigma}_i^t$  : tensão admissível à tração no elemento  $i$ , considerada positiva;

$\sigma_{ik}$  : tensão no elemento  $j$  devido ao carregamento  $k$ ;

$k$  : varia de 1 até número de condições de carregamento.

**b. Restrição às Tensões de Flambagem:**

$$\bar{\sigma}_i^b \leq \sigma_{ik} \quad (2.4)$$

onde:  $\bar{\sigma}_i^b$  : tensão admissível de flambagem de Eüler no elemento  $i$ .

O efeito de flambagem ocorre quando o elemento de barra está sujeito a compressão. Portanto, segundo Eüler, podemos escrever:

$$\frac{\pi^2 \cdot E_i \cdot I_i}{A_i \cdot l_i^2} \leq \sigma_{ik} \quad (2.5)$$

onde:  $E_i$  : módulo de elasticidade longitudinal do material do elemento  $i$ ;

$l_i$  : comprimento de flambagem dado por  $l = \frac{L}{f}$ ;

$f$  : fator de flambagem; para barras bi-rotulados, que é o caso comum em elementos de treliças,  $f = 1.0$ ;

$I_i$  : menor momento de inércia da seção do elemento  $i$ .

De maneira geral a inequação que generaliza as restrições à tensão é dada por:

$$\max(\bar{\sigma}_i^c, \bar{\sigma}_i^b) \leq \sigma_{ik} \leq \bar{\sigma}_i^t \quad (2.6)$$

### c. Restrição aos Deslocamentos Nodais

$$u_j^L \leq u_{jk} \leq u_j^U \quad (2.7)$$

onde:  $u_j^L$  : limite inferior (*lower*) para restrição no grau de liberdade  $j$ ;

$u_j^U$  : limite superior (*upper*) para restrição no grau de liberdade  $j$ ;

$j$  : varia de 1 até o número de graus de liberdade;

$u_{jk}$  : deslocamento nodal no grau de liberdade  $j$  devido ao carregamento  $k$ .

### d. Restrição a Área da Seção Transversal na Análise Contínua

$$A_i^L \leq A_{ik} \leq A_i^U \quad (2.8)$$

onde:  $A_i^U$  : limite inferior para área do elemento  $i$ ;

$A_i^L$  : limite superior para área do elemento  $i$ ;

$A_{ik}$  : área no elemento  $i$  devido ao carregamento  $k$ .

Esta restrição pode considerar a imposição de grupos de barras para que se tenha a mesma área, material, comprimento ou inércia. Na modelagem, ao agrupamento de áreas de barras é feito igualando as mesmas entre si.

#### e. Restrição para a Escolha das Áreas na Análise Discreta

As áreas podem estar vinculadas a perfis comerciais, e portanto com variação descontínua. Para lançar mão desta restrição, trabalha-se simultaneamente com as **Equações 2.9 e 2.10**. A primeira vincula uma variável binária ao vetor de perfis disponíveis, determinado pelo projetista em virtude da demanda de mercado. A segunda equação garante que apenas um perfil da biblioteca dos perfis disponíveis será escolhido. Assim:

$$A_{ik} = \sum_{l=1}^{NPD} \phi_{il} \cdot APD_l \quad (2.9)$$

$$\sum_{l=1}^{NPD} \phi_{il} = 1 \quad (2.10)$$

onde:  $\phi_{il}$  : variável binária que para cada barra  $i$  assumirá o valor 1 se o perfil disponível  $l$  for ótimo e 0 para os demais perfis;  
 $APD_l$  : área do perfil disponível;  
 $NPD$  : número de perfis disponíveis.

## f. Equações de Equilíbrio

O deslocamento dos nós quando sujeito a uma condição de carregamento é obtido pelo Método dos Deslocamentos, em virtude da Matriz de Rigidez da estrutura ser computacionalmente mais fácil de ser obtida do que a Matriz de Flexibilidade (Método dos Esforços). As restrições representadas na **Equação 2.11** são equações de equilíbrio estático da treliça:

$$K(A_i, x_i, y_i) \cdot U = F \quad (2.11)$$

onde:  $K(A_i, x_i, y_i)$  : Matriz de Rigidez Global da estrutura em função do vetor de variáveis de área  $A_i$  e das coordenadas nodais  $(x_i, y_i)$ ;

$U$  : vetor de deslocamentos nodais;

$F$  : vetor das ações externas.

Assim como se pode escrever a Matriz de Rigidez em função das áreas das seções transversais das barras ( $A_i$ ), pode-se também escrever em função do posicionamento nodal ( $Y_j$ ) ou ainda em função de ambas as variáveis ( $A_i, Y_j$ ).

### 2.4.1. Outras Restrições

Outras restrições poderiam fazer parte da formulação de um

problema de otimização de treliças, mas não foram levada em contas nesta dissertação.

Quando se trabalha com a otimização geométrica da treliça, a restrição representada pela **Equação 2.12**, de posicionamento nodal, se torna fundamental [KOEHLER e ALVES, 1999].

$$Y_j^L \leq Y_{jk} \leq Y_j^U \quad (2.12)$$

onde:  $Y_j^L$  : limite inferior do posicionamento nodal segundo o grau de liberdade  $j$ ;

$Y_j^U$  : limite superior do posicionamento nodal segundo o grau de liberdade  $j$ ;

$Y_{jk}$  : posicionamento nodal do grau de liberdade  $j$  devido ao carregamento  $k$ .

Entretanto, esta restrição pode sofrer um acréscimo no seu comportamento caso o projetista resolva manter obrigatoriamente alguns nós alinhados. Este tipo de restrição permite que as barras adjacentes se mantenham alinhadas, para facilitar o processo de fabricação e montagem da estrutura ou por imposição do projeto arquitetônico, muito comum na prática para o uso de treliças. Entretanto há poucos trabalhos que tratam deste assunto, provavelmente em virtude da maioria dos trabalhos de otimização estarem voltados para aplicações na indústria aeroespacial, onde o peso estrutural representa um fator mais crítico que os custos com fabricação [VECCI e LAU, 1999].

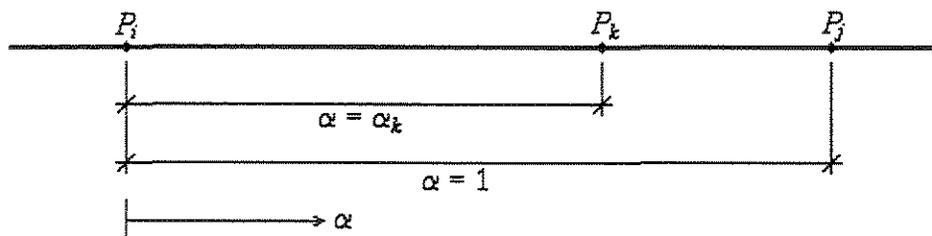
Para a implementação desta restrição deveriam ser utilizados vínculos não lineares. Isto se deve ao fato de que as coordenadas dos nós restritos devam satisfazer a equação da reta pré-definida e que esta

reta, em geral, pode se deslocar e/ou ter a sua orientação alterada. Esta restrição é melhor satisfeita quando se faz uso de vínculos nas variáveis de coordenada; estes vínculos, lineares ou não-lineares, são relações explícitas estabelecidas e tornam a otimização mais eficiente.

Para determinar a relação das restrições de alinhamento são utilizados três pontos colineares,  $(P_i, P_j, P_k)$ , sendo os dois primeiros definidores da reta  $r$  como ponto inicial e final, respectivamente, conforme mostra a **Fig. 2.1**. Assim, estes pontos ficam relacionados da seguinte forma:

$$P_k = P_i + \alpha_k \cdot (P_j - P_i) \quad (2.13)$$

onde  $P_k$  é um ponto alinhado à reta  $r$ ; observe na **Fig. 2.1** que caso  $P_k$  esteja à direita de  $P_j$ ,  $\alpha_k$  será maior que 1 e caso esteja à esquerda de  $P_i$ ,  $\alpha_k$  assume valores negativos.



**Fig. 2.1** – Esquema generalizado de alinhamento de nós

Pode-se restringir ainda o comprimento das barras; permitir a colocação ou a retirada de outras barras de forma a minimizar o comprimento de flambagem e a área da seção da barra solicitada, o que caracteriza uma otimização topológica; considerar efeitos de vibração, ressonância e choque, efeitos térmicos e de recalques; pode-se

considerar os momentos residuais das ligações metálicas, inclusive ponderando estas como rígidas, entre outras.

É importante salientar que no presente trabalho, pela opção de se trabalhar com geometria estrutural fixada, a restrição dada pela **Equação 2.13** bem como as demais consideradas neste item não foram implementadas no programa pTRUSS e na análise usando o programa GAMS, servindo apenas como ilustrativo e como referência para trabalhos futuros.

## 3. Programação Matemática – Técnicas de Otimização

---

### 3.1. Introdução

A fim de demonstrar os fundamentos teóricos das formulações mais empregadas na resolução de problemas afins de otimização, são descritos aqui, de forma sucinta, os principais métodos dentro da *Programação Matemática*.

Os métodos da Programação Matemática podem ser classificados em Diretos e Indiretos [BAZARAA e SHETTY, 1979]. Os *Métodos Diretos* exploram pontos de uma região viável, segundo uma certa estratégia (sistemática ou aleatória) e compara os valores da Função Objetivo; já os *Métodos Indiretos* (regidos por Critérios de Otimalidade), que são as bases da otimização clássica, baseiam-se em fundamentos matemáticos tais como: análise funcional, cálculo integral, cálculo variacional, problema de autovalor, álgebra linear, entre outros; que permitem o

desenvolvimento de algoritmos capazes de buscar um ponto extremo da função, ponto este que deve satisfazer as Condições de Otimalidade, chamadas de *Condições de Kuhn-Tucker* [HAFTKA e GÜRDAL, 1992].

A Programação Matemática pode, de acordo com a Função Objetivo e/ou com as restrições do problema, ser classificada como:

- *Programação Linear*: quando a função objetivo e todas as restrições são funções lineares das variáveis de projeto;
- *Programação Não-linear*: quando a função objetivo ou pelo menos uma das restrições não são funções lineares das variáveis de projeto.

Dentro destas duas classes da Programação Matemática existem vários métodos de otimização que podem ser empregados, submetidos ou não a restrições, além dos métodos que fazem parte da *Programação Multiobjetivo*.

Na dissertação o autor procurou apresentar, de forma bastante resumida, os métodos de otimização mais empregados.

### **3.2. Programação Linear**

Apesar da facilidade de implementação, a aplicação em um processo de otimização em engenharia estrutural utilizando esta técnica é ainda mais restrita, pois como se sabe a maioria dos problemas reais são

governados por funções não-lineares das variáveis de projeto.

A formulação padrão para a Programação Linear é dada da seguinte forma:

$$\begin{array}{ll} \text{Minimize/Maximize} & f(\bar{x}) = c^T \cdot \bar{x} \\ \text{Sujeito à} & A \cdot \bar{x} = b \\ & \bar{x} \geq b \end{array} \quad (3.1)$$

onde:  $c$  : vetor ( $n \times 1$ );

$A$  : matriz ( $m \times n$ );

$B$  : vetor ( $m \times 1$ );

$\bar{x}$  : vetor ( $n \times 1$ ).

Quando é possível empregar a Programação Linear, o método de maior aplicação é o Método Simplex [LOESCH e HEIN, 1999].

Como este método é baseado na resolução de sistemas de equações lineares, algumas definições são indispensáveis [PUCCINI, 1978]:

- *Solução básica*: dado um sistema  $A \cdot \bar{x} = b$  de  $m$  equações lineares simultâneas e  $n$  incógnitas, onde o número de soluções é finito e o posto da matriz  $A$  é  $m$  (tal que  $m < n$ ), pode-se selecionar  $m \times m$  submatrizes não-singulares de  $A$  e colocar as restantes  $n - m$  variáveis nulas; a solução do sistema resultante das  $m$  variáveis e  $m$  equações é chamada de *solução básica*.
- *Variáveis básicas*: são as  $m$  variáveis associadas com a solução básica.
- *Variáveis não-básicas*: são as  $n - m$  variáveis nulas.

- *Base*: é a matriz  $m \times m$  não-singular, também chamada de *matriz básica*.

Este método se baseia em três teoremas [PUCCINI, 1978]:

- *Teorema I*: "O conjunto solução de todas as soluções compatíveis do modelo de Programação Linear é um conjunto convexo."
- *Teorema II*: "Toda solução compatível básica do sistema  $A \cdot \bar{x} = b$  é um ponto extremo do conjunto de soluções compatíveis, isto é, do conjunto convexo  $C$  do *Teorema I*."
- *Teorema III*: "Se a função objetivo possui um máximo (ou mínimo) finito, então pelo menos uma solução ótima é um ponto extremo do conjunto convexo  $C$  do *Teorema I*."

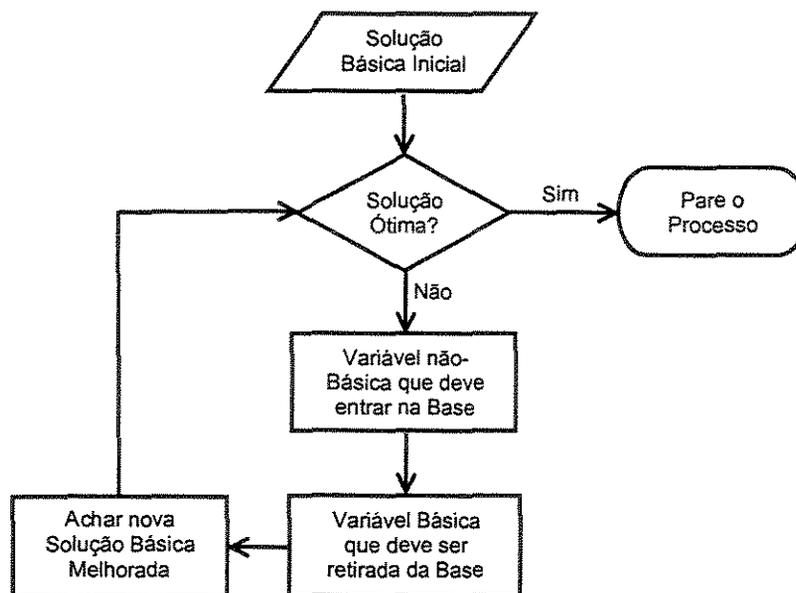


Fig. 3.1 – Algoritmo do Método Simplex

O procedimento geral para resolver um problema da Programação Linear pelo Método Simplex consiste em ir de uma base para outra base melhorada até que o ponto ótimo seja alcançado. Os passos básicos para

a resolução de problemas de otimização pelo Método Simplex são mostrados no fluxograma da **Fig. 3.1**.

Para a resolução através deste método é comum se utilizar *variáveis de folga*, para transformar as restrições de desigualdade em restrições de igualdade, e *variáveis artificiais* nas restrições originalmente de igualdade, com o objetivo de se colocar o problema na chamada "*forma padrão*."

Os pontos críticos deste método são: como passar de uma solução compatível básica para outra; e como definir quais os critérios para se obter uma nova solução compatível básica viável e melhorada [PUCCINI, 1978].

### **3.3. Programação Não-linear**

A Programação Não-linear trata da análise e resolução de problemas semelhantes ao que é exposto na seção "Função Objetivo e Restrições". Supõe-se aqui que as restrições e a função objetivo sejam funções contínuas em um domínio de soluções  $\bar{X}$  e este estando contido em um espaço  $R^n$ . Pode haver, no entanto, problemas sem qualquer restrição em  $\bar{X}$ .

Os algoritmos de solução de um problema da Programação Não-linear possuem uma forma geral de resolução iterativa sobre a

**Equação 3.2**, onde parte-se de um ponto inicial  $x_0$  e, se “bem sucedidos”, convergem para um ponto ótimo  $x^*$ .

$$\bar{x}_{k+1} = \bar{x}_k + \alpha_k \cdot \bar{d}_k \quad (3.2)$$

onde:  $\bar{x}$  : é o vetor das variáveis de projeto;

$\alpha$  : é o escalar que representa o tamanho do passo, partindo de  $x_k$  até  $x_{k+1}$ ,

$\bar{d}_k$  : é a direção do vetor de busca, que geralmente, é uma direção de decréscimo de  $f(\bar{x})$ ;

$k$  : é o índice que representa a atual iteração.

A diferença básica entre os diversos algoritmos de solução de problemas de otimização, dentro da Programação Não-linear, consiste na estratégia de se determinar o vetor  $\bar{d}_k$ , correspondente às sucessivas direções de busca.

Um algoritmo geral para estes métodos da Programação Não-linear está ilustrado na **Fig. 3.2**.

**Início**

Escolher um ponto de partida  $x_0$

**Enquanto** o critério de convergência não for atendido **faça**

Determinar uma direção de busca  $\bar{d}_k$

Minimizar a função objetivo  $f(\bar{x})$  na direção  $\bar{d}_k$ , a partir do ponto  $x_k$ ,

determinando assim o tamanho do passo  $\alpha_k$

Atualizar as variáveis de projeto,  $x_k$ , através da **Equação 3.2**

**Fim do Enquanto**

**Fim**

**Fig. 3.2** – Algoritmo Geral para um Método da Programação Não-linear

Existe uma grande variedade de métodos empregáveis para a solução de problemas de Programação Matemática, em geral eles são desenvolvidos para classes específicas de problemas e são definidos pelas características da função objetivo e/ou das restrições. A seguir são abordados alguns dos principais métodos para solução de problemas com e sem restrições.

### 3.3.1. Problemas de Projeto Ótimo sem Restrições

Problemas de otimização irrestritos podem ser definidos como minimizar ou maximizar a função  $f(\bar{x})$  sem qualquer restrição em  $\bar{x}$ . Tais problemas ocorrem com pouca freqüência em aplicações práticas da engenharia [BAZARAA e SHETTY, 1979].

Entre os vários métodos existentes para solução de problemas de Programação Não-linear sem restrições, alguns requerem apenas informações sobre o valor da função, outros, além do valor da função, utilizam a primeira derivada, e há ainda os que utilizam a segunda derivada. Estes métodos são denominados métodos de ordem zero, um e dois, respectivamente. Dentre os métodos de ordem zero, também conhecidos como Métodos Diretos, encontram-se o *Pattern Search* e o de *Powell* [BAZARAA e SHETTY, 1979].

Serão apresentados a seguir métodos de primeira e segunda ordem, e um método que se situa numa ordem entre a primeira e a segunda ordem. O *Método do Gradiente* e do *Gradiente Conjugado* pertencem à primeira ordem, o *Método de Newton* é um exemplo típico de

segunda ordem e o *Método Quasi-Newton* se situa em uma ordem intermediária.

### 3.3.1.1. Método da Máxima Descida (Steepest Descent)

Este método é um dos mais antigos e é largamente aplicado para problemas de minimização de funções de várias variáveis. É freqüentemente conhecido como *Método do Gradiente* e serve como ponto de partida para métodos mais sofisticados de 1ª ordem, como por exemplo, o Método dos Gradientes Conjugados, das Métricas Variáveis, dentre outros.

**Início**

Determinar uma tolerância ( $\varepsilon$ ) e um ponto de partida  $\bar{x}$ ,

Fazer  $k \leftarrow 1$

**Se**  $\|\nabla f(\bar{x}_k)\| < \varepsilon$  **então**

Parar o processo

**Fim do Se**

**Faça**

$$\bar{d}_k = -\nabla f(\bar{x}_k)$$

Minimizar  $f(\bar{x}_k + \alpha \cdot \bar{d}_k)$

Atualizar as variáveis de projeto através da **Equação 3.2**

Fazer  $k \leftarrow k + 1$

**Até que**  $\|\nabla f(\bar{x})\| < \varepsilon$

**Fim**

**Fig. 3.3** – Algoritmo do Método do Gradiente

Partindo-se da **Equação 3.2**, a direção de busca  $\bar{d}_k$  é tomada como a direção oposta ao gradiente da função objetivo, ou seja:

$$\bar{d}_k = -\nabla f(\bar{x}_k) \quad (3.3)$$

O tamanho do passo na direção  $\bar{d}_k$  é calculado resolvendo-se um problema de minimização unidimensional, como visto anteriormente. A **Equação 3.2** é aplicada iterativamente até se atingir o mínimo, onde o gradiente da função se anula. O algoritmo representado na **Fig. 3.3** descreve os passos principais.

### 3.3.1.2. Método do Gradiente Conjugado

Antes de passar à descrição do método propriamente dito é importante apresentar o conceito de direções conjugadas. Seja  $\bar{Q}$  uma matriz definida positiva de ordem  $n$ . Dois vetores  $d_1$  e  $d_2$  são ditos conjugados ou simplesmente conjugados [BAZARAA e SHETTY, 1979] se:

$$\bar{d}_1^T \cdot \bar{Q} \cdot \bar{d}_2 = 0 \quad (3.4)$$

Estendendo-se o conceito de vetores conjugados para  $n$  vetores com direções linearmente independentes, um conjunto destes vetores  $(\bar{d}_0, \bar{d}_1, \dots, \bar{d}_{n-1})$  serão chamados conjugados se:

$$\bar{d}_i^T \cdot \bar{Q} \cdot \bar{d}_j = 0 \quad \text{para } i \neq j \quad (3.5)$$

Note que, na definição acima, a matriz  $\bar{Q}$  é uma matriz genérica. Um exemplo específico de uma matriz  $\bar{Q}$  é a matriz Hessiana  $\bar{H}$ , a qual é a derivada segunda da função objetivo em relação às variáveis de projeto e variáveis auxiliares em problemas de otimização. É comum, em aplicações práticas, assumir que a matriz  $\bar{Q}$  seja positiva definida, mas isso só ocorre se a função for convexa, ou quando se está próximo da solução.

O conceito de direções ortogonais é um caso especial do conceito de direções conjugadas, quando a matriz  $\bar{Q}$  é igual a matriz identidade, ou seja, quando  $\bar{Q} = \bar{I}$ , tem-se:

$$\bar{d}_i^T \cdot \bar{d}_j = 0 \quad \text{para } i \neq j \quad (3.6)$$

Os métodos que utilizam as direções conjugadas como direção de busca fazem-no baseando-se no fato de que uma função quadrática de  $n$  variáveis independentes pode ser minimizada, pelo menos, em  $n$  passos, desde que as buscas ocorram ao longo das direções conjugadas em relação à matriz Hessiana  $\bar{H}$  da função objetivo [OLIVEIRA, 1989].

O Método dos Gradientes Conjugados é também referido como Método das Direções Conjugadas de Fletcher e Reeves [OLIVEIRA, 1989]. Trata-se de uma pequena modificação no método do gradiente, resultando num método de convergência superior. A modificação consiste em se gerar direções conjugadas, determinadas como uma combinação linear do gradiente corrente e das direções anteriores. A princípio, pode-se pensar

que é necessário armazenar todas as direções anteriores a fim de se gerar uma nova direção que seja  $H$ -conjugada às mesmas, como visto anteriormente sobre direções conjugadas. No entanto, de acordo com SKEWCHUK (1994), a direção conjugada atual só depende da direção conjugada anterior, e como será visto mais adiante, para se gerar a direção de busca conjugada corrente basta que se armazene apenas o gradiente da função obtido no passo anterior. Em outras palavras, a direção de busca corrente é uma combinação linear da direção do gradiente atual e da direção usada no passo anterior, conforme mostra a **Equação 3.7**.

$$\bar{d}_k = -\nabla f(\bar{x}_k) + \beta_k \cdot \bar{d}_{k-1} \quad (3.7)$$

onde o parâmetro  $\beta_k$  é calculado como a razão entre o quadrado da norma do gradiente da função objetivo avaliada em  $\bar{x}_k$  e o quadrado da norma do gradiente da função objetivo avaliada em  $\bar{x}_{k-1}$ , de tal forma que o algoritmo pode ser assim estabelecido: parte-se com a direção  $\bar{d}_k = -\nabla f(\bar{x}_0)$ ; minimiza-se a função  $f(\bar{x}_k + \alpha \cdot \bar{d}_k)$  com relação a  $\alpha$ ; e atualizam-se as variáveis de projeto através da **Equação 3.2**. Para as iterações seguintes, a direção de busca conjugada é dada pela **Equação 3.7**.

Ao contrário do Método de *Powell*, onde a direção conjugada é gerada após  $n$  buscas unidimensionais, ou seja, na iteração  $n + 1$ ; neste método, cada direção de busca é uma direção conjugada, de modo que se a função objetivo for quadrática, teoricamente, o mínimo será obtido em  $n$  passos. Quando a função não é quadrática, o processo é reinicializado após  $n$  iterações, tomando-se a direção de busca contrária a do gradiente calculado no ponto  $n + 1$ , ou seja,  $\bar{d}_{n+1} = -\nabla f(\bar{x}_{n+1})$ .

### 3.3.1.3. Método de Newton

Neste método, a função a ser minimizada é aproximada por uma função quadrática, fazendo-se uso da expansão dos termos até segunda ordem da Série de Taylor. Portanto, a aproximação quadrática para a função objetivo é dada por:

$$f(\bar{x}) = f(\bar{x}_k) + \nabla f(\bar{x}_k)^T \cdot (\bar{x} - \bar{x}_k) + \frac{1}{2} \cdot (\bar{x} - \bar{x}_k)^T \cdot \bar{H}(\bar{x}_k) \cdot (\bar{x} - \bar{x}_k) \quad (3.8)$$

Aplicando-se a condição necessária de mínimo na **Equação 3.8**, obtém-se:

$$\nabla f(\bar{x}) = 0 \quad \Rightarrow \quad \nabla f(\bar{x}_k) + \bar{H}(\bar{x}_k)(\bar{x} - \bar{x}_k) = 0 \quad (3.9)$$

ou seja:

$$\bar{x}_{k+1} = \bar{x}_k - \bar{H}(\bar{x}_k)^{-1} \cdot \nabla f(\bar{x}_k) \quad (3.10)$$

onde  $\bar{H}(\bar{x}_k)^{-1}$  é a inversa da matriz Hessiana  $\bar{H}(\bar{x}_k)$  da função  $f(\bar{x})$ , definida como a matriz das segundas derivadas parciais em relação a  $\bar{x}$  e avaliadas em  $\bar{x}_k$ .

A **Equação 3.10** representa a forma recursiva dos pontos gerados pelo Método de Newton para o caso de  $n$  variáveis. Comparando-se a

**Equação 3.10** com a **Equação 3.2**, observa-se que o tamanho do passo  $\alpha_k$  é fixado em 1 e que a direção de busca fica definida por:

$$\bar{d}_k = -\bar{H}(\bar{x}_k)^{-1} \cdot \nabla f(\bar{x}_k) \quad (3.11)$$

de modo que se a função objetivo for quadrática, apenas um passo é requerido para se alcançar o mínimo da função. O valor  $\bar{H}(\bar{x}_k)^{-1}$  pode ser interpretado como uma correção na direção  $-\nabla f(\bar{x}_k)$  de forma a acelerar o processo.

Em geral, para funções não-lineares, o mínimo não será alcançado em um único passo, de tal maneira que o método é executado em um processo iterativo. Introduzindo-se o parâmetro  $\alpha$  correspondente ao comprimento do passo na **Equação 3.10**, obtém-se:

$$\bar{x}_{k+1} = \bar{x}_k - \alpha \cdot \bar{H}(\bar{x}_k)^{-1} \cdot \nabla f(\bar{x}_k) \quad (3.12)$$

onde a direção de busca é dada pela **Equação 3.11** e o comprimento do passo pode ser encontrado por qualquer um dos métodos de busca unidimensional.

O Método de Newton se comporta bem para pontos próximos à solução do problema, já que a aproximação de segunda ordem utilizada é local. A convergência quadrática do Método de Newton na vizinhança do mínimo da função objetivo só é obtida se a função puder ser bem aproximada por uma função quadrática e se forem atendidas as duas condições seguintes: primeiro, que a Hessiana  $\bar{H}(\bar{x}_k)$  seja não-singular, para que se possa obter a sua inversa; e segundo, que  $\bar{H}(\bar{x}_k)$  seja

definida positiva, a fim de se garantir que a direção  $\bar{d}_k = -\bar{H}(\bar{x}_k)^{-1} \cdot \nabla f(\bar{x}_k)$  seja uma direção de descida [AVRIL, 1976; HIMMEBLAU, 1972].

Na prática, o Método de Newton é modificado de maneira que se possa manter a matriz Hessiana definida positiva. A aproximação adotada para a matriz Hessiana é conhecida na literatura [MATEUS e LUNA, 1986; OLIVEIRA, 1989] e é expressa da seguinte forma:

$$[\bar{H}(\bar{x}_k)]^{-1} \cong [\beta_k \cdot \bar{I} + \bar{H}(\bar{x}_k)]^{-1} \quad (3.13)$$

Como pode ser visto, o procedimento consiste em se adicionar termos aos elementos da diagonal da Hessiana  $\bar{H}(\bar{x}_k)$ . A aproximação acima pode ser vista como um tipo de compromisso entre o Método do Gradiente ( $\beta_k$  muito grande, de modo que  $\beta_k \cdot \bar{I}$  domina  $\bar{H}(\bar{x}_k)$ ) e o Método de Newton ( $\beta_k = 0$ ). Existe sempre um  $\beta_k$  que torna a matriz Hessiana aproximada, positiva definida.

#### 3.3.1.4. Método Quasi-Newton

O Método Quasi-Newton pertence a uma classe de métodos onde o cálculo da inversa da matriz Hessiana, necessária no Método de Newton, é feito de maneira aproximada. Este método é visto como algo intermediário entre o Método do Gradiente e o Método de Newton. Ao invés de se calcular a inversa da Hessiana, o Método Quasi-Newton

aproxima esta inversa através de um processo iterativo finito, utilizando-se apenas derivadas de primeira ordem.

Este método é também conhecido como Método de Métrica Variável (nome dado inicialmente ao Método Quasi-Newton de *Davidon-Fletcher-Powell*). Analiticamente é considerado o mais sofisticado para problemas sem restrição.

A idéia básica do método consiste em gerar uma matriz que aproxima a inversa da matriz Hessiana à medida que o processo de otimização evolui. Neste método, a direção de busca para cada iteração é dada por:

$$\bar{d}_k = -\bar{D} \cdot \nabla f(\bar{x}_k) \quad (3.14)$$

onde  $\bar{D}$  é a matriz que aproxima a inversa da Hessiana ou a própria matriz Hessiana. Dependendo da forma utilizada para calcular a matriz  $\bar{D}$ , tem-se diferentes métodos [BAZARAA e SHETTY, 1979; HIMMEBLAU, 1972], tais como: Método de Broyden, Método de *Davidon-Fletcher-Powell* e o Método BFGS (devido a *Broyden-Fletcher-Goldfarb-Shanno*).

### 3.3.2. Problemas de Projeto Ótimo com Restrições

O problema de Programação Matemática de solução mais complexa é o que envolve funções e restrições lineares e/ou não-lineares. Este

grau de complexidade é refletido na diversidade de métodos de solução existentes que, de uma forma geral, podem ser englobados em uma das três categorias de técnicas ou combinação destas: técnicas de transformação, técnicas diretas, técnicas recursivas. Essas classificações são feitas de acordo com os artifícios usados por cada uma das estratégias para a solução de um dado problema:

A *Técnica de Transformação* ou Minimização Seqüencial Irrestrita, historicamente, foi a primeira a ser tentada e é a mais difundida no campo da otimização por ser de implementação computacional mais simples; consiste na solução de uma seqüência de problemas de minimização irrestrita, obtidos pela aplicação de uma função de transformação ao problema original, restrito. Em outras palavras transforma um problema com restrições em um problema sem restrições: as funções de penalidades, de barreira e o lagrangeano são as normalmente empregadas para esta transformação [BAZARAA e SHETTY, 1979].

A *Técnica Direta* engloba os métodos que atacam diretamente o problema restrito, utilizando uma seqüência de minimizações unidimensionais, ao longo de direções viáveis. O Método das Direções Viáveis, do Gradiente Reduzido e do Gradiente Projetado são exemplos típicos desta técnica [LUENBERGER, 1984; BAZARAA e SHETTY, 1979; HAFTKA e GÜRDAL, 1992].

A *Técnica Recursiva* refere-se aos métodos que utilizam uma seqüência de problemas restritos de solução bem mais simples e acessível, como um recurso para o problema original. São semelhantes aos métodos diretos, no sentido que preservam o problema original, resolvendo o problema através de aproximações nas condições de Otimalidade [BAZARRA e SHETTY, 1979; HAFTKA e GÜRDAL, 1992].

### 3.3.2.1. Método das Penalidades

No Método das Penalidades, a solução do problema restrito (**Equação 3.1**, considerando agora pelo menos uma função não-linear) é obtida através da solução de uma seqüência ( $k = 1, 2, \dots$ ) de problemas de minimização irrestrita:

$$\min \phi^k(\bar{x}, r^k) \quad \text{com} \quad r^k > 0 \quad (3.15)$$

envolvendo uma função de penalidade  $\phi^k$  e um parâmetro de controle  $r^k$ , sempre positivo [OLIVEIRA, 1989].

A função de penalidade é construída de forma a prescrever um alto custo à violação das restrições no problema original. A função objetivo transformada fica:

$$\phi^k(\bar{x}, r^k) = f(\bar{x}) + r^k P(\bar{x}) \quad \text{com} \quad r^k > 0 \quad (3.16)$$

O termo de penalidade  $P(\bar{x})$  é definido como sendo sempre não-negativo; sendo nulo, somente se  $\bar{x}$  estiver na região viável. O parâmetro  $r^k > 0$  é usado para justificar a magnitude da penalidade. Em outras palavras, a violação das restrições é gradualmente ponderada: a medida que  $r^k \rightarrow \infty$ , para que  $\phi^k(\bar{x}, r^k)$  seja minimizada, é preciso que  $P^k(\bar{x}) \rightarrow 0$ , implicando em:

$$\phi^k(\bar{x}, r^k) \rightarrow f(\bar{x}^k) \quad \text{e} \quad \bar{x}^k \rightarrow \bar{x}^* \quad (3.17)$$

Existem várias construções do termo de penalidade, todas com a mesma finalidade: penalizar a violação das restrições. Segundo HAFTKA e GÜRGAL (1992), a função de penalidade mais comumente usada é a que torna a penalidade proporcional ao quadrado da violação.

O interesse está em se determinar a forma mais simples de se penalizar a violação, que é dada pela função de perda de Zangwill (ou penalidade exata) [HIMMEBLAU, 1972]:

$$P_q(\bar{x}) = \begin{cases} \max(0, g_q(\bar{x})) & \text{se } 1 \leq q \leq l \\ |h_q(\bar{x})| & \text{se } l+1 \leq q \leq m \end{cases} \quad (3.18)$$

Uma observação final é que a seqüência de soluções  $\bar{x}^k$  se aproxima da solução  $\bar{x}^*$ , por fora da região viável; sendo, por isto a técnica também conhecida como de *Minimização Irrestrita pelo Ponto Exterior*.

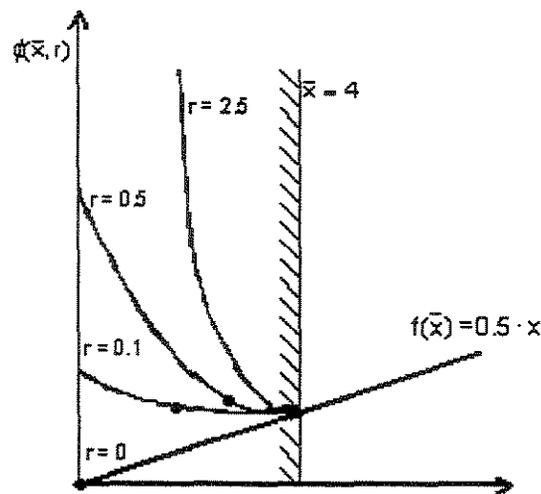


Fig. 3.4 – Ilustração do Método das Penalidades (fonte: HAFTKA e GÜRDAL, 1992)

Note que na solução do problema não-linear pelo Método das Penalidades, ilustrada na Fig. 3.4, os pontos de mínimo da função  $\phi^k(\bar{x}, r^k)$ , para os mais diversos valores de  $r^k$ , estão sempre na região infactível, mas a medida que o valor de  $r^k$  aumenta estes pontos se aproximam da fronteira.

### 3.3.2.2. Método do Lagrangeano Aumentado

Antes da apresentação deste método é interessante mostrar o conceito dos Multiplicadores de Lagrange e os seus significados, já que o Método do Lagrangeano Aumentado é uma extensão deste conceito; devido a este fato, o método é também conhecido como Método dos Multiplicadores [OLIVEIRA, 1989].

Considere o problema de otimização abaixo:

$$\begin{array}{ll} \text{Minimize} & f(\bar{x}) \\ \text{Sujeito a} & h(\bar{x}) = 0 \end{array} \quad (3.19)$$

Uma maneira de se resolver este problema seria explicitar as variáveis nas restrições de igualdade ( $h(\bar{x})=0$ ) e substituí-las na função  $f(\bar{x})$  de tal maneira que o problema com restrições de igualdade seria transformado em um problema sem restrições. Muitas vezes não é possível (ou conveniente) obter uma solução explícita para a equação  $h(\bar{x})=0$ . Uma outra maneira, também baseada na idéia de transformar um problema com restrições em outro sem restrições, é usar os multiplicadores de Lagrange.

O Lagrangeano do problema é definido como:

$$L(\bar{x}, \lambda) = f(\bar{x}) + \lambda \cdot h(\bar{x}) \quad (3.20)$$

onde  $\lambda$  são os multiplicadores de Lagrange.

As condições necessárias (porém, não suficientes) para se determinar o mínimo de  $f(\bar{x})$  com restrições são:

$$\begin{cases} \frac{\partial L(\bar{x}, \lambda)}{\partial \bar{x}} = 0 \\ \frac{\partial L(\bar{x}, \lambda)}{\partial \lambda} = 0 \end{cases} \Rightarrow h(\bar{x}) = 0 \quad (3.21)$$

Isto gera um sistema de equações através do qual se encontram os valores de  $\bar{x}$  e  $\lambda$ . O ponto encontrado deve ser verificado através das condições suficientes [BAZARAA e SHETTY, 1979].

O problema de Programação Não-linear geral pode ser transformado em um problema que contém somente restrições de igualdade. Por exemplo, seja o seguinte problema:

$$\begin{aligned} & \text{Minimize} && f(\bar{x}) \\ & \text{Sujeito a} && h(\bar{x}) = 0 \\ & && g(\bar{x}) \leq 0 \end{aligned} \quad (3.22)$$

As restrições de desigualdade são convertidas em restrições de igualdade, com a introdução de variáveis de folga não-negativas (ao quadrado)  $u^2$ , nas restrições  $g(\bar{x}) \leq 0$ . O uso do quadrado da variável de folga, garantindo um valor positivo para qualquer  $u$ , evita adicionar restrições de desigualdade do tipo  $u \geq 0$ , como é feito na Programação

Linear. Com isso, o problema representado pela **Equação 3.22** pode ser reescrito como:

$$\begin{aligned}
 & \text{Minimize} && f(\bar{x}) \\
 & \text{Sujeito a} && h(\bar{x}) = 0 \quad i = 1, \dots, m \\
 & && g(\bar{x}) + u^2 = 0 \quad j = m+1, \dots, p
 \end{aligned} \tag{3.23}$$

de modo que o lagrangeano fica:

$$L(\bar{x}, \lambda, u) = f(\bar{x}) + \sum_{i=1}^m \lambda_i h_i(\bar{x}) + \sum_{j=m+1}^p \lambda_j (g_j(\bar{x}) + u_j^2) \tag{3.24}$$

As condições necessárias para se encontrar um ponto estacionário são:

$$\frac{\partial L(\bar{x}, \lambda, u)}{\partial \bar{x}} = 0 \quad \frac{\partial L(\bar{x}, \lambda, u)}{\partial \lambda_i} = 0 \quad \frac{\partial L(\bar{x}, \lambda, u)}{\partial u_j} = 0 \tag{3.25}$$

As condições necessárias que o mínimo de  $f(\bar{x})$  devem satisfazer são:

- i.  $\frac{\partial L(\bar{x}, \lambda, u)}{\partial \bar{x}} = 0 \Rightarrow \frac{\partial f}{\partial \bar{x}} + \sum_{i=1}^m \lambda_i \frac{\partial h_i(\bar{x})}{\partial \bar{x}} + \sum_{j=m+1}^p \lambda_j \frac{\partial g_j(\bar{x})}{\partial \bar{x}} = 0;$
- ii.  $\bar{x}$  seja viável, ou seja, que atenda a todas as restrições;
- iii.  $\lambda_j \cdot g_j(\bar{x}) = 0 \quad j = m+1, \dots, p;$
- iv.  $\lambda_j \geq 0 \quad j = m+1, \dots, p,$  ou seja, os multiplicadores de Lagrange (para as restrições de desigualdade) sejam não-negativos;
- v.  $\lambda_i \quad i = 1, \dots, m$  (para as restrições de igualdade) não tenham restrições de sinal.

As condições acima são conhecidas como Condições de *Kuhn-Tucker* [BAZARAA e SHETTY, 1979; AVRIEL, 1976; HIMMEBLAU, 1972].

Em vista do exposto, pode-se dizer que o Método dos Multiplicadores de Lagrange consiste em transformar um problema de extremização de uma função submetida a restrições de igualdade em um outro, no qual se determina o ponto estacionário do Lagrangeano [MATEUS e LUNA, 1986].

Voltando-se ao Método do Lagrangeano Aumentado, o mesmo surgiu para superar algumas dificuldades encontradas nos métodos de penalidades e de barreiras, como a descontinuidade sobre a fronteira da região viável e o não condicionamento da matriz Hessiana para pontos próximos à fronteira. Busca-se assim, definir funções que tenham a boa característica da penalidade exata (não exigir parâmetros tendendo ao infinito), sendo também diferenciáveis.

O lagrangeano aumentado é obtido adicionando-se uma função de penalidade ao lagrangeano usual.

Considerando-se restrições de igualdade, como na **Equação 3.19**, o lagrangeano aumentado é dado por:

$$A(\bar{x}, \lambda, \mu) = f(\bar{x}) + \sum_{i=1}^m \lambda_i h_i(\bar{x}) + r \cdot \sum_{i=1}^m h_i^2(\bar{x}) \quad (3.26)$$

onde  $r \cdot \sum_{i=1}^m h_i^2(\bar{x})$  é o termo correspondente à penalidade. E o lagrangeano usual do problema, de acordo com a **Equação 3.26**, é dado por:

$$L(\bar{x}, \lambda) = f(\bar{x}) + \sum_{i=1}^m \lambda_i h_i(\bar{x}) \quad (3.27)$$

A principal motivação para a inclusão do lagrangeano ao método é reduzir a dependência do algoritmo na escolha do parâmetro de penalidade e a maneira pela qual eles são atualizados durante o processo de otimização.

Com restrições de desigualdade, tem-se:

$$\begin{aligned} & \text{Minimize} && f(\bar{x}) \\ & \text{Sujeito a} && g_j(\bar{x}) \leq 0 \quad j = m+1, \dots, p \end{aligned} \quad (3.28)$$

Como visto no Método dos Multiplicadores de Lagrange, transforma-se as restrições de desigualdade em restrições de igualdade, fazendo-se uso das variáveis de folga, ou seja:

$$g_j(x) + u_j^2 = 0 \quad (3.29)$$

O lagrangeano aumentado é dado por:

$$A(\bar{x}, \lambda, \mu) = f(\bar{x}) + \sum_{j=m+1}^p \lambda_j (g_j(\bar{x}) + u_j^2) + r \cdot \sum_{j=m+1}^p (g_j(\bar{x}) + u_j^2)^2 \quad (3.30)$$

No caso geral, com restrições de ambos os tipos, o Lagrangeano aumentado é dado por:

$$A(\bar{x}, \lambda, u, r) = f(\bar{x}) + T_l + \sum_{i=1}^m [\lambda_i h_i(\bar{x}) + r \cdot (h_i(\bar{x}))^2] \quad (3.31)$$

BAZARAA e SHETTY (1979) apresentam uma descrição dos algoritmos para as restrições separadas e para o caso geral. Segundo estes autores, o método tem vários aspectos atrativos, tais como:

- i.* Relativa insensibilidade ao valor do parâmetro  $r$ ; não sendo necessário fazer  $r \rightarrow \infty$ ;
- ii.* Possibilidade de se obter  $g_j(\bar{x})=0$  e  $h_i(\bar{x})=0$ ;
- iii.* Acelera-se o método atualizando os multiplicadores de Lagrange;
- iv.* O ponto de partida pode ser viável ou não viável;
- v.* Na solução ótima, os valores de  $\lambda_j^* \neq 0$  automaticamente identificam o conjunto das restrições ativas.

## **4. Otimizadores do GAMS**

Como já fora enfatizado, o processo de otimização proposto neste trabalho consiste em modelar treliças planas na linguagem *GAMS* (*General Algebraic Modelling System*) e submeter esses modelos a duas formas diferentes de otimização, no que diz respeito às características do espaço solução. Primeiramente o modelo será resolvido com o uso de um otimizador contínuo e não-linear, o *MINOS*; posteriormente os resultados do primeiro servirão para compor a base de dados para o modelo discreto e também não-linear, que será resolvido pelo otimizador *DICOPT*.

Assim, será apresentado a seguir esse dois otimizadores do *GAMS*, onde será visto um breve histórico do seu desenvolvimento, como funciona o seu algoritmo e como trabalham em parceria com o *GAMS*.

## 4.1. *MINOS*

A maioria dos modelos criados com *GAMS* podem ser resolvidos com uma versão especial adaptada do otimizador *MINOS* (versão 5.1), desenvolvido por MURTAGH e SAUNDERS, em 1987. Antes de se buscar uma solução para um modelo, precisa-se saber se o mesmo é linear ou não-linear, embora idealmente não seja necessário saber como este será resolvido, desde que a solução seja encontrada.

Na prática nem todos os modelos serão solucionados com sucesso, exigindo do programador uma adequada interpretação dos dados de saída para que se possa especificar parâmetros que controlem o otimizador *MINOS*, influenciando assim o processo, para torná-lo mais eficaz.

### 4.1.1. Descrição Geral do Otimizador *MINOS*

O *MINOS* foi projetado e adaptado para resolver problemas de otimização de grande porte com as seguintes características:

$$\begin{aligned} \text{Minimize} \quad & F(x) + c^T x + d^T y \\ & x, y \end{aligned} \tag{4.1}$$

$$\text{Sujeito a} \quad f(x) + A_1 y \sim b_1 \tag{4.2}$$

$$A_2 x + A_3 y \sim b_2 \tag{4.3}$$

$$l \leq \begin{pmatrix} x \\ y \end{pmatrix} \leq u \tag{4.4}$$

onde os vetores  $c$ ,  $d$ ,  $b_1$ ,  $b_2$ ,  $l$  e  $u$ , bem como as matrizes  $A_1$ ,  $A_2$  e  $A_3$  são constantes,  $F(x)$  é uma função escalar lisa e  $f(x)$  é um vetor de funções lisas.

As componentes de  $x$  são chamadas variáveis não-lineares e as componentes de  $y$  são as variáveis lineares. Analogamente, as **Equações 4.2** são chamadas restrições não-lineares e as **Equações 4.3** são restrições lineares. Em particular, os vetores  $l$  e  $u$ , nas **Equações 4.4** são respectivamente representantes dos limites inferiores e superiores dados às variáveis  $x$  e  $y$ . O sinal  $\sim$  nas **Equações 4.2** e **4.3** significam que as restrições individuais podem ser  $\leq$ ,  $=$  ou  $\geq$ . Conjuntamente, as **Equações 4.2** e **4.3** são ditas restrições gerais.

Suponha que  $m_1$  e  $n_1$  denotem respectivamente o número de restrições e de variáveis não-lineares, e que  $m$  e  $n$  representem o número total de restrições (gerais) e de variáveis. Desta forma,  $A_3$  possui  $m - m_1$  e  $n - n_1$  colunas.

#### 4.1.1.1. Programação Linear

Se as Funções  $F(x)$  e  $f(x)$  (das **Equações 4.1** e **4.2**, respectivamente) encontram-se ausentes do problema, o programa torna-se um programa linear, conseqüentemente não é necessário fazer distinção entre variáveis lineares ou não-lineares. *MINOS*, então, atribui a  $x$  o conjunto das variáveis, converte todas as restrições em igualdades, e as desigualdades que tiverem serão os limites simples sobre as variáveis. Assim, o problema de otimização descrito pelas **Equações 4.1** a **4.4**, serão simplificados para:

$$\begin{array}{ll} \text{Minimize} & c^T x \\ & x \end{array} \quad (4.5)$$

$$\text{Sujeito a } Ax + Is = 0 \quad (4.6)$$

$$l \leq \begin{pmatrix} x \\ s \end{pmatrix} \leq u \quad (4.7)$$

onde  $s$  é o conjunto de variáveis de folga, sendo uma para cada restrição. Por razões computacionais, o vetor  $b$  é incorporado nos limites do vetor  $s$ . Na **Equação 4.6**, a matriz identidade  $I$  é escrita se estivermos lidando com as colunas da matriz associada  $A$ ; caso contrário, basta usar a notação  $Ax + Is = 0$ .

O otimizador *MINOS* resolve programas lineares usando a implementação do Método Simplex (PUCCINI, 1978) (veja item 3.2).

#### 4.1.1.2. Problemas com a Função Objetivo Não-linear

Quando as não-linearidades se encontram limitadas ao termo  $F(x)$ , na Função Objetivo (**Equação 4.1**), o problema é de Programação Não-linear com restrições lineares. *MINOS* resolve tais problemas usando um algoritmo do Método do Gradiente Reduzido combinado com um algoritmo do Método Quasi-Newton que geralmente conduz à convergência superlinear.

#### 4.1.1.3. Problemas com Restrições Não-lineares

Se algumas das restrições são não-lineares, independentemente de  $F(x)$ , *MINOS* utiliza o algoritmo do Lagrangeano Aumentado, baseado no método de ROBINSON (1972; em BROOKE et al., 1992). Isso envolve uma seqüência de iterações bem maiores, cada uma das quais exigindo a solução de um subproblema linearmente restrito. Cada subproblema contém versões linearizadas das restrições não-lineares, bem como as restrições lineares originais e os limites.

Na  $k$ -ésima iteração, seja  $x_k$  uma estimativa das variáveis não-lineares, e seja  $\lambda_k$  uma estimativa dos multiplicadores de Lagrange

associados às restrições não-lineares. As restrições são linearizadas através da substituição de  $f(x)$  na **Equação 4.2** para a sua aproximação linear:

$$\tilde{f}(x, x_k) = f(x_k) + J(x_k)(x - x_k)$$

onde  $J(x_k)$  é a matriz Jacobiana, ou seja, o vetor gradiente da  $i$ -ésima função de restrição linear, calculada em  $x_k$ .

O subproblema a ser resolvido durante a  $k$ -ésima iteração é então:

$$\begin{array}{ll} \text{Minimize} & F(x) + c^T x + d^T y - \lambda_k^T (f - \tilde{f}) + \frac{1}{2} \rho (f - \tilde{f})^T (f - \tilde{f}) \\ x, y & \\ \text{Sujeito a} & f(x) + A_1 y \sim b_1 \\ & A_2 x + A_3 y \sim b_2 \\ & l \leq \begin{pmatrix} x \\ y \end{pmatrix} \leq u \end{array}$$

onde a Função Objetivo é chamada Lagrangeana aumentada, o escalar  $\rho$  é um parâmetro de penalização, e o termo envolvendo  $\rho$  é a função de penalização quadrática. *MINOS* utiliza o algoritmo do gradiente reduzido para minimizar este problema.

## 4.2. *DICOPT*

É um programa para resolver problemas de Programação Não-linear Inteira-mista (PNLIM), que são aqueles que envolvem variáveis inteiras e/ou binárias em conjunto com variáveis contínuas lineares ou não-lineares. Embora a modelagem e solução desses problemas de otimização ainda não tenha obtido maturidade e confiança suficiente como já alcançaram as modelagem por programação linear, não-linear e inteira, este tipo de problema tem uma vasta área de aplicação, principalmente nos projetos de engenharia, que requerem descrições nas suas variáveis em função de disponibilidades práticas e comerciais.

*DICOPT*, ou seja, *Discrete and Continuous OPTimizer*, foi desenvolvido por VISWANATHAN e GROSSMANN (1990), no *Engineering Design Research Center*, da Universidade Carnegie Mellon. O programa é baseado nas extensões de um algoritmo de aproximação externa para uma estratégia de igualdade relaxada. O algoritmo de Programação Não-linear Inteira-mista (PNLIM) dentro do *DICOPT* resolve uma série de subproblemas de Programação Não-linear (PNL) e Programação Inteira-mista (PIM). Estes subproblemas podem ser resolvidos usando qualquer *solver* de PNL e PIM que funcione dentro do *GAMS*. Assim, como o *DICOPT* apenas gera os subproblemas, o *GAMS* pode habilitar em sua configuração otimizadores de PNL (por exemplo, *MINOS*, *CONOPT*, etc.) e de PIM (por exemplo, *CPLEX*, *OSL*, entre outros).

Apesar do algoritmo ter condições de lidar com funções não-convexas, não obtém necessariamente o ótimo global.

O sistema *GAMS-DICOPT* foi projetado com dois objetivos em mente: tornar fácil e compatível a modelagem e resolução de problemas com variáveis discretas e/ou inteiras, sem majorar o trabalho do usuário com outros comandos e extensões; permitir que se use outros

otimizadores existentes no GAMS para resolver os subproblemas gerados pelo *DICOPT*.

#### 4.1.1. Descrição Geral do Otimizador *DICOPT*

O *DICOPT* resolve modelos da seguinte forma:

$$\begin{array}{ll} \text{Minimize} & f(x,y) \\ & x,y \end{array}$$

$$\begin{array}{ll} \text{Sujeito a} & g(x,y) \sim b \\ & l_x \leq x \leq u_x \\ & y \in [l_y, \dots, u_y] \end{array}$$

onde  $x$  é o vetor das variáveis contínuas e  $y$  é o vetor das variáveis discretas; o símbolo  $\sim$  significam que os operadores relacionais das restrições individuais podem ser  $\leq$ ,  $=$  ou  $\geq$ . Estas restrições podem ser linear ou não-linear. Os valores de contorno  $l$  e  $u$  nas variáveis são manipulados diretamente pelo usuário.  $\lceil x \rceil$  indica o menor inteiro, maior que ou igual a  $x$ . Analogamente,  $\lfloor x \rfloor$  indica o maior inteiro, menor que ou igual a  $x$ . As variáveis discretas podem ser inteiras ou binárias.

O algoritmo em *DICOPT* é baseado em três idéias fundamentais: aproximação externa, relaxamento das igualdades e penalidade aumentada.

A *Aproximação externa* refere-se ao fato de que a superfície descrita pela função convexa fica acima da tangente do hiperplano de qualquer ponto interior da superfície. No algoritmo, aproximações externas são realizadas pelas linearizações geradas a cada iteração e acumulando-as em ordem para garantir sucessivamente melhores aproximações lineares de funções não-lineares convexas que para subestimar a função objetivo e superestimar a região factível.

O Relaxamento das igualdades é baseado na seguinte suposição da Programação Não-linear. Suponha que o problema de PNLIM é formulado da seguinte maneira:

$$\begin{array}{ll}
 \text{Minimize} & f(x) + c^T y \\
 x & \\
 & G(x) + H y \sim b \\
 \text{Sujeito a} & l \leq x \leq u \\
 & y \in \{0,1\}
 \end{array}$$

onde as variáveis discretas  $y$  são variáveis binárias e aparecem linearmente no modelo.

Se o modelo acima tiver suas equações reordenadas em equações de igualdade e desigualdade, pode-se escrevê-lo na forma:

$$\begin{array}{ll}
 \text{Minimize} & c^T y + f(x) \\
 x & \\
 & A y + h(x) = 0 \\
 \text{Sujeito a} & B y + g(x) \leq 0 \\
 & l \leq x \leq u \\
 & y \in \{0,1\}
 \end{array}$$

Considerando que  $y^{(0)}$  seja qualquer vetor binário fixo e  $x^{(0)}$  a solução do subproblema de PNL:

$$\begin{array}{ll}
 \text{Minimize} & c^T y^{(0)} + f(x) \\
 x & \\
 & A y^{(0)} + h(x) = 0 \\
 \text{Sujeito a} & B y^{(0)} + g(x) \leq 0 \\
 & l \leq x \leq u \\
 & y \in \{0,1\}
 \end{array}$$

Além disso, consideremos também:

$$\begin{array}{l}
 T^{(0)} = \text{diagonal}(t_{i,i}) \\
 t_{i,i} = \text{senal}(\lambda_i)
 \end{array}$$

onde  $\lambda_i$  é o multiplicador de Lagrange da  $i$ -ésima restrição de igualdade.

Se  $f(x)$  for uma função pseudo-convexa,  $h(x)$  e  $g(x)$  forem funções quase-convexas, então  $x^0$  também é uma solução do seguinte modelo de PNL:

$$\begin{array}{ll}
 \text{Minimize} & c^T y^{(0)} + f(x) \\
 x & \\
 & T^{(0)}(A y^{(0)} + h(x)) \leq 0 \\
 \text{Sujeito a} & B y^{(0)} + g(x) \leq 0 \\
 & l \leq x \leq u
 \end{array}$$

Em termos gerais, sob algumas hipóteses a respeito da convexidade de funções não-lineares, uma restrição de igualdade pode ser relaxada para ser uma restrição de desigualdade. Essa propriedade é

utilizada no problema-mestre de PIM para acumular aproximações lineares.

A Penalidade Aumentada refere-se a introdução de variáveis de folga não-negativa do lado direito das restrições de desigualdade e a modificação da função objetivo quando hipóteses a respeito de convexidade não são válidas.

O algoritmo *DICOPT* abaixo começa pela resolução do problema não-linear, no qual as condições 0 ou 1 nas variáveis binárias são relaxadas. Se a solução deste problema resultar numa solução inteira, a busca pára; caso contrário, continua com uma sequência alternada de programas não-lineares, chamados de subproblemas, e programas lineares inteiro-misto, chamados de problemas-mestre. Os subproblemas são resolvidos pelas variáveis fixas 0-1 que são prognosticadas pelo problema-mestre a cada iteração. Para o caso de problemas convexos, o problema-mestre também fornece um limite inferior para a função objetivo. Esse limite, exclusivo para o caso de minimização, aumenta uniformemente tanto quanto as iterações ocorrem, devido ao acúmulo de aproximações lineares. Nota-se que no caso de maximização, este limite é um limite superior. Este limite pode ser usado como um critério de parada para o *DICOPT*. Outro critério de parada que tende a funcionar muito bem na prática, para problemas convexos e não-convexos, é baseado na heurística: pára assim que os subproblemas começam a piorar, isto é, o subproblema corrente gera uma função objetiva ótima que é pior que a gerada no subproblema anterior. Este critério de parada confia no uso da penalidade aumentada e é usado na descrição do algoritmo abaixo. Este é também o critério de parada padrão na implementação do *DICOPT*. O algoritmo segue os passos:

- 1) Resolver o problema não-linear relaxado do programa PNLIM.

Se  $y^{(0)} = y$  é inteira, a busca pára. Senão, continua a partir do passo 2.

- 2) Procurar um ponto inteiro  $y^{(1)}$  com um problema-mestre de PIM que caracterize uma função de penalidade aumentada para achar o mínimo sobre a superfície convexa determinada pelo semi-espço na solução  $(x^{(0)}, y^{(0)})$ .
- 3) Fixar as variáveis binárias  $y = y^{(1)}$  e com elas, obter o resultado do problema não-linear. Deixar que  $(x^{(1)}, y^{(1)})$  seja a solução correspondente.
- 4) Encontrar a solução inteira  $y^{(2)}$  com um problema-mestre que corresponde a minimização sobre a interseção da superfície convexa descrita pelos semi-espços em  $y^{(0)}$  e  $y^{(1)}$ .
- 5) Repetir os passos 3 e 4 até que haja um aumento no valor da função objetivo; em particular, repetir o passo 4 significa aumentar o conjunto sobre o qual a minimização é realizada com linearizações adicionais.

### 4.3. Limitações

Infelizmente, não existe nenhuma garantia que os algoritmos descritos irão sempre convergir partindo de um ponto inicial arbitrário ou apresentar uma solução ótima, principalmente com referência aos algoritmos de Programação Não-linear Mista-inteira, que envolve variáveis

binárias. O programador poderá e deverá, na maioria das vezes, influir sobre a probabilidade de convergência especificando os níveis de atividade inicial das variáveis o mais cuidadosamente possível, incluindo limites superiores e inferiores para as mesmas, de forma a diminuir o delta entre os valores iniciais e os que se aproximem dos valores tidos como ótimos [BROOKE et al., 1997].

O conjunto *GAMS / MINOS* foi projetado para encontrar soluções que sejam localmente ótimas. Uma certa região fica definida pelas restrições de um problema e pelos limites das variáveis. Caso a função objetivo e as funções de restrição sejam convexas nessa região, a solução ótima obtida será necessariamente o ótimo global.

O conjunto *GAMS / DICOPT* tem sua solução ótima vinculada a pontos discretos dentro do espaço solução fornecido pelo usuário. Da mesma forma, se a função objetivo for convexa nesta região, retornará a solução ótima global; caso contrário, se a função objetivo for não-convexa e factível no espaço solução, o otimizador *DICOPT* poderá vir a relaxar algumas variáveis binárias envolvidas, ou seja, permitir valores reais destas variáveis compreendido entre 0 e 1, a fim de que possa retornar uma solução ótima, entretanto acusando infactibilidade nas variáveis binárias violadas.

Quanto ao seu custo operacional, o *GAMS* não representa nenhum problema. Por exemplo, um problema com 500 variáveis e 100 restrições não-lineares resolvidos pelo otimizador *MINOS* ocupa aproximadamente 1 *Mbyte* de memória RAM e um processador *Pentium* da primeira geração executa-o em menos de 30 segundos.

## **5. Descrição do Programa e Modelos Gerados**

Para a realização da otimização de peso de treliças planas, objeto principal da Dissertação de Mestrado, utilizou-se o software *GAMS*, que é um programa que interpreta um arquivo escrito em uma linguagem própria de comandos e é capaz de resolver um problema de otimização linear ou não-linear, através do cálculo de uma função objetivo sujeita a restrições lineares ou não-lineares. Esta interpretação é necessária para que os otimizadores trabalhem realizando a parte matemática do processo de otimização.

Na dissertação de mestrado, foi elaborado um programa de computador, escrito em linguagem Pascal, denominado *pTRUSS*, o qual escreve de forma sistemática o arquivo de dados com os comandos da linguagem *GAMS*. O programa *pTRUSS* cria um arquivo de dados com extensão *.GMS*, que faz então a otimização esperada.

Como forma de organização das análises possíveis, o problema não-linear de otimização do peso das treliças foi dividido no trabalho em dois tipos:

- Análise Contínua – tipo de análise onde a área escolhida para as barras pode ter qualquer valor no campo dos números reais positivos.
- Análise Discreta – tipo de análise onde o software de otimização deve escolher um perfil dentre um conjunto discreto fornecido pelo usuário.

O programa pTRUSS monta o conjunto de restrições obedecendo os requisitos da norma de tensões admissíveis AISC-ASD/1989. No trabalho foi admitido que as barras são tubulares. Assim as equações de restrição geradas automaticamente no programa pTRUSS referem-se a este particular tipo de perfil.

Resumindo, o programa pTRUSS pode preparar um arquivo GAMS para a análise contínua ou discreta. No último caso o usuário deve incluir uma tabela de perfis disponíveis fornecendo para cada um deles o diâmetro externo e a espessura da parede do tubo.

Então, este arquivo “.GMS” contém comandos para a minimização da função peso da estrutura, restringida pelas equações de equilíbrio representada pelo sistema  $K \cdot U = F$  do Método dos Deslocamentos, pelas tensões admissíveis de tração e compressão, por áreas e deslocamentos mínimos e máximos. Nos dois tipos de análises o usuário pode ainda optar para que várias barras tenham a mesma área. É o que foi chamado na dissertação de arranjo de barras por grupos, onde as barras de cada grupo terão a mesma área. Neste caso, outra restrição ao problema de otimização se faz presente: variáveis binárias são usadas para escolher um ou outro perfil dentro do grupo de barras.

Uma vez preparado o arquivo de comandos pelo programa pTRUSS, com extensão .GMS, executa-se a análise com o software

GAMS, obtendo o peso ótimo e as áreas das barras ou grupos de barras. O programa, para fins de controle, mostra no pós-processamento a tensão efetiva nas barras e os fatores de solicitação das mesmas, obtidos da razão entre a tensão efetiva e a tensão admissível.

### 5.1. Programa pTRUSS

O programa utiliza como entrada de dados um arquivo com todas as características da estrutura, conforme mostra o exemplo apresentado na **Fig. 5.1**. Este arquivo é definido pelo usuário, ou seja, a execução do programa começa com a digitação do nome do arquivo de entrada e o nome do arquivo de saída, necessariamente com a extensão do *.GMS*.

Após definidos os arquivos de entrada e saída, o programa procede a leitura dos dados. No bloco 1, lê-se o número de nós, o número de barras, o número de nós restringidos, o número de nós carregados, o Módulo de Elasticidade Longitudinal (em *ksi*), a densidade do material (podendo ser qualquer unidade, desde que coincida a unidade de área com polegadas, *in*) e a tensão de escoamento do material (em *ksi*).

No segundo bloco deve ser fornecida a primeira restrição estrutural. São indicados os valores dos deslocamentos nodais mínimo e máximo. Caso não se queira impor esse tipo de restrição, basta colocar um valor suficientemente grande de forma a não influir nos resultados da otimização.

No terceiro bloco, é fornecido um parâmetro que indica o tipo de análise: escolhido entre 0 para análise contínua e 1 para análise discreta; juntamente com o número de grupos de barras. Se a opção pelo tipo de análise for 1, na mesma linha deve ser fornecido ainda o Índice de Esbeltez máximo a que as barras estarão sujeitas, normalmente igual a 200 na maioria das Normas de Cálculo.

```

1      6   10   2   4   1.0E4   0.1   41.66666667
2      -2.0   2.0
3      0   10
4      0.1   30.0   15.0
5      1      0.0   360.0
      2   360.0   360.0
      3   720.0   360.0
      4      0.0      0.0
      5   360.0      0.0
      6   720.0      0.0
6      1   1   1
      4   1   1
7      2   0.0      50.0
      3   0.0      50.0
      5   0.0   -150.0
      6   0.0   -150.0
8      1   1   2   1
      2   4   5   2
      3   2   3   3
      4   5   6   4
      5   1   5   5
      6   4   2   6
      7   2   6   7
      8   5   3   8
      9   2   5   9
     10   3   6  10

```

**Fig. 5.1** – Exemplo de arquivo de entrada de dados (a numeração a esquerda não faz parte, serve apenas para numerar o blocos).

No quarto bloco, se o tipo de análise for igual a 0 (análise contínua), são fornecidos na seqüência a área mínima, a área máxima e a área inicial, todas em polegadas quadradas. Caso seja optado por fazer a

análise discreta, o bloco trará subdivisões onde, na mesma linha, se entrará com o número do grupo e a quantidade de perfis disponíveis para aquele grupo, conforme mostra a **Fig 5.2**. Nesse caso, nas próximas linhas, entrar-se-á com os diâmetros e espessuras de cada perfil tubular, em polegadas. Este conjunto se repetirá então, de acordo com o número de grupos de barras da estrutura.

|        |   |       |
|--------|---|-------|
| 1      | 4 |       |
| 8.626  |   | 0.252 |
| 8.626  |   | 0.276 |
| 8.626  |   | 0.311 |
| 8.626  |   | 0.323 |
| 2      | 4 |       |
| 12.748 |   | 0.331 |
| 12.748 |   | 0.343 |
| 12.748 |   | 0.374 |
| 12.748 |   | 0.406 |
| 3      | 3 |       |
| 8.626  |   | 0.276 |
| 8.626  |   | 0.311 |
| 10.748 |   | 0.252 |
| 4      | 5 |       |
| 12.748 |   | 0.280 |
| 12.748 |   | 0.311 |
| 12.748 |   | 0.311 |
| 12.748 |   | 0.331 |
| 12.748 |   | 0.343 |

**Fig. 5.2** – Exemplo alternativo para o bloco 4 de entrada de dados.

O próximo bloco de dados, indicado na **Fig. 5.1**, traz as coordenadas dos nós. Cada linha trará, na seqüência, o número do nó, a coordenada na direção x e a coordenada na direção y, ambas expressas em polegadas.

A lista de restrições nodais é incluída no 6º bloco de dados. Em cada linha deste bloco será dado o número do nó restringido, o índice para o deslocamento na direção global x e o índice para o deslocamento na direção y. Esses índices são representados por 0 se o deslocamento naquela direção for livre e por 1 se for impedido.

O carregamento nodal é fornecido no bloco 7. Em cada linha, o

usuário entrará com o número do nó carregado e as forças externas aplicadas segundo a direção x e y, respectivamente, expressas em *Kips*. Caso a força em alguma direção seja nula, dever-se-á colocar zero na referida direção.

O último bloco trata dos dados da barra. Para os dois tipos de análise, entra-se consecutivamente com o número da barra, o índice referente ao nó inicial, outro para o nó final e um quarto número representando o grupo a que a barra pertence. Caso a análise seja contínua e as variáveis de área sejam tratadas individualmente, isto é, o número de grupos de barras é igual ao número de barras da estrutura, este último índice pode ser dispensado. Para a análise discreta, introduz-se ainda, na mesma linha, um índice que determinará a equação de tensão admissível para compressão de cada barra: se o Índice de Esbeltez da barra for menor ou igual a Esbeltez Crítica, este índice será 0; caso contrário será 1.

## **5.2. Processamento e Impressão da Saída .GMS**

Até o terceiro bloco de dados (conforme item 5.1), a leitura é feita de uma vez. A partir de então, os dados são processados para serem impressos no arquivo de saída, no padrão exigido pelo software de otimização GAMS.

A próxima etapa refere-se a determinação das áreas que serão

simultaneamente dados e restrição do modelo GAMS. As áreas mínima, máxima e inicial, na análise contínua, são armazenadas em variáveis e posteriormente impressas no arquivo de saída. Quando a análise é discreta, as áreas, momentos de inércia e raios de giração são calculados a partir dos diâmetros e espessuras fornecidos para os que farão parte de cada grupo de barras.

A seguir, é feita a leitura da lista de restrição dos nós, que é reordenada de forma que os graus de liberdade fiquem posicionados no início do vetor e que as coordenadas locais restringidas ocupem o final do mesmo, em um total de 2 vezes o número de nós.

A partir desta leitura, continua-se a impressão do arquivo do modelo GAMS. Em primeiro bloco os valores constantes, chamados em linguagem GAMS de `SCALARS`. Neste bloco imprime-se uma tolerância, o valor de  $\pi$ , o Módulo de Elasticidade, a densidade do material, a tensão de escoamento do material, a tensão admissível de tração e os deslocamentos nodais máximo e mínimo. Caso a análise seja discreta, inclui-se o Índice de Esbeltez Máximo e a Esbeltez Crítica da Norma AISC-ASD; senão, entra-se com a tensão admissível de compressão.

Depois, fornece-se os conjuntos, chamados de `SETS`. Neste bloco, o programa `pTRUSS` imprime o número de nós, o número de barras, o número de coordenadas e o número de graus de liberdade. Caso o problema seja discreto, inclui-se ainda o número de grupos e o número de perfis por grupo.

Na próxima etapa, elabora-se o bloco `VARIABLES`. São as variáveis do problema de otimização a serem manipuladas pelo otimizador do GAMS: os deslocamentos nodais, as tensões atuantes nas barras, as áreas do grupo de barras e o peso da estrutura. Para a análise discreta,

acrescenta-se a este bloco variáveis que representam o raio de giração, o Índice de Esbeltez e a tensão admissível de compressão para cada barra, além de uma variável definida binária  $\alpha_i$ , que servirá para determinar uma entre as diversas áreas disponíveis por grupo.

O próximo bloco, o PARAMETERS, define-se os parâmetros onde mais tarde serão atribuídos valores para compor a base de dados necessária aos otimizadores do GAMS: o comprimento das barras, os cossenos diretores em x e y, as forças externas aplicadas aos nós. Se a análise for discreta, inclui-se também parâmetros para as áreas e raio de giração dos perfis disponíveis por grupo vezes o número de grupos de barras. Então, pTRUSS inicia o processamento dos dados para atribuir os respectivos valores a esses parâmetros.

Na seqüência, tem-se a impressão do bloco de atribuição direta dos limites (.LO e .UP) e valores iniciais (.L) para algumas variáveis. São elas: áreas mínimas, máximas e iniciais; Índice de Esbeltez máximo; tensão admissível de tração (máxima) e compressão (mínima); e deslocamentos nodais máximo e mínimo.

A partir deste ponto o programa pTRUSS escreve as equações de restrição do problema de otimização de peso a ser tratado. O bloco EQUATION inicia-se com a definição e atribuição da Função Objetivo.

Depois um grupo de equações algébricas  $K \cdot U = F$  que representam o equilíbrio estrutural via Processo dos Deslocamentos é introduzido.

Na seqüência é processado um conjunto de equações para definir a área dos grupos. Se a análise for contínua e as barras forem tratadas como desagrupadas, esta parte não é processada, ficando apenas os limites dados às áreas na atribuição direta como restrição.

Na imposição das equações de restrição para a escolha das áreas das barras de cada grupo na análise discreta, usando o conceito de programação inteira, são geradas equações onde é feito o somatório dos produtos entre as áreas dos perfis disponíveis e as respectivas variáveis binárias  $\alpha_i$ , para as quais impõe-se uma soma de valor 1 (conforme item 2.4, alínea "e"). É este processo que escolhe apenas uma entre as áreas disponíveis para determinado grupo. Esta mesma variável binária irá definir o raio de giração para aquela área escolhida também através de equações de restrição.

O próximo sub-bloco de equações de restrição calcula as tensões atuantes nas barras a partir dos deslocamentos nos nós de extremidade e o comprimento da barra, definido anteriormente no bloco de atribuições a parâmetros. No caso da análise discreta, haverá também um bloco de equações para calcular as tensões admissíveis de compressão a que todas as barras possam suportar.

Encerrando o bloco das equações de restrição, o programa pTRUSS imprime as restrições de desigualdades que verifica se as tensões atuantes nas barras são menores que as tensões admissíveis.

Encerrando a geração do arquivo de dados escrito na linguagem GAMS, o programa pTRUSS imprime a definição do modelo, MODEL, do otimizador e do tipo de critério de otimização e por fim o comando que realiza executa todo o processo, SOLVE. Caso a análise seja contínua o otimizador será o MINOS e o modelo do tipo NLP (*non-linear programming*); na análise discreta, utiliza-se o otimizador DICOPT e o modelo de otimização será MINLP (*mix-integer non-linear programming*).

Para facilitar a interpretação dos resultados, o programa pTRUSS cria ainda uma série de comandos para um pós-processamento dos

resultados gerados pelo GAMS. Define-se nesta fase novas constantes com o comando `SCALARS`: o peso já definido, a tensão de escoamento do material, o módulo de elasticidade e o Índice de Esbeltez Máximo, sendo este último somente se a análise for discreta. É definida também uma tabela de resultados como parâmetro (comando `PARAMETER`), onde constarão os dados processados de área ótima, força e tensão atuante em cada barra, fator de solicitação da barra (razão entre a tensão atuante e a tensão admissível); além disso, poderá conter também dados do Índice de Esbeltez e coeficiente de segurança se o modelo analisado for discreto. Para completar o programa imprime também os deslocamentos nodais segundo as coordenadas globais.

## **6. Exemplos e Discussão**

---

Neste capítulo são apresentados resultados de exemplos de treliças planas, empregando o programa em linguagem Pascal, pTRUSS, que gerará os arquivos contendo os modelos para o software GAMS realizar a otimização do peso destas estruturas.

São apresentados dados e resultados para modelos de 5 treliças existentes na bibliografia. Os resultados obtidos com a proposta desta dissertação são comparados com os apresentados em alguns artigos de referência, com o software AutoMETAL, desenvolvido por professores da Faculdade de Engenharia Civil da Universidade Estadual de Campinas, e com o programa comercial SAP2000.

Como explicado no Capítulo 1, o programa pTRUSS permite a preparação de dados para se fazer a “análise contínua” e “análise discreta”. No segundo caso foram impostas as restrições da Norma Americana AISC-ASD para perfis tubulares. As treliças foram inicialmente submetidas a uma análise contínua utilizando-se o programa pTRUSS e o software de otimização GAMS, utilizando-se os dados utilizados pelos autores usados como referência, salvo algumas exceções. A partir daí,

com os resultados da análise contínua, para efeito de comparação, foram criados os modelos executados com o programa SAP2000 a fim de se analisar a estabilidade da estrutura. Nesta análise com o SAP2000 corrigiu-se, quando necessário, os perfis, de forma que o fator entre a tensão atuante em cada barra e a tensão admissível situe-se numa zona entre 80 e 100% de solicitação.

O programa AutoMETAL é um software que permite a análise discreta, com escolha de perfis que atendam as condições de peso mínimo e as restrições da Norma Brasileira NBR-8800, que se baseia no Método dos Estados Limites.

Apesar do autor ter consciência de que as curvas de flambagem da norma NBR-8800 serem diferentes daquelas da norma AISC-ASD e que a forma de verificação de tensões pelo Método dos Estados Limites é diferente da verificação pelo Método das Tensões Admissíveis, o programa AutoMETAL será utilizado para comparação, pelo fato de não ter sido encontrado na literatura outra fonte de referência com resultados obtidos pela norma AISC-ASD para perfis tubulares discretos. Para tal comparação é necessário então que os valores da cargas nominais sejam multiplicados por um coeficiente de ponderação  $\gamma$ . Nos exemplos vai ser admitido que todas as cargas serão permanentes, sendo portando majoradas com um fator igual a 1.4.

Tomando como referência os resultados da análise contínua do GAMS, cria-se com mais acerto os conjuntos de perfis disponíveis que serão escolhidos visando otimizar os vários grupos de barras na análise discreta. O objetivo desta pré-análise procurou evitar a não convergência dos resultados com uma tabela aleatória de perfis, problema comum em processos de otimização discreta. Por uma questão de simplificação, optou-se por utilizar uma tabela de perfis tubulares, disponível no

mercado brasileiro.

A Norma AISC-ASD utilizada no programa pTRUSS utiliza unidades inglesas. Assim, as tensões e Módulo de Elasticidade são dadas em *Ksi* (*kilo-pounds per square inch*, ou seja, quilo-libras por polegada quadrada); as unidades de comprimento e área são baseadas em polegadas inglesas ( $1 \text{ in} = 0.0254 \text{ m}$ ); quanto a densidade do material, pode ser expressa em qualquer unidade de massa ou peso, desde que em função de unidades de volume em polegadas, como por exemplo, *pci* (*pounds per cubic inch*, isto é, libras por polegada cúbica).

No site [www.carlosfmcortes.hpg.com.br](http://www.carlosfmcortes.hpg.com.br) estará disponível o programa pTRUSS, o código em linguagem Pascal para estudo e alguns exemplos de entrada de dados .DAT, os respectivos modelos gerados em linguagem GAMS e os arquivos de saída .LST.

### **6.1. Treliça de 6 Nós e 10 Barras.**

Este exemplo é clássico na bibliografia de otimização de treliças. A princípio, a comparação se dará como o trabalho proposto por ALLWOOD e CHUNG (1984).

Trata-se de uma treliça com diagonais cruzadas possuindo um total de 6 nós e 10 barras. A estrutura está sujeita a 2 cargas verticais  $P_1 = 50 \text{ Kips}$  e 2 cargas verticais  $P_2 = 150 \text{ Kips}$ , conforme mostra a **Fig. 6.1**.

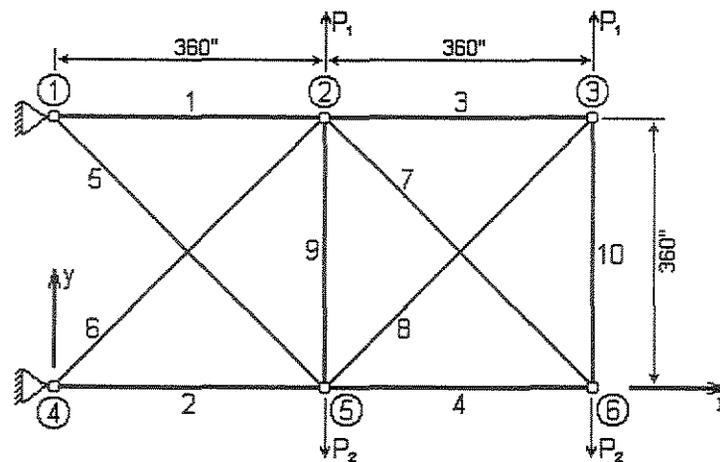


Fig. 6.1 – Treliza plana de 6 nós e 10 barras.

### 6.1.1. Resultados da Análise Contínua.

Tabela 6.1 – Comparação de resultados da análise contínua da treliza de 10 barras

| Barra                | Área (em $in^2$ ) |                |
|----------------------|-------------------|----------------|
|                      | ALLWOOD e CHUNG   | GAMS           |
| 1                    | 23.531            | 23.530         |
| 2                    | 25.285            | 25.285         |
| 3                    | 0.100             | 0.100          |
| 4                    | 14.375            | 14.374         |
| 5                    | 12.391            | 12.390         |
| 6                    | 12.828            | 12.827         |
| 7                    | 20.329            | 20.328         |
| 8                    | 0.100             | 0.100          |
| 9                    | 0.100             | 0.100          |
| 10                   | 1.970             | 1.969          |
| <b>Peso (em lbf)</b> | <b>4676.93</b>    | <b>4676.92</b> |

A treliça será modelada inicialmente com um Módulo de Elasticidade de 10000 *Ksi*, densidade do material igual a 0.1 *pci*, tensões admissíveis de compressão e tração iguais a  $\pm 25$  *Ksi*, deformação nodal limitada a 2 *in* e uma área mínima de 0.1 *in*<sup>2</sup>.

No programa *GAMS* foi escolhido para a análise contínua o pacote do otimizador *MINOS*, que utiliza o Método do Lagrangeano Aumentado. Observa-se que o Método de Newton-Raphson proposto por ALLWOOD e CHUNG obteve a mesma convergência, tornando os resultados praticamente iguais.

#### **6.1.2. Resultados da Análise Discreta.**

Para a análise discreta da treliça utilizou-se dados referentes ao aço ASTM-36 tanto no processo *pTRUSS/GAMS* como no *SAP2000* e *AutoMETAL*. O aço ASTM-36 tem um Módulo de Elasticidade de 29000 *Ksi*, densidade 0.28 *pci* e tensão de escoamento de 36 *Ksi*. É importante salientar novamente que o programa *SAP2000* e o programa *pTRUSS* utilizam restrições da norma de tensões admissíveis norte-americana *AISC-ASD/1989* e o programa *AutoMETAL* utiliza a Norma Brasileira *NBR-8800*, nos estados limites.

A amplitude dos deslocamentos nodais máximo e mínimo foi mantida em 2 *in*. A treliça foi dividida em 4 grupos distintos de barras.

**Tabela 6.2 – Perfis disponíveis por grupo de barras usados no processamento GAMS da treliça de 10 barras (vide Anexo AN-1)**

| Grupos                     | Perfis Disponíveis   |
|----------------------------|--|
| $A_1 = A_7$                | 12.7 – 12.8 – 12.13 – 12.14 – 12.15 – 12.16<br>13.1 – 13.2 – 13.3 – 13.4                     |
| $A_2 = A_4$                | 14.10 – 14.11 – 15.2 – 15.4 – 15.5   |
| $A_3 = A_8 = A_9 = A_{10}$ | 12.7 – 12.8 – 12.13 – 12.14 – 12.15 – 12.16<br>13.1 – 13.2 – 13.3 – 13.4                     |
| $A_5 = A_6$                | 11.11 – 12.12 – 13.8 – 14.4 – 14.5 – 14.8 – 14.9<br>14.10 – 15.1 – 15.2 – 15.3 – 15.4 – 15.5 |

**Tabela 6.3 – Comparação de resultados da análise discreta da treliça de 10 barras**

| Grupos                     | AutoMETAL            |                        | SAP2000              |                        | GAMS<br>(análise discreta) |                        |
|----------------------------|----------------------|------------------------|----------------------|------------------------|----------------------------|------------------------|
|                            | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ )       | Perfil<br>(D/t, em in) |
| $A_1 = A_7$                | 8.125                | 8.826/0.311            | 8.094                | 4.500/0.673            | 8.124                      | 8.626/0.311            |
| $A_2 = A_4$                | 13.349               | 12.748/0.343           | 15.724               | 12.748/0.374           | 15.742                     | 12.748/0.374           |
| $A_3 = A_8 = A_9 = A_{10}$ | 8.308                | 10.748/0.252           | 8.913                | 8.626/0.343            | 8.309                      | 10.748/0.252           |
| $A_5 = A_6$                | 10.949               | 12.748/0.281           | 10.949               | 12.748/0.281           | 10.967                     | 12.748/0.281           |
| Peso (em lbf)              | 11641.00             |                        | 12001.03             |                        | 11974.81                   |                        |

D = diâmetro do tubo; t = espessura da parede do tubo.

O que se vê neste ensaio e bem como nos próximos exemplos é uma ligeira diferença entre os pesos conseguidos com o programa AutoMETAL em comparação com os pesos obtidos com os softwares SAP2000 e GAMS. Isto se deve à diferença de metodologia de cálculo e das curvas de flambagem da Norma Americana AISC-ASD/1989 e da Norma Brasileira NBR 8800. Para o exemplo, a diferença entre o resultado do AutoMETAL com o SAP2000 e o programa GAMS foi respectivamente de 3% e 2.79%.

Uma diferença de 0.22% foi encontrada entre os resultados do SAP2000 e do GAMS. Ambos utilizam um conjunto de restrições obtidos da mesma norma de cálculo. A otimização conseguida com o programa GAMS se mostrou mais satisfatória, pois se utilizou uma técnica de otimização para conseguir os perfis ótimos; no SAP2000, o melhor resultado foi conseguido manualmente, até se obter solicitações das barras próximas a 100%.

Neste exemplo, o GAMS utilizou 4 variáveis de projeto, 42 variáveis auxiliares, 38 variáveis binárias, 12 restrições de limites e 60 equações de restrição, realizando 730 interações em cerca de 5 segundos.

## **6.2. Treliça de 11 Nós e 18 Barras.**

Este exemplo está referenciado em HANSEN e VANDERPLAATS (1990). Trata-se de uma treliça isostática de 11 nós e 18 barras sujeitas a 5 cargas  $P = 20 \text{ Kips}$  aplicadas nos nós do banzo superior, conforme a **Fig. 6.2**. A treliça será modelada, para análise contínua, com um Módulo de Elasticidade de  $10000 \text{ Ksi}$ , densidade do material igual a  $0.1 \text{ pci}$ , tensões admissíveis de compressão e tração iguais a  $\pm 20 \text{ Ksi}$  e uma área mínima de  $0.1 \text{ in}^2$ . O deslocamento nodal será desconsiderado. A fim de se ter uma melhor comparação dos resultados do artigo de referência e os resultados do GAMS, foi incluída no modelo contínuo do GAMS a restrição

da tensão de flambagem proposta por HANSEN e VANDERPLAATS, obtida a partir de uma aproximação da tensão crítica de flambagem de Eüler, dada por:

$$\sigma_{ib} = -k_i \cdot \frac{A_i}{L_i}$$

onde  $k_i = 0.4 \times 10^8 \text{ Kci}$ .

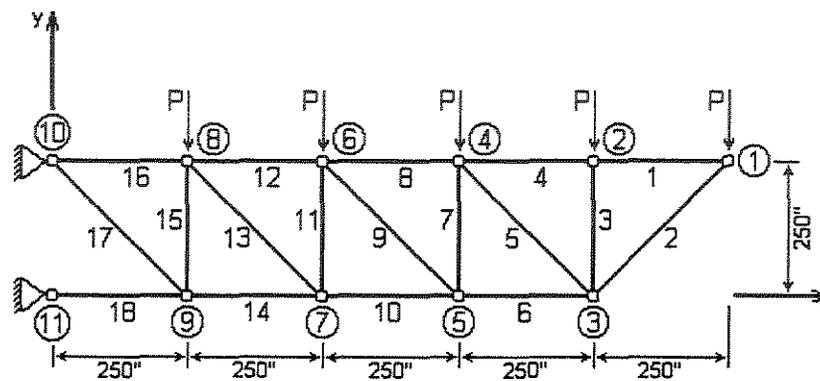


Fig. 6.2 – Treliza plana de 11 nós e 18 barras.

### 6.2.1. Resultados da Análise Contínua.

Considerou-se para esta análise contínua 4 grupos de barras. A estratégia para considerar grupos de barras no problema contínuo foi acrescentar um conjunto de restrições que impõe a igualdade entre as áreas das barras de um mesmo grupo. Os resultados obtidos estão mostrados na **Tabela 6.4**.

**Tabela 6.4 – Comparação de resultados da análise contínua da treliça de 18 barras**

| Grupos de Barras                       | Área (em $in^2$ )        |                |
|--|--------------------------|----------------|
|  | HANSEN e<br>VANDERPLAATS | GAMS           |
| $A_1 = A_4 = A_8 = A_{12} = A_{12}$    | 9.998                    | 10.000         |
| $A_2 = A_6 = A_{10} = A_{14} = A_{18}$ | 21.645                   | 21.651         |
| $A_3 = A_7 = A_{11} = A_{15}$          | 12.497                   | 12.500         |
| $A_5 = A_9 = A_{13} = A_{17}$          | 7.072                    | 7.071          |
| <b>Peso (em <math>lbf</math>)</b>      | <b>6430.00</b>           | <b>6430.52</b> |

Nota-se que os resultados foram praticamente os mesmos

### 6.2.2. Resultados da Análise Discreta.

A análise discreta foi realizada para 4 grupos de barras distintos e dados do aço *ASTM-36*. Os perfis disponíveis para cada grupo de barras usado no processamento estão mostrado na **Tabela 6.5**. No programa *GAMS*, optou-se pelo otimizador *DICOPT*.

**Tabela 6.5 – Perfis disponíveis por grupo de barras usados no processamento GAMS da treliça de 18 barras (vide Anexo AN-1)**

| Grupos                                 | Perfis Disponíveis                             |
|--|--|
| $A_1 = A_4 = A_8 = A_{12} = A_{12}$    | 14.1 – 14.2 – 14.3                             |
| $A_2 = A_6 = A_{10} = A_{14} = A_{18}$ | 16.1 – 16.2 – 16.3                             |
| $A_3 = A_7 = A_{11} = A_{15}$          | 13.1 – 13.2                                    |
| $A_5 = A_9 = A_{13} = A_{17}$          | 12.1 – 12.2 – 12.3 – 12.4 – 12.5 – 12.6 – 12.7 |

**Tabela 6.6** – Comparação de resultados da análise discreta da treliça de 18 barras

| Grupos                                 | AutoMETAL            |                           | SAP2000              |                           | GAMS<br>(análise discreta) |                           |
|--|----------------------|---------------------------|----------------------|---------------------------|----------------------------|---------------------------|
|  | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em $in$ ) | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em $in$ ) | Área<br>(em $in^2$ )       | Perfil<br>(D/t, em $in$ ) |
| $A_1 = A_4 = A_8 = A_{12} = A_{12}$    | 0.343                | 8.826/0.343               | 9.623                | 6.626/0.500               | 9.193                      | 10.748/0.280              |
| $A_2 = A_6 = A_{10} = A_{14} = A_{18}$ | 14.696               | 14.000/0.34<br>3          | 16.902               | 12.748/0.43<br>7          | 17.319                     | 14.000/0.406              |
| $A_3 = A_7 = A_{11} = A_{15}$          | 6.629                | 8.626/0.252               | 7.230                | 8.626/0.276               | 7.230                      | 8.626/0.276               |
| $A_5 = A_9 = A_{13} = A_{17}$          | 6.170                | 6.626/0.311               | 6.626                | 4.500/0.531               | 6.170                      | 6.626/0.311               |
| <b>Peso (em <math>lbf</math>)</b>      | 13187.08             |                           | 14420.58             |                           | 14349.65                   |                           |

Observa-se na Tabela 6.6 que o resultado obtido no GAMS ficou muito próximo ao do SAP2000, apresentando uma diferença de 0.49%. O dimensionamento que se adotou no SAP 2000 atendeu às condições de estabilidade mantendo pelo menos uma das barras do grupo próximo aos 100% de solicitação. O GAMS e o otimizador *DICOPT* encontraram uma configuração dimensional melhor que a estimada pelo programa SAP2000. A diferença de peso em relação ao software AutoMETAL é explicada pela diferença entre as normas utilizadas; comparando com o resultado do GAMS e do SAP2000, a diferença ficou em 8.10% e 8.55%, respectivamente.

Na análise discreta deste exemplo, o GAMS utilizou 4 variáveis de projeto, 72 variáveis auxiliares, 15 variáveis binárias, 13 restrições de limites e 102 equações de restrição, realizando 180 interações em cerca de 2 segundos.

### 6.3. Treliça de 11 Nós e 23 Barras.

Esta treliça da **Fig. 6.3** foi analisada no artigo de IMAI e SCHMIT Jr. (1981). A comparação de resultados fornecido no artigo com aqueles do programa da dissertação é feita para análise contínua.

Trata-se de uma treliça hiperestática de 11 nós, 3 apoios e 23 barras, sujeita a 4 carregamentos verticais  $P = 20 \text{ Kips}$  e uma quinta carga vertical igual a  $P/2$ , conforme ilustrado na **Fig. 6.3**. A treliça será modelada, para análise contínua, com um Módulo de Elasticidade de  $10000 \text{ Ksi}$ , densidade do material igual a  $0.1 \text{ pci}$ , tensões admissíveis de compressão e tração iguais a  $\pm 20 \text{ Ksi}$  e uma área mínima de  $0.1 \text{ in}^2$ . O deslocamento nodal será considerado livre.

#### 6.3.1. Resultados da Análise Contínua.

Os resultados comparativos estão expressos na **Tabela 6.7**. Nela, verifica-se uma melhor convergência do Método Primal-Dual proposto por IMAI e SCHMIT Jr. com relação ao Método do Lagrangeano Aumentado utilizado pelo otimizador *MINOS* do *GAMS*. Observa-se também que, por causa da hiperestaticidade, os otimizadores utilizaram caminhos distintos para convergência pelo fato das áreas por grupo serem bastante

diferentes. Independente disso, a diferença entre o resultado do GAMS e o da referência foi igual a 1.84%.

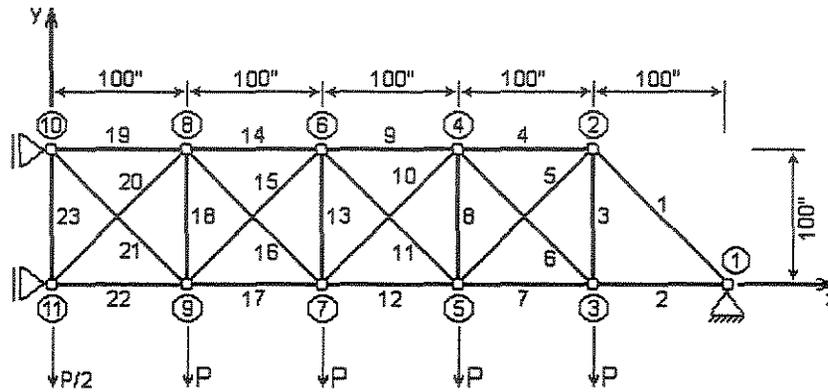


Fig. 6.3 – Treliza plana de 11 nós e 23 barras.

Tabela 6.7 – Comparação de resultados da análise contínua da treliza de 23 barras

| Grupos de Barras                       | Área (em $in^2$ ) |         |
|--|-------------------|---------|
|  | IMAI e SCHMIT Jr. | GAMS    |
| $A_1 = A_4 = A_9 = A_{14} = A_{19}$    | 12.467            | 12.305  |
| $A_2 = A_7 = A_{12} = A_{17} = A_{22}$ | 3.832             | 4.146   |
| $A_3 = A_8 = A_{13} = A_{18} = A_{23}$ | 2.109             | 2.591   |
| $A_5 = A_{10} = A_{15} = A_{20}$       | 3.688             | 2.700   |
| $A_6 = A_{11} = A_{16} = A_{21}$       | 1.392             | 2.250   |
| Peso (em $lbf$ )                       | 1259.40           | 1283.08 |

### 6.3.2. Resultados da Análise Discreta.

Para a análise discreta, utilizou-se 4 grupos de barras. Os perfis disponíveis para cada grupo de barra foram os seguintes (Tabela 6.8):

**Tabela 6.8** – Perfis disponíveis por grupo de barras usados no processamento GAMS da treliça de 23 barras (vide Anexo AN-1)

| Grupos                                 | Perfis Disponíveis   |
|--|--|
| $A_1 = A_4 = A_9 = A_{14} = A_{19}$    | 14.1 – 14.2 – 14.3 – 14.4 – 14.5 – 14.6 – 15.2                                       |
| $A_2 = A_7 = A_{12} = A_{17} = A_{22}$ | 14.1 – 14.2 – 14.3 – 14.4 – 14.5 – 14.6 – 14.7<br>14.8 – 14.9 – 14.10 – 14.11 – 12.3 |
| $A_3 = A_8 = A_{13} = A_{18} = A_{23}$ | 9.1 – 9.2 – 9.3 – 9.4 – 9.5 – 9.6 – 9.7 – 10.5                                       |
| $A_5 = A_{10} = A_{15} = A_{20}$       | 7.1 – 7.2 – 7.3 – 7.4 – 7.5 – 7.6 – 7.7 – 7.8 – 15.2                                 |
| $A_6 = A_{11} = A_{16} = A_{21}$       | 11.1 – 12.1 – 12.2 – 12.3 – 12.4   |

**Tabela 6.9** – Comparação de resultados da análise discreta da treliça de 23 barras

| Grupos                                 | AutoMETAL            |                        | SAP2000              |                        | GAMS<br>(análise discreta) |                        |
|--|----------------------|------------------------|----------------------|------------------------|----------------------------|------------------------|
|  | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ )       | Perfil<br>(D/t, em in) |
| $A_1 = A_4 = A_9 = A_{14} = A_{19}$    | 12.901               | 12.748/0.31<br>1       | 10.967               | 12.748/0.28<br>0       | 10.070                     | 10.748/0.30<br>7       |
| $A_2 = A_7 = A_{12} = A_{17} = A_{22}$ | 5.046                | 6.626/0.252            | 10.967               | 12.748/0.28<br>0       | 10.070                     | 10.748/0.30<br>7       |
| $A_3 = A_8 = A_{13} = A_{18} = A_{23}$ | 3.164                | 4.500/0.236            | 2.766                | 4.500/0.205            | 2.937                      | 2.874/0.374            |
| $A_5 = A_{10} = A_{15} = A_{20}$       | 2.076                | 2.874/0.252            | 1.842                | 2.374/0.280            | 1.894                      | 1.902/0.402            |
| $A_6 = A_{11} = A_{16} = A_{21}$       | 3.822                | 6.626/0.189            | 3.712                | 4.500/0.280            | 3.822                      | 6.626/0.189            |
| <b>Peso (em lbf)</b>                   | <b>4078.15</b>       |                        | <b>4464.89</b>       |                        | <b>4253.07</b>             |                        |

Verifica-se uma diferença respectivamente igual a 4.11% e 8.66% entre os resultados do programa AutoMETAL e os resultados dos programas GAMS e SAP2000. Entre os resultados dos programas que utilizam a Norma AISC-ASD, a diferença ficou em 4.74%, vantagem para a técnica de otimização utilizada pelo GAMS. Entretanto, em um ensaio paralelo, fez-se uma análise no programa SAP2000 com o lançamento dos

perfis resultantes do programa AutoMETAL; verificou-se que as áreas escolhidas não atenderam as tensões admissíveis da Norma AISC-ASD apenas nas barras comprimidas; isto deve-se ao fato dos métodos utilizados pelas distintas normas utilizarem curvas de flambagem diferentes. Apesar disso, os resultados podem ser considerados satisfatórios.

O otimizador DICOPT utilizado pelo GAMS, utilizou para esta treliça 5 variáveis de projeto, 93 variáveis auxiliares, 41 variáveis binárias, 15 restrições de limites e 125 equações de restrição, realizando 211 interações em cerca de 2 segundos.

#### **6.4. Torre Plana de 22 Nós e 47 Barras.**

Este exemplo mostra na **Fig. 6.4** um modelo de torre simplificada em duas dimensões, sugerida no trabalho de VANDERPLAATS e MOSES (1972). Eles ensaiaram esta treliça com tensão de compressão limitada em 15 *Ksi* e tensão de tração de 20 *Ksi*. O Módulo de Elasticidade adotado foi de 30000 *Ksi* e densidade do material igual a 0.3 *pci*. O deslocamento nodal não foi restringido, a área mínima foi limitada em  $10^{-6}$  *in*<sup>2</sup> e 27 variáveis de área independentes foram consideradas. Outra restrição adotada por VANDERPLAATS e MOSES e inserida no modelo GAMS, apenas para a análise contínua, foi um limitante para tensão crítica de Eüler, aproximada pela seguinte equação:

$$\sigma_{ib} = \frac{10.1 \cdot \pi \cdot E \cdot A_i}{8 \cdot L_i^2}$$

#### 6.4.1. Resultados da Análise.

Neste exemplo, houve uma dificuldade muito grande em se conseguir uma convergência com o *GAMS*. Provavelmente por problemas no otimizador *DICOPT*, a variável binária de algumas grupos de barras não conseguia retornar o valor unitário, que identifica o perfil escolhido dentro dos disponíveis para aquele grupo de barra. A análise com programação inteira-mista apresentou resultado infactível, apresentando folga de variáveis e retornando valores de área intermediários entre dois perfis consecutivos. Desta forma, a aproximação foi feita manualmente para o maior perfil subsequente, sem violar o quesito de segurança proposto.

Quanto analisado por um modelo contínuo, o resultado obtido pelo programa *GAMS* foi de 1758.48 *lb*, muito bom em relação ao resultado de *VANDERPLAATS* e *MOSES*, que foi de 1861.00 *lb*, resultando uma diferença de aproximadamente 5%. As áreas ótimas encontradas nesta análise serviram de referência para a montagem dos grupos de perfis disponíveis necessários na modelagem do problema discreto.

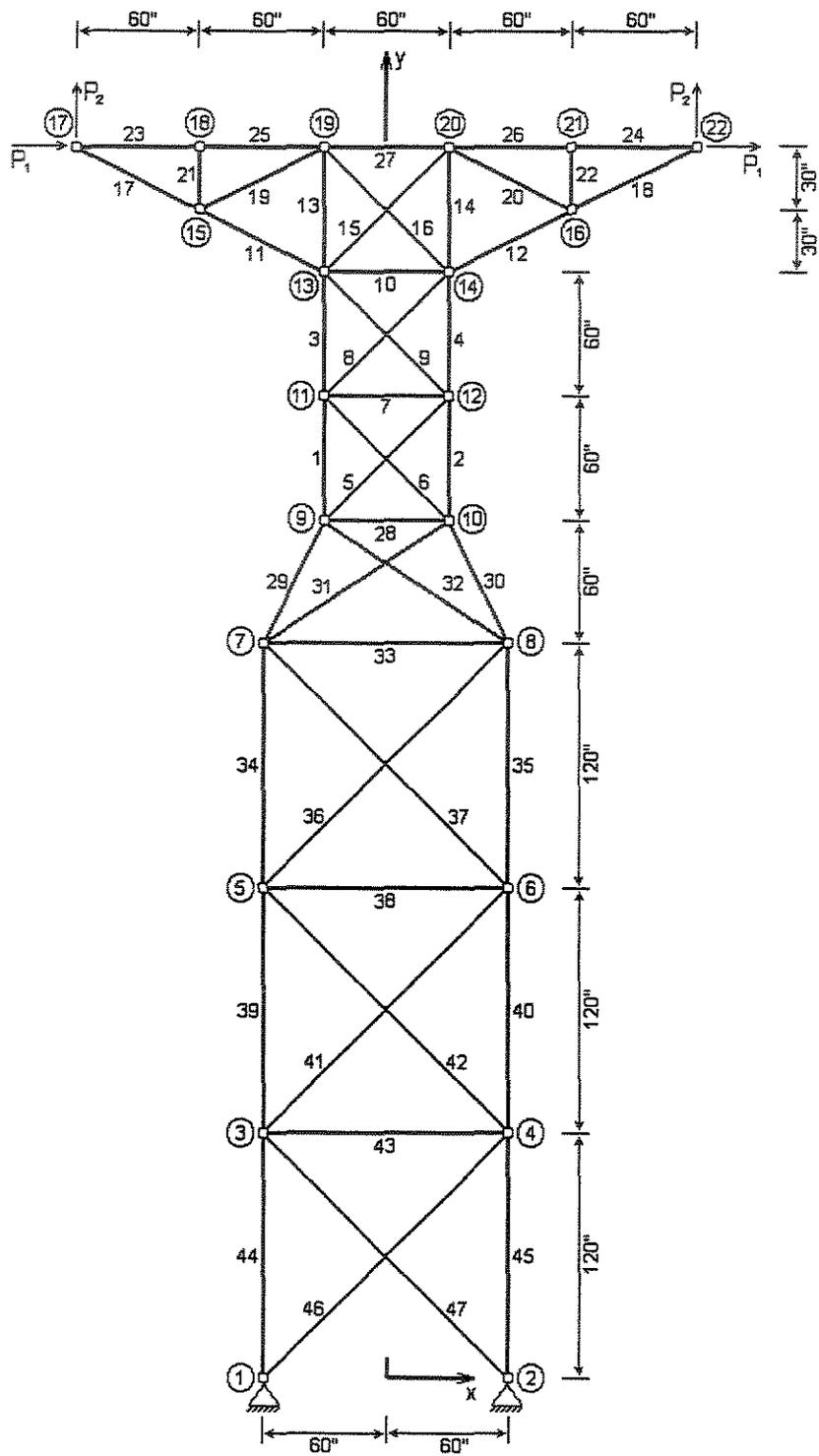


Fig. 6.4 – Torre plana de 22 nós e 47 barras.

**Tabela 6.10** – Perfis disponíveis por grupo de barras usados no processamento GAMS da treliça de 47 barras (vide Anexo AN-1)

| Grupos                            | Perfis Disponíveis                    |
|-----------------------------------|---------------------------------------|
| A <sub>1</sub> = A <sub>2</sub>   | 7.2 – 7.4 – 7.6 – 7.8 – 7.10          |
| A <sub>3</sub> = A <sub>4</sub>   | 7.2 – 7.4 – 7.6 – 7.8                 |
| A <sub>5</sub> = A <sub>6</sub>   | 6.1 – 6.2                             |
| A <sub>7</sub>                    | 1.1 – 1.2                             |
| A <sub>8</sub> = A <sub>9</sub>   | 7.1 – 7.2                             |
| A <sub>10</sub>                   | 6.1 – 6.3 – 6.5 – 6.7 – 6.9           |
| A <sub>11</sub> = A <sub>12</sub> | 7.2 – 7.4 – 7.6 – 7.8                 |
| A <sub>13</sub> = A <sub>14</sub> | 4.2 – 4.4 – 4.6 – 5.3                 |
| A <sub>15</sub> = A <sub>16</sub> | 6.1 – 6.2                             |
| A <sub>17</sub> = A <sub>18</sub> | 7.2 – 7.4 – 7.6 – 7.8                 |
| A <sub>19</sub> = A <sub>20</sub> | 2.1 – 2.2                             |
| A <sub>21</sub> = A <sub>22</sub> | 1.1 – 1.2                             |
| A <sub>23</sub> = A <sub>24</sub> | 5.4 – 6.9 – 7.9 – 7.11                |
| A <sub>25</sub> = A <sub>26</sub> | 5.4 – 6.9 – 7.9 – 7.11                |
| A <sub>27</sub>                   | 4.2 – 4.4 – 4.6                       |
| A <sub>28</sub>                   | 3.1 – 3.2                             |
| A <sub>29</sub> = A <sub>30</sub> | 9.2 – 9.4 – 9.6                       |
| A <sub>31</sub> = A <sub>32</sub> | 7.1 – 7.3                             |
| A <sub>33</sub>                   | 5.1 – 5.2                             |
| A <sub>34</sub> = A <sub>35</sub> | 10.1 – 10.3 – 10.5 – 10.7             |
| A <sub>36</sub> = A <sub>37</sub> | 8.1 – 8.2                             |
| A <sub>38</sub>                   | 5.1 – 5.2                             |
| A <sub>39</sub> = A <sub>40</sub> | 10.10 – 10.12 – 10.14 – 11.10         |
| A <sub>41</sub> = A <sub>42</sub> | 8.1 – 8.3                             |
| A <sub>43</sub>                   | 5.1 – 5.2                             |
| A <sub>44</sub> = A <sub>45</sub> | 10.10 – 10.12 – 10.14 – 11.10 – 12.11 |
| A <sub>46</sub> = A <sub>47</sub> | 8.1 – 8.3                             |

A **Tabela 6.10** mostra os perfis disponíveis por cada grupo de barras adotados no processamento. A **Tabela 6.11** mostra os resultados comparados.

**Tabela 6.11 – Comparação de resultados da análise discreta da treliça de 47 barras**

| Grupos                            | AutoMETAL                     |                        | SAP2000                       |                        | GAMS<br>(análise discreta)    |                        |
|-----------------------------------|-------------------------------|------------------------|-------------------------------|------------------------|-------------------------------|------------------------|
|                                   | Área<br>(em in <sup>2</sup> ) | Perfil<br>(D/t, em in) | Área<br>(em in <sup>2</sup> ) | Perfil<br>(D/t, em in) | Área<br>(em in <sup>2</sup> ) | Perfil<br>(D/t, em in) |
| A <sub>1</sub> = A <sub>2</sub>   | 1.594                         | 2.874/0.189            | 2.355                         | 4.500/0.173            | 1.217                         | 2.874/0.142            |
| A <sub>3</sub> = A <sub>4</sub>   | 1.344                         | 2.874/0.157            | 1.654                         | 3.500/0.157            | 2.076                         | 2.874/0.252            |
| A <sub>5</sub> = A <sub>6</sub>   | 0.994                         | 2.374/0.142            | 0.890                         | 2.374/0.126            | 0.994                         | 2.374/0.142            |
| A <sub>7</sub>                    | 0.336                         | 1.051/0.114            | 0.434                         | 1.051/0.154            | 0.433                         | 1.051/0.154            |
| A <sub>8</sub> = A <sub>9</sub>   | 0.994                         | 2.374/0.142            | 1.088                         | 2.874/0.126            | 1.088                         | 2.874/0.126            |
| A <sub>10</sub>                   | 1.071                         | 2.374/0.154            | 1.470                         | 2.874/0.173            | 1.680                         | 2.374/0.252            |
| A <sub>11</sub> = A <sub>12</sub> | 1.594                         | 2.874/0.189            | 1.810                         | 3.500/0.173            | 1.217                         | 2.874/0.142            |
| A <sub>13</sub> = A <sub>14</sub> | 1.116                         | 1.661/0.252            | 0.590                         | 0.661/0.122            | 1.116                         | 1.661/0.252            |
| A <sub>15</sub> = A <sub>16</sub> | 0.994                         | 2.374/0.142            | 0.890                         | 2.374/0.126            | 0.890                         | 2.374/0.126            |
| A <sub>17</sub> = A <sub>18</sub> | 1.694                         | 2.874/0.189            | 1.810                         | 3.500/0.173            | 1.217                         | 2.874/0.142            |
| A <sub>19</sub> = A <sub>20</sub> | 0.497                         | 1.315/0.134            | 0.336                         | 1.051/0.114            | 0.497                         | 1.315/0.134            |
| A <sub>21</sub> = A <sub>22</sub> | 0.433                         | 1.051/0.154            | 0.336                         | 1.051/0.114            | 0.336                         | 1.015/0.114            |
| A <sub>23</sub> = A <sub>24</sub> | 1.892                         | 1.092/0.402            | 1.594                         | 2.874/0.189            | 1.892                         | 1.902/0.402            |
| A <sub>25</sub> = A <sub>26</sub> | 1.892                         | 1.902/0.402            | 1.594                         | 2.874/0.189            | 1.892                         | 1.902/0.402            |
| A <sub>27</sub>                   | 1.535                         | 1.661/0.382            | 1.340                         | 2.874/0.157            | 1.535                         | 1.661/0.382            |
| A <sub>28</sub>                   | 0.806                         | 1.500/0.197            | 0.806                         | 1.500/0.197            | 0.806                         | 1.500/0.197            |
| A <sub>29</sub> = A <sub>30</sub> | 2.967                         | 4.000/0.252            | 2.967                         | 4.000/0.252            | 2.967                         | 4.000/0.252            |
| A <sub>31</sub> = A <sub>32</sub> | 1.344                         | 2.874/0.157            | 1.088                         | 2.874/0.126            | 1.088                         | 2.874/0.126            |
| A <sub>33</sub>                   | 0.804                         | 1.902/0.146            | 0.804                         | 1.902/0.146            | 1.073                         | 1.902/0.201            |
| A <sub>34</sub> = A <sub>35</sub> | 3.164                         | 4.500/0.236            | 2.958                         | 4.500/0.220            | 1.941                         | 4.500/0.142            |
| A <sub>36</sub> = A <sub>37</sub> | 1.654                         | 3.500/0.157            | 1.498                         | 3.500/0.142            | 1.654                         | 3.500/0.157            |
| A <sub>38</sub>                   | 0.804                         | 1.902/0.146            | 0.804                         | 1.902/0.146            | 0.804                         | 1.902/0.146            |
| A <sub>39</sub> = A <sub>40</sub> | 4.093                         | 4.500/0.311            | 3.693                         | 5.563/0.220            | 4.093                         | 4.500/0.311            |
| A <sub>41</sub> = A <sub>42</sub> | 1.810                         | 3.500/0.173            | 1.498                         | 3.500/0.142            | 1.810                         | 3.500/0.173            |
| A <sub>43</sub>                   | 0.804                         | 1.902/0.146            | 0.804                         | 1.902/0.146            | 0.804                         | 1.902/0.146            |
| A <sub>44</sub> = A <sub>45</sub> | 1.470                         | 4.500/0.437            | 4.426                         | 4.500/0.339            | 4.093                         | 4.500/0.311            |
| A <sub>46</sub> = A <sub>47</sub> | 1.810                         | 3.500/0.173            | 1.498                         | 3.500/0.142            | 1.810                         | 3.500/0.173            |
| <b>Peso (em lbf)</b>              | <b>2022.06</b>                |                        | <b>2063.77</b>                |                        | <b>2064.72</b>                |                        |

Neste exemplo, o programa da dissertação conseguiu os resultados mais compatíveis com os resultados do programa SAP2000 em relação aos demais exemplos, com a insignificante diferença de 0.05%. Uma diferença mais significativa, de 2.07%, foi registrada na comparação com o programa AutoMETAL, devido a diferenças entre as normas adotadas.

Nesta análise, o GAMS utilizou 27 variáveis de projeto, 209 variáveis auxiliares, 81 variáveis binárias, 59 restrições de limites e 309 equações de restrição, realizando 300 interações em cerca de 8 segundos.

## **6.5 Treliza de 28 Nós e 64 Barras.**

O último exemplo é uma treliza de 28 nós, 4 apoios de 2º gênero e 64 barras retirada do artigo de SCHMIT Jr. e LAI, 1994. A treliza foi analisada com 19 variáveis de áreas independentes. Neste exemplo, mostrado na **Fig. 6.5**, a carga  $P_1$  vale 70 *Kips*, a carga  $P_2$  é igual a 100 *Kips* e a carga  $P_3$  vale 20 *Kips*; a tensão admissível para cada membro foi limitada em 25 *Ksi* para tração ou compressão e os deslocamentos nodais mínimo e máximo foram de 10 *in*.

### **6.5.1. Resultados da Análise.**

Os resultados da análise contínua, cujas áreas serviram como

referência para a escolha dos perfis disponíveis para o ensaio discreto, obtidos pelo programa *GAMS* retornou um peso ótimo de 4533.13 lb. Na referência utilizada, os autores não mostraram os valores das áreas ótimas nem do peso final da treliça, mostrando apenas uma tabela com os valores dos deslocamentos nodais.

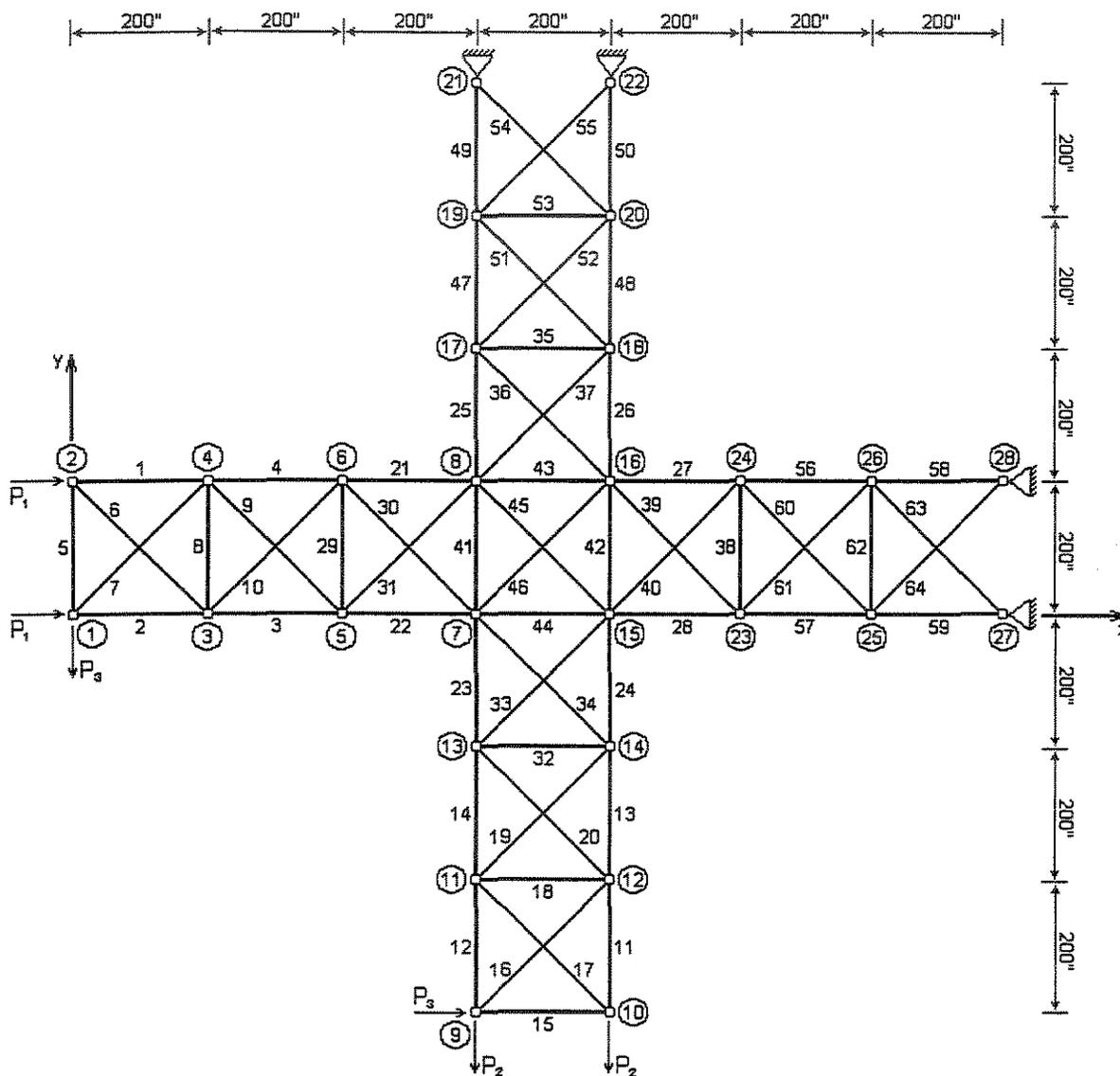


Fig. 6.5 – Treliça plana de 28 nós e 64 barras.

Os resultados discretos desta treliça estão apresentados na Tabela 6.13. Como na análise da treliça do exemplo anterior, a

convergência teve de ser conseguida manualmente, depois de alterar várias vezes a tabela de perfis, pois a análise se mostrou infactível com os perfis inicialmente lançados, sem, no entanto, prejudicar a estabilidade da estrutura.

**Tabela 6.12** – Perfis disponíveis por grupo de barras usados no processamento GAMS da treliça de 64 barras (vide Anexo AN-1)

| Grupos  | Perfis Disponíveis                        |
|---|---|
| $A_1 = A_2 = A_3 = A_4$                               | 12.2 – 12.4 – 12.7                        |
| $A_5 = A_6 = A_7$                                     | 11.1 – 11.2 – 11.5                        |
| $A_8 = A_9 = A_{10}$                                  | 11.1 – 11.4 – 11.5                        |
| $A_{11} = A_{12} = A_{13} = A_{14}$                   | 12.1 – 12.3 – 12.4 – 12.6                 |
| $A_{15} = A_{16} = A_{17}$                            | 10.1                                      |
| $A_{18} = A_{19} = A_{20}$                            | 10.1                                      |
| $A_{21} = A_{22} = A_{27} = A_{28} = A_{43} = A_{44}$ | 13.1 – 13.6                               |
| $A_{23} = A_{24} = A_{25} = A_{26} = A_{41} = A_{42}$ | 11.1 – 11.5 – 11.9 – 11.10                |
| $A_{29} = A_{30} = A_{31}$                            | 10.1 – 10.3                               |
| $A_{32} = A_{33} = A_{34}$                            | 10.1 – 10.4                               |
| $A_{35} = A_{36} = A_{37}$                            | 10.1 – 10.12 – 10.13                      |
| $A_{38} = A_{39} = A_{40}$                            | 10.10 – 10.13 – 11.9 – 12.9 – 13.5 – 14.2 |
| $A_{45} = A_{46}$                                     | 10.1 – 10.13 – 11.9                       |
| $A_{47} = A_{48} = A_{49} = A_{50}$                   | 12.6 – 12.16                              |
| $A_{51} = A_{52}$                                     | 10.1 – 10.3 – 10.14                       |
| $A_{53} = A_{54} = A_{55}$                            | 10.1 – 10.9 – 10.14                       |
| $A_{56} = A_{57} = A_{58} = A_{59}$                   | 12.6 – 12.9 – 12.10                       |
| $A_{60} = A_{61}$                                     | 10.5 – 10.11 – 10.12 – 11.6               |
| $A_{62} = A_{63} = A_{64}$                            | 11.1 – 11.4 – 11.6                        |

Na comparação de resultados, ao se comparar os softwares SAP2000 e o da Dissertação, que utilizaram a mesma norma de tensões admissíveis, o programa pTRUSS/GAMS obteve melhor resultado, com

2.16% de diferença a menos. Com relação aos resultados do AutoMETAL, a diferença ficou em 11.08% e 9.12%, respectivamente.

O GAMS utilizou para esta análise 19 variáveis de projeto, 260 variáveis auxiliares, 55 variáveis binárias, 43 restrições de limites e 361 equações de restrição, realizando 336 interações em cerca de 13 segundos.

**Tabela 6.13** – Comparação de resultados da análise discreta da treliça de 64 barras

| Grupos   | AutoMETAL            |                        | SAP2000              |                        | GAMS<br>(análise discreta) |                        |
|--|----------------------|------------------------|----------------------|------------------------|----------------------------|------------------------|
|  | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ ) | Perfil<br>(D/t, em in) | Área<br>(em $in^2$ )       | Perfil<br>(D/t, em in) |
| $A_1 = A_2 = A_3 = A_4$                                    | 6.170                | 6.626/0.311            | 6.761                | 6.626/0.343            | 4.130                      | 6.626/0.205            |
| $A_5 = A_6 = A_7$  | 3.700                | 5.563/0.220            | 3.700                | 5.563/0.220            | 5.132                      | 5.563/0.311            |
| $A_8 = A_9 = A_{10}$                                       | 3.700                | 5.563/0.220            | 4.640                | 5.563/0.280            | 5.132                      | 5.563/0.311            |
| $A_{11} = A_{12} = A_{13} = A_{14}$                        | 5.455                | 3.500/0.598            | 6.170                | 6.626/0.311            | 3.822                      | 6.626/0.189            |
| $A_{15} = A_{16} = A_{17}$                                 | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 1.941                      | 4.500/0.142            |
| $A_{18} = A_{19} = A_{20}$                                 | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 1.941                      | 4.500/0.142            |
| $A_{21} = A_{22} = A_{27} = A_{28}$<br>$= A_{43} = A_{44}$ | 8.125                | 8.626/0.311            | 9.696                | 8.626/0.374            | 9.696                      | 8.626/0.374            |
| $A_{23} = A_{24} = A_{25} = A_{26}$<br>$= A_{41} = A_{42}$ | 7.230                | 8.626/0.276            | 7.953                | 5.563/0.500            | 3.190                      | 5.563/0.189            |
| $A_{29} = A_{30} = A_{31}$                                 | 2.148                | 4.500/0.157            | 2.355                | 4.500/0.173            | 2.355                      | 4.500/0.173            |
| $A_{32} = A_{33} = A_{34}$                                 | 2.355                | 4.500/0.173            | 2.559                | 4.500/0.189            | 2.559                      | 4.500/0.189            |
| $A_{35} = A_{36} = A_{37}$                                 | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 6.626                      | 4.500/0.531            |
| $A_{38} = A_{39} = A_{40}$                                 | 3.190                | 5.563/0.189            | 4.093                | 4.500/0.311            | 6.626                      | 4.500/0.531            |
| $A_{45} = A_{46}$  | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 1.941                      | 4.500/0.142            |
| $A_{47} = A_{48} = A_{49} = A_{50}$                        | 6.626                | 4.500/0.531            | 7.346                | 6.626/0.374            | 7.346                      | 6.626/0.374            |
| $A_{51} = A_{52}$  | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 1.941                      | 4.500/0.142            |
| $A_{53} = A_{54} = A_{55}$                                 | 1.941                | 4.500/0.142            | 1.941                | 4.500/0.142            | 1.941                      | 4.500/0.142            |
| $A_{56} = A_{57} = A_{58} = A_{59}$                        | 7.230                | 8.626/0.276            | 9.623                | 6.626/0.500            | 6.170                      | 6.626/0.311            |
| $A_{60} = A_{61}$  | 3.190                | 5.563/0.189            | 4.426                | 4.500/0.339            | 5.618                      | 5.563/0.343            |
| $A_{62} = A_{63} = A_{64}$                                 | 3.190                | 5.563/0.189            | 3.190                | 5.563/0.189            | 3.190                      | 5.563/0.189            |
| <b>Peso (em lbf)</b>                                       | 17793.58             |                        | 20013.00             |                        | 19580.48                   |                        |

## **7. Conclusões**

Desenvolveu-se neste trabalho um procedimento para otimização de treliças planas, através da minimização do peso desta estrutura, sujeito a restrições de equilíbrio estático, de deslocamentos canalizados e tensões nas barras limitadas por tensões admissíveis calculadas conforme a Norma de Tensões Admissíveis norte-americana *AISC-ASD/1989*.

Foi utilizado o programa comercial *GAMS*, que dentre as suas potencialidades permite a resolução de problemas de programação matemática não lineares com Função Objetivo e equações de restrição também não lineares. O programa *GAMS* requer a elaboração de um arquivo de dados específico, escrito numa linguagem própria, a ser interpretado pelo software, disparando as rotinas adequadas de programação não linear em que recai o problema em análise.

A minimização do peso de treliças, da forma como conduzida na dissertação, recai num problema de programação não linear com algumas variáveis contínuas e outras inteiras (binárias).

Na dissertação foi escrito um programa denominado *pTRUSS*, escrito em linguagem Pascal, capaz de gerar o arquivo necessário para a

execução do programa *GAMS*, onde o problema da otimização de treliças planas basicamente recai na minimização do peso da estrutura, com restrições de descolamentos que garantam o equilíbrio estático e tensões respeitando determinadas tensões admissíveis.

No presente trabalho, limitou-se a utilização de perfis tubulares. Para a utilização de perfis com outras características será necessário a modificação do programa *pTRUSS*, incorporando ao mesmo as restrições normativas referentes aos perfis desejados. Caso se deseje a verificação segundo outras normas de cálculo, serão necessárias as trocas das equações de restrições implementadas em tal programa.

Foram implementados dois tipos distintos de análise: análise contínua e análise discreta. No primeiro caso as áreas ótimas escolhidas para as barras podem assumir qualquer valor positivo. No caso da análise discreta os perfis escolhidos devem constar de uma biblioteca de perfis fornecidos pelo usuário. Nos dois casos é possível que o usuário escolha que a seção transversal de um determinado conjunto de barras tenham as mesmas características geométricas. Esse tipo de arranjo recebeu no trabalho o nome de arranjo por grupos.

A utilização do software comercial *GAMS*, projetado para fins de Pesquisa Operacional, se constituiu num grande desafio nos trabalhos de análise para esta Dissertação, pois pouco se conhecia do comportamento do *GAMS* na geração dos modelos com muitas variáveis e equações com grau de complexidade elevado, principalmente no que concerne ao emprego de uma técnica de otimização não-linear com variáveis discretas. O uso do programa *GAMS* para esta finalidade parece ter um caráter inovador para a Engenharia Civil, principalmente na área de Estruturas.

Os resultados obtidos para a análise contínua foram bastante satisfatórios. Nesse tipo de problema os modelos gerados foram analisados através do software *GAMS* por uma técnica de programação não-linear usando o Método do Lagrangeano Aumentado. Nos exemplos analisados o peso das treliças planas analisadas com o programa *GAMS* se aproximou bem dos resultados obtidos nos artigos que serviram de referência. Em alguns casos a convergência para o ponto ótimo foi melhor conseguida com o programa *GAMS* do que nas soluções propostas pelos autores usados como referência.

Na análise discreta, o problema se mostrou mais complexo. Para tal análise é necessário fornecer uma biblioteca com um conjunto de perfis disponíveis para cada grupo de barras. Muitas vezes a solução se mostrou infactível para a biblioteca inicialmente fornecida, mas com a inclusão de alguns perfis, de diâmetro e espessura próximos aos pré-existentes, uma solução ótima foi encontrada. Os modelos maiores e mais complexos, com muitos grupos de barras foram os que apresentaram maiores problemas de convergência para uma solução ótima. Em algumas análises, dependendo da combinação dos valores iniciais, e mesmo sabendo que aqueles perfis eram satisfatórios, o *GAMS* retornou resultados infactíveis. Foi necessário adquirir uma certa sensibilidade na interpretação do arquivo de saída do *GAMS*, a fim de encontrar os fatores que geravam as infactibilidades, procurando meios de superar tais efeitos, modificando a biblioteca de elementos disponíveis. Em alguns casos de infactibilidade, o *GAMS* retornou um valor ótimo, entretanto escolheu áreas que eram intermediárias entre dois perfis disponíveis no grupo, revelando um problema no programa de relaxação das variáveis binárias, que eram coeficientes multiplicadores das áreas disponíveis, para garantir a escolha de um único perfil. Quanto isto ocorreu foi necessário alterar a biblioteca de tubos disponíveis, incluindo o perfil imediatamente superior disponível no respectivo grupo.

Para conferência utilizou-se o software comercial SAP2000 que possui um módulo de inclusão de uma biblioteca de perfis e um módulo de dimensionamento estrutural por tensões admissíveis usando também a Norma norte-americana AISC/ASD 1989, baseada no Método das Tensões Admissíveis. Utilizou-se também, mais como referência que como critério de comparação, o software acadêmico denominado AutoMETAL, que faz o dimensionamento pela norma brasileira NBR-8800 da ABNT, elaborada segundo o Método dos Estados Limites.

Pelos exemplos apresentados conclui-se que a norma das tensões admissíveis *AISC-ASD* apresentou sempre resultados com diferenças entre 3 e 11% com relação a norma brasileira NBR 8800. Isto deve-se ao fato de que para esforços de compressão, a norma americana utiliza apenas uma curva de flambagem, que só é utilizada quando a razão entre a tensão de compressão atuante na barra e a tensão admissível para aquele perfil fica aquém de 52% (vide Anexo AN-3), enquanto a norma brasileira utiliza diversas curvas de flambagem. Outra diferença é que a norma NBR8800, usada no programa AutoMetal faz as verificações baseado no conceito dos estados limites, minimizando as tensões e majorando as cargas externas atuante na estrutura, o que não ocorre na norma *AISC-ASD*, que utiliza os valores nominais das cargas. As barras sujeitas a tração não influenciaram significativamente as diferenças dos resultados entre as duas normas consideradas, pois o coeficiente de minoração da tensão de escoamento é fixado em 60% para a norma americana e de gira em torno de 64% para norma brasileira NBR-8800.

Como proposta de continuação do trabalho recomenda-se a inclusão no programa *pTRUSS* da possibilidade de verificação de outros tipos de perfis, utilizando a norma das tensões admissíveis, bem como a Norma Brasileira NBR-8800. Uma outra proposta, mais ambiciosa, seria a

de uma nova abordagem do problema, incluindo a possibilidade de se realizar uma otimização geométrica e até topológica.

Finalizando considera-se que a modelagem proposta se mostrou interessante e a utilização do software *GAMS*, uma alternativa viável pela sua versatilidade na escolha de métodos consagrados de otimização.

## **8. Referência Bibliográfica**

---

**ALLWOOD, R. J.; CHUNG, Y. S.** Minimum-weight Design of Trusses by an Optimality Criteria Method. International Journal for Numerical in Engineering, v. 20, p. 697-713, March/1984.

**ALSPAUGH, Dale; KUNOO, Kazuo.** Optimum Configurational and Dimensional Design of Truss Structures. Computers and Structures, Vol. 4, pp. 755-770, 1974.

**AVRIEL, Mordecai.** Nonlinear Programming – Analysis and Methods. Prentice-Hall, Inc. – Englewood Cliffs, New Jersey, 1976.

**BARBIERI, E.; LOMBARDI, M.** Minimum Weight Shape and Size Optimization of Truss Structures Made on Uncertain Materials. Structural Optimization, v. 16, i. 2-3, p. 147-154, October/1998.

**BAZARAA, M. S.; SHETTY, C. M.** Nonlinear Programming - Theory and Algorithms. John Wiley & Sons, New York, 1979.

**BOFFEY, T. B.; YATES, D. F.** A Simplex-based Approach to a Class of Problems Associated with Truss Design. Engineering Optimization, v. 28, i. 1-2, p. 127-156, 1997.

**BROOKE**, Anthony; **DAVID**, Kendrik; **MEERAUS**, Alexander. GAMS – Sistema Geral de Modelagem Algébrica. 1ª edição. Ed. Edgard Blücher Ltda., São Paulo, 1997.

**CHATTOPADHYAY**, Debabrata. Application of General Algebraic Modelling System to Power System Optimization. IEEE Transactions on Power Systems, v. 14, i. 1, p 15-22, february/1999.

**CHENG**, G. Some Aspects of Truss Topology Optimization. Structural Optimization, v. 10, i. 3-4, p. 173-179, December/1995.

**DOBBS**, M. W.; **FELTON**, L. P. Optimization of Truss Geometry. Journal of the Structural Division, Vol. 95, i. 10, pp. 2105-2118, 1969.

**DORN**, W. S.; **GOMORY**, R. E.; **GREENBERG**, H. J. Automatic Design of Optimal Structures. Journal de Mechanique, Vol. 3, i. 1, 1969. In: MOURA [1977].

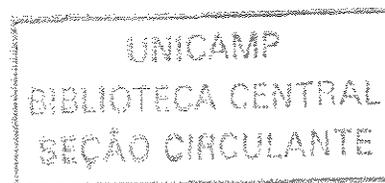
**FALCÓN**, Ginés A. S. Otimização Geométrica de estruturas Treliçadas. Tese – COPPE/UFRJ, Rio de Janeiro, 1991.

**FREITAS Neto**, José A. Análise Matricial de Estruturas – Vol. II. Ed. Hoje, Curitiba, 1984.

**GERE**, James M.; **WEAVER Jr.**, William. Análise de Estruturas Reticuladas. Ed. Guanabara Dois, Rio de Janeiro, 1981.

**GIL**, Lluís; **ANDREU**, Antoni. Shape and Cross-section Optimization of a Truss Structure. Computers and Structures, v. 79, p. 681-689, 2001.

- GUILHERME**, Carlos E. M.; **FONSECA**, Jun S. O.; **SANTOS**, Evandro D. M. Otimização Topológica de Treliças e Pórticos Sujeita à Restrições de Flexibilidade e Flambagem. Computational Methods in Engineering, 1999.
- GUTKOWSKI**, W.; **BAUER**, J. Structural Optimization with Sensitivity Constraints: Statics. Computers and Structures, v. 52, i. 1, p. 121-125, July/1994.
- HAFTKA**, R. T.; **GÜRDAL**, Z. Elements of Structural Optimization. 3<sup>rd</sup> Edition, Kluwer Academic Publishers, 1992.
- HANSEN**, S. R.; **VANDERPLAATS**, G. N. An Approximation Methods for Configuration Optimization of Trusses. AIAA Journal, v. 28, n. 1, p. 161-168, 1990.
- HIMMEBLAU**, D. M. Applied Nonlinear Programming. McGraw-Hill Book Company, New York, 1972.
- IMAI**, K; **SCHMIT**, L. A. Configuration Optimization of Trusses. Journal of the Structural Division; p. 745-756, May/1981.
- KOEHLER**, Nederson S.; **ALVES**, Marcelo K. Otimização de Forma e Dimensional de Estruturas Treliçadas. XV Congresso Brasileiro de Engenharia Mecânica, Águas de Lindóia, São Paulo, novembro/1999.
- LAMBERTI**, Luciano; **PAPPALETTERE**, Carmine. Comparison of the Numerical Efficiency of Different Sequential Linear Programming Based Algorithms for Structural Optimization Problems. Computers and Structures, v. 76, i. 6, p. 713-728, July/2000.



- LASSEN, Tom.** Optimum Design of Three-dimensional Framework Structures. Journal of Structural Engineering, v. 119, i. 3, p. 713-727, March/1993.
- LEVY, R.** Optimization for Buckling with Exact Geometry. Computers and Structures, v. 53, i. 5, p. 1139-1244, December/1994.
- LOESCH, C.; HEIN, N.** Pesquisa Operacional: Fundamentos e Modelos. Ed. Da FURB, Blumenau, 1999.
- LUENBERGER, D. G.** Introduction to Nolinear Programming. Addison-Wesley Publishing Company, London, 1984.
- MATEUS, G. R.; LUNA, H. P. L.** Programação Não-linear. V Escola de Computação, Belo Horizonte, 1986.
- MOURA, Vinicius F.** Otimização Geométrica de Treliças. Tese – COPPE/UFRJ, Rio de Janeiro, 1977.
- MURTAGH, Bruce A.; SAUNDERS, Michael A.** MINOS 5.1: User's Guide. Stanford University, 1987.
- OBENDORFER, J. M.; ACHTZIGER, W.; HORNLEIN, H. R. E. M.** Two Approaches for Truss Topology Optimization: a Comparison for Practical Use. Structural Optimization, v. 11, i. 3-4, p. 137-144, June/1996.
- OLIVEIRA, P. R.** Introdução a Programação Não-Linear. I Escola Brasileira de Otimização (EBO), Rio de Janeiro, 1989.
- PEDERSEN, Pauli.** On the Optimal Layout of Multi-Purpose Trusses. Computers and Structures, Vol. 2, pp. 695-712, 1972.

- PRUDENTE**, Mauro. Otimização de Estruturas de Aço Treliçadas Planas com Variáveis Discretas. Tese de Doutorado - Escola de Engenharia de São Carlos, Universidade de São Paulo, 1998.
- PUCCHINI**, A. L. Introdução à Programação Linear. Livros Técnicos e Científicos Editora, Rio de Janeiro, 1978.
- REQUENA**, João A. V.; **SOUZA**, José L. A. O. Estudo e Automação do Dimensionamento de Elementos Estruturais Constituído de Perfis Formados a Frio (Apóstila). Universidade Estadual de Campinas, Faculdade de Engenharia Civil, Campinas, São Paulo, 2000.
- SAKA**, M. P.; **ULKER**, M. Optimum Design of Geometrically Non-linear Space Trusses. Computers and Structures, v. 41, i. 6, p. 1387-1396, 1991.
- SCHMIT**, Lucien A.; **LAI**, Ye-Chen. Structural Optimization Based on Preconditioned Conjugate Gradient Analysis Methods. International Journal for Numerical Methods in Engineering, v. 37, i. 6, p. 943-964, March/1994.
- SEPÚLVEDA**, A.; **CASSIS**, J. H. An Efficient Algorithm for the Optimum Design of Trusses with Discrete Variables. International Journal for Numerical Methods in Engineering; v. 23, p. 1111-1130, 1986.
- SHEWCHUK**, J. R. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain. Carnegie Mellon University, Pittsburgh, 1994.
- SILVA**, Renato A. Otimização Estrutural Sujeita à Restrições Dinâmicas. Tese – COPPE/UFRJ, Rio de Janeiro, 1992.

- STANDER**, Nielsen; **SNYMAN**, J. A. *New First-order Interior Feasible Direction Method for Structural Optimization*. International Journal for Numerical Methods in Engineering, v. 36, i. 23, p. 4009-4025, 1993.
- VANDERPLAATS**, Garret. N.; **MOSES**, Fred. *Structural Optimization by Methods of Feasible Directions*. Computers and Structures, vol. 3, i. 4, p. 105-132, July/1973.
- VECCI**, Marco A. M.; **LAU**, Valentino. *Otimização de Configuração de Treliças Sujeitas a Restrições de Alinhamento de Nós*. XV Congresso Brasileiro de Engenharia Mecânica, Águas de Lindóia, São Paulo, 1999.
- VENÂNCIO Fº**, Fernando. *Análise Matricial de Estruturas*. Ed. Almeida Neves, Rio de Janeiro, 1975.
- VISWANATHAN**, Jagadisan; **GROSSMANN**, Ignacio E. *A Combined Penalty Function and Outer Approximation Method for MINLP Optimization*. Computers and Chemical Engineering, 14, 1990, pp. 769-782.
- YAM**, L. H.; **LI**, K. Y. *On Optimality conditions for Least-weight Truss Layouts*. Structural Optimization, v. 16, i. 2-3, p. 186-192, October/1998.

## **Anexos**

---

## AN-1. Tabela de Perfis Disponíveis

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 1.1    | 1.051            | 0.114             | 0.336                      | 0.0374                        | 0.3337               | 26.7      | 2.9       | 2.17                    |
| 1.2    | 1.051            | 0.154             | 0.433                      | 0.0449                        | 0.3220               | 26.7      | 3.9       | 2.79                    |
| 2.1    | 1.315            | 0.134             | 0.497                      | 0.0877                        | 0.4203               | 33.4      | 3.4       | 3.20                    |
| 2.2    | 1.315            | 0.181             | 0.645                      | 0.1063                        | 0.4060               | 33.4      | 4.6       | 4.16                    |
| 2.3    | 1.315            | 0.252             | 0.841                      | 0.1255                        | 0.3862               | 33.4      | 6.4       | 5.43                    |
| 3.1    | 1.500            | 0.118             | 0.513                      | 0.1233                        | 0.4904               | 38.1      | 3.0       | 3.31                    |
| 3.2    | 1.500            | 0.197             | 0.806                      | 0.1750                        | 0.4660               | 38.1      | 5.0       | 5.20                    |
| 4.1    | 1.661            | 0.122             | 0.590                      | 0.1759                        | 0.5460               | 42.2      | 3.1       | 3.81                    |
| 4.2    | 1.661            | 0.142             | 0.677                      | 0.1970                        | 0.5396               | 42.2      | 3.6       | 4.37                    |
| 4.3    | 1.661            | 0.161             | 0.761                      | 0.2164                        | 0.5334               | 42.2      | 4.1       | 4.91                    |
| 4.4    | 1.661            | 0.193             | 0.890                      | 0.2440                        | 0.5237               | 42.2      | 4.9       | 5.74                    |
| 4.5    | 1.661            | 0.252             | 1.116                      | 0.2859                        | 0.5062               | 42.2      | 6.4       | 7.20                    |
| 4.6    | 1.661            | 0.382             | 1.535                      | 0.3421                        | 0.4721               | 42.2      | 9.7       | 9.90                    |
| 5.1    | 1.902            | 0.146             | 0.804                      | 0.3118                        | 0.6229               | 48.3      | 3.7       | 5.18                    |
| 5.2    | 1.902            | 0.201             | 1.073                      | 0.3933                        | 0.6055               | 48.3      | 5.1       | 6.92                    |
| 5.3    | 1.902            | 0.280             | 1.424                      | 0.4824                        | 0.5819               | 48.3      | 7.1       | 9.19                    |
| 5.4    | 1.902            | 0.402             | 1.892                      | 0.5704                        | 0.5490               | 48.3      | 10.2      | 12.21                   |
| 6.1    | 2.374            | 0.126             | 0.890                      | 0.5638                        | 0.7960               | 60.3      | 3.2       | 5.74                    |
| 6.2    | 2.374            | 0.142             | 0.994                      | 0.6216                        | 0.7908               | 60.3      | 3.6       | 6.41                    |
| 6.3    | 2.374            | 0.154             | 1.071                      | 0.6633                        | 0.7869               | 60.3      | 3.9       | 6.91                    |
| 6.4    | 2.374            | 0.173             | 1.198                      | 0.7296                        | 0.7805               | 60.3      | 4.4       | 7.73                    |

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 6.5    | 2.374            | 0.189             | 1.297                      | 0.7800                        | 0.7754               | 60.3      | 4.8       | 8.37                    |
| 6.6    | 2.374            | 0.217             | 1.468                      | 0.8625                        | 0.7666               | 60.3      | 5.5       | 9.47                    |
| 6.7    | 2.374            | 0.252             | 1.680                      | 0.9589                        | 0.7555               | 60.3      | 6.4       | 10.84                   |
| 6.8    | 2.374            | 0.280             | 1.839                      | 1.0266                        | 0.7471               | 60.3      | 7.1       | 11.87                   |
| 6.9    | 2.374            | 0.343             | 2.186                      | 1.1598                        | 0.7284               | 60.3      | 8.7       | 14.10                   |
| 6.10   | 2.374            | 0.437             | 2.659                      | 1.3107                        | 0.7020               | 60.3      | 11.1      | 17.16                   |
| 7.1    | 2.874            | 0.126             | 1.088                      | 1.0289                        | 0.9726               | 73        | 3.2       | 7.02                    |
| 7.2    | 2.874            | 0.142             | 1.217                      | 1.1383                        | 0.9673               | 73        | 3.6       | 7.85                    |
| 7.3    | 2.874            | 0.157             | 1.344                      | 1.2439                        | 0.9621               | 73        | 4         | 8.67                    |
| 7.4    | 2.874            | 0.173             | 1.470                      | 1.3457                        | 0.9568               | 73        | 4.4       | 9.48                    |
| 7.5    | 2.874            | 0.189             | 1.594                      | 1.4437                        | 0.9517               | 73        | 4.8       | 10.28                   |
| 7.6    | 2.874            | 0.205             | 1.717                      | 1.5380                        | 0.9465               | 73        | 5.2       | 11.08                   |
| 7.7    | 2.874            | 0.217             | 1.808                      | 1.6065                        | 0.9427               | 73        | 5.5       | 11.66                   |
| 7.8    | 2.874            | 0.252             | 2.076                      | 1.8002                        | 0.9313               | 73        | 6.4       | 13.39                   |
| 7.9    | 2.874            | 0.276             | 2.250                      | 1.9201                        | 0.9238               | 73        | 7         | 14.51                   |
| 7.10   | 2.874            | 0.374             | 2.938                      | 2.3463                        | 0.8937               | 73        | 9.5       | 18.95                   |
| 7.11   | 2.874            | 0.551             | 4.022                      | 2.8655                        | 0.8440               | 73        | 14        | 25.95                   |
| 8.1    | 3.500            | 0.142             | 1.495                      | 2.1118                        | 1.1884               | 88.9      | 3.6       | 9.65                    |
| 8.2    | 3.500            | 0.157             | 1.654                      | 2.3146                        | 1.1831               | 88.9      | 4         | 10.67                   |
| 8.3    | 3.500            | 0.173             | 1.810                      | 2.5114                        | 1.1778               | 88.9      | 4.4       | 11.68                   |
| 8.4    | 3.500            | 0.189             | 1.966                      | 2.7025                        | 1.1725               | 88.9      | 4.8       | 12.68                   |
| 8.5    | 3.500            | 0.217             | 2.234                      | 3.0232                        | 1.1634               | 88.9      | 5.5       | 14.41                   |
| 8.6    | 3.500            | 0.252             | 2.571                      | 3.4109                        | 1.1518               | 88.9      | 6.4       | 16.59                   |
| 8.7    | 3.500            | 0.280             | 2.828                      | 3.6941                        | 1.1429               | 88.9      | 7.1       | 18.25                   |
| 8.8    | 3.500            | 0.299             | 3.009                      | 3.8868                        | 1.1366               | 88.9      | 7.6       | 19.41                   |
| 8.9    | 3.500            | 0.437             | 4.205                      | 5.0320                        | 1.0939               | 88.9      | 11.1      | 27.13                   |
| 8.10   | 3.500            | 0.598             | 5.455                      | 5.9850                        | 1.0475               | 88.9      | 15.2      | 35.19                   |
| 9.1    | 4.000            | 0.142             | 1.718                      | 3.2010                        | 1.3650               | 101.6     | 3.6       | 11.08                   |
| 9.2    | 4.000            | 0.157             | 1.901                      | 3.5145                        | 1.3597               | 101.6     | 4         | 12.26                   |
| 9.3    | 4.000            | 0.173             | 2.083                      | 3.8200                        | 1.3544               | 101.6     | 4.4       | 13.44                   |
| 9.4    | 4.000            | 0.189             | 2.263                      | 4.1177                        | 1.3491               | 101.6     | 4.8       | 14.60                   |
| 9.5    | 4.000            | 0.224             | 2.662                      | 4.7598                        | 1.3372               | 101.6     | 5.7       | 17.17                   |
| 9.6    | 4.000            | 0.252             | 2.967                      | 5.2333                        | 1.3281               | 101.6     | 6.4       | 19.14                   |
| 9.7    | 4.000            | 0.280             | 3.267                      | 5.6849                        | 1.3191               | 101.6     | 7.1       | 21.08                   |
| 9.8    | 4.000            | 0.319             | 3.688                      | 6.2935                        | 1.3063               | 101.6     | 8.1       | 23.79                   |

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 10.1   | 4.500            | 0.142             | 1.941                      | 4.6124                        | 1.5417               | 114.3     | 3.6       | 12.52                   |
| 10.2   | 4.500            | 0.157             | 2.148                      | 5.0709                        | 1.5363               | 114.3     | 4         | 13.86                   |
| 10.3   | 4.500            | 0.173             | 2.355                      | 5.5191                        | 1.5310               | 114.3     | 4.4       | 15.19                   |
| 10.4   | 4.500            | 0.189             | 2.559                      | 5.9572                        | 1.5256               | 114.3     | 4.8       | 16.51                   |
| 10.5   | 4.500            | 0.205             | 2.763                      | 6.3854                        | 1.5203               | 114.3     | 5.2       | 17.82                   |
| 10.6   | 4.500            | 0.220             | 2.964                      | 6.8038                        | 1.5150               | 114.3     | 5.6       | 19.12                   |
| 10.7   | 4.500            | 0.236             | 3.164                      | 7.2126                        | 1.5098               | 114.3     | 6         | 20.41                   |
| 10.8   | 4.500            | 0.252             | 3.363                      | 7.6119                        | 1.5045               | 114.3     | 6.4       | 21.69                   |
| 10.9   | 4.500            | 0.280             | 3.706                      | 8.2884                        | 1.4954               | 114.3     | 7.1       | 23.91                   |
| 10.10  | 4.500            | 0.311             | 4.093                      | 9.0274                        | 1.4851               | 114.3     | 7.9       | 26.41                   |
| 10.11  | 4.500            | 0.339             | 4.426                      | 9.6453                        | 1.4761               | 114.3     | 8.6       | 28.56                   |
| 10.12  | 4.500            | 0.437             | 5.578                      | 11.6435                       | 1.4448               | 114.3     | 11.1      | 35.99                   |
| 10.13  | 4.500            | 0.531             | 6.626                      | 13.2789                       | 1.4156               | 114.3     | 13.5      | 42.75                   |
| 10.14  | 4.500            | 0.673             | 8.094                      | 15.2742                       | 1.3737               | 114.3     | 17.1      | 52.22                   |
| 11.1   | 5.563            | 0.189             | 3.190                      | 11.5319                       | 1.9012               | 141.3     | 4.8       | 20.58                   |
| 11.2   | 5.563            | 0.220             | 3.700                      | 13.2249                       | 1.8905               | 141.3     | 5.6       | 23.87                   |
| 11.3   | 5.563            | 0.260             | 4.329                      | 15.2550                       | 1.8772               | 141.3     | 6.6       | 27.93                   |
| 11.4   | 5.563            | 0.280             | 4.640                      | 16.2351                       | 1.8706               | 141.3     | 7.1       | 29.93                   |
| 11.5   | 5.563            | 0.311             | 5.132                      | 17.7558                       | 1.8601               | 141.3     | 7.9       | 33.11                   |
| 11.6   | 5.563            | 0.343             | 5.618                      | 19.2194                       | 1.8497               | 141.3     | 8.7       | 36.24                   |
| 11.7   | 5.563            | 0.350             | 5.738                      | 19.5766                       | 1.8471               | 141.3     | 8.9       | 37.02                   |
| 11.8   | 5.563            | 0.374             | 6.097                      | 20.6274                       | 1.8393               | 141.3     | 9.5       | 39.34                   |
| 11.9   | 5.563            | 0.500             | 7.953                      | 25.7316                       | 1.7987               | 141.3     | 12.7      | 51.31                   |
| 11.10  | 5.563            | 0.626             | 9.709                      | 30.0567                       | 1.7595               | 141.3     | 15.9      | 62.64                   |
| 11.11  | 5.563            | 0.752             | 11.365                     | 33.6864                       | 1.7216               | 141.3     | 19.1      | 73.33                   |
| 12.1   | 6.626            | 0.189             | 3.822                      | 19.8104                       | 2.2768               | 168.3     | 4.8       | 24.66                   |
| 12.2   | 6.626            | 0.205             | 4.130                      | 21.3074                       | 2.2714               | 168.3     | 5.2       | 26.64                   |
| 12.3   | 6.626            | 0.220             | 4.437                      | 22.7819                       | 2.2660               | 168.3     | 5.6       | 28.62                   |
| 12.4   | 6.626            | 0.252             | 5.046                      | 25.6640                       | 2.2553               | 168.3     | 6.4       | 32.55                   |
| 12.5   | 6.626            | 0.280             | 5.573                      | 28.1138                       | 2.2460               | 168.3     | 7.1       | 35.96                   |
| 12.6   | 6.626            | 0.311             | 6.170                      | 30.8331                       | 2.2354               | 168.3     | 7.9       | 39.81                   |
| 12.7   | 6.626            | 0.343             | 6.761                      | 33.4681                       | 2.2248               | 168.3     | 8.7       | 43.62                   |
| 12.8   | 6.626            | 0.374             | 7.346                      | 36.0207                       | 2.2144               | 168.3     | 9.5       | 47.39                   |
| 12.9   | 6.626            | 0.433             | 8.426                      | 40.5903                       | 2.1949               | 168.3     | 11        | 54.36                   |
| 12.10  | 6.626            | 0.500             | 9.623                      | 45.4403                       | 2.1731               | 168.3     | 12.7      | 62.08                   |

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 12.11  | 6.626            | 0.563             | 10.724                     | 49.6995                       | 2.1528               | 168.3     | 14.3      | 69.18                   |
| 12.12  | 6.626            | 0.626             | 11.800                     | 53.6758                       | 2.1328               | 168.3     | 15.9      | 76.13                   |
| 12.13  | 6.626            | 0.720             | 13.367                     | 59.1380                       | 2.1034               | 168.3     | 18.3      | 86.24                   |
| 12.14  | 6.626            | 0.752             | 13.877                     | 60.8309                       | 2.0937               | 168.3     | 19.1      | 89.53                   |
| 12.15  | 6.626            | 0.866             | 15.673                     | 66.4648                       | 2.0593               | 168.3     | 22        | 101.12                  |
| 12.16  | 6.626            | 0.874             | 15.794                     | 66.8255                       | 2.0570               | 168.3     | 22.2      | 101.90                  |
| 13.1   | 8.626            | 0.252             | 6.629                      | 58.1568                       | 2.9620               | 219.1     | 6.4       | 42.77                   |
| 13.2   | 8.626            | 0.276             | 7.230                      | 63.0838                       | 2.9539               | 219.1     | 7         | 46.64                   |
| 13.3   | 8.626            | 0.311             | 8.125                      | 70.3139                       | 2.9418               | 219.1     | 7.9       | 52.42                   |
| 13.4   | 8.626            | 0.323             | 8.421                      | 72.6816                       | 2.9378               | 219.1     | 8.2       | 54.33                   |
| 13.5   | 8.626            | 0.343             | 8.913                      | 76.5814                       | 2.9311               | 219.1     | 8.7       | 57.51                   |
| 13.6   | 8.626            | 0.374             | 9.696                      | 82.7016                       | 2.9205               | 219.1     | 9.5       | 62.56                   |
| 13.7   | 8.626            | 0.406             | 10.472                     | 88.6766                       | 2.9099               | 219.1     | 10.3      | 67.56                   |
| 13.8   | 8.626            | 0.437             | 11.243                     | 94.5090                       | 2.8994               | 219.1     | 11.1      | 72.53                   |
| 13.9   | 8.626            | 0.500             | 12.764                     | 105.7545                      | 2.8784               | 219.1     | 12.7      | 82.35                   |
| 13.10  | 8.626            | 0.563             | 14.261                     | 116.4563                      | 2.8576               | 219.1     | 14.3      | 92.01                   |
| 13.11  | 8.626            | 0.594             | 15.000                     | 121.6089                      | 2.8473               | 219.1     | 15.1      | 96.77                   |
| 13.12  | 8.626            | 0.626             | 15.733                     | 126.6322                      | 2.8371               | 219.1     | 15.9      | 101.50                  |
| 13.13  | 8.626            | 0.720             | 17.894                     | 140.9482                      | 2.8066               | 219.1     | 18.3      | 115.44                  |
| 13.14  | 8.626            | 0.752             | 18.601                     | 145.4760                      | 2.7966               | 219.1     | 19.1      | 120.01                  |
| 13.15  | 8.626            | 0.811             | 19.912                     | 153.6477                      | 2.7778               | 219.1     | 20.6      | 128.46                  |
| 13.16  | 8.626            | 0.874             | 21.285                     | 161.9202                      | 2.7581               | 219.1     | 22.2      | 137.32                  |
| 13.17  | 8.626            | 0.906             | 21.963                     | 165.8896                      | 2.7483               | 219.1     | 23        | 141.70                  |
| 13.18  | 8.626            | 1.000             | 23.958                     | 177.1544                      | 2.7193               | 219.1     | 25.4      | 154.57                  |
| 14.1   | 10.748           | 0.252             | 8.308                      | 114.4816                      | 3.7120               | 273       | 6.4       | 53.60                   |
| 14.2   | 10.748           | 0.280             | 9.193                      | 126.0224                      | 3.7025               | 273       | 7.1       | 59.31                   |
| 14.3   | 10.748           | 0.307             | 10.073                     | 137.3775                      | 3.6930               | 273       | 7.8       | 64.99                   |
| 14.4   | 10.748           | 0.343             | 11.197                     | 151.7072                      | 3.6809               | 273       | 8.7       | 72.24                   |
| 14.5   | 10.748           | 0.366             | 11.942                     | 161.0936                      | 3.6728               | 273       | 9.3       | 77.04                   |
| 14.6   | 10.748           | 0.437             | 14.156                     | 188.4664                      | 3.6488               | 273       | 11.1      | 91.33                   |
| 14.7   | 10.748           | 0.500             | 16.098                     | 211.8282                      | 3.6275               | 273       | 12.7      | 103.86                  |
| 14.8   | 10.748           | 0.563             | 18.014                     | 234.3016                      | 3.6065               | 273       | 14.3      | 116.22                  |
| 14.9   | 10.748           | 0.594             | 18.963                     | 245.2123                      | 3.5960               | 273       | 15.1      | 122.34                  |
| 14.10  | 10.748           | 0.626             | 19.906                     | 255.9094                      | 3.5855               | 273       | 15.9      | 128.42                  |
| 14.11  | 10.748           | 0.720             | 22.697                     | 286.7472                      | 3.5544               | 273       | 18.3      | 146.43                  |

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 14.12  | 10.748           | 0.811             | 25.319                     | 314.5891                      | 3.5249               | 273       | 20.6      | 163.35                  |
| 14.13  | 10.748           | 0.843             | 26.218                     | 323.8928                      | 3.5148               | 273       | 21.4      | 169.15                  |
| 14.14  | 10.748           | 0.874             | 27.112                     | 333.0045                      | 3.5046               | 273       | 22.2      | 174.92                  |
| 14.15  | 10.748           | 0.937             | 28.881                     | 350.6626                      | 3.4845               | 273       | 23.8      | 186.33                  |
| 14.16  | 10.748           | 0.961             | 29.537                     | 357.0935                      | 3.4770               | 273       | 24.4      | 190.56                  |
| 14.17  | 10.748           | 1.126             | 34.037                     | 399.3020                      | 3.4251               | 273       | 28.6      | 219.59                  |
| 14.18  | 10.748           | 1.252             | 37.350                     | 428.3197                      | 3.3864               | 273       | 31.8      | 240.97                  |
| 15.1   | 12.748           | 0.280             | 10.949                     | 212.8854                      | 4.4094               | 323.8     | 7.1       | 70.64                   |
| 15.2   | 12.748           | 0.311             | 12.152                     | 235.1108                      | 4.3985               | 323.8     | 7.9       | 78.40                   |
| 15.3   | 12.748           | 0.331             | 12.901                     | 248.8269                      | 4.3917               | 323.8     | 8.4       | 83.23                   |
| 15.4   | 12.748           | 0.343             | 13.349                     | 256.9925                      | 4.3877               | 323.8     | 8.7       | 86.12                   |
| 15.5   | 12.748           | 0.374             | 14.540                     | 278.5341                      | 4.3769               | 323.8     | 9.5       | 93.80                   |
| 15.6   | 12.748           | 0.406             | 15.724                     | 299.7391                      | 4.3661               | 323.8     | 10.3      | 101.44                  |
| 15.7   | 12.748           | 0.437             | 16.902                     | 320.6111                      | 4.3553               | 323.8     | 11.1      | 109.04                  |
| 15.8   | 12.748           | 0.500             | 19.239                     | 361.3699                      | 4.3339               | 323.8     | 12.7      | 124.12                  |
| 15.9   | 12.748           | 0.563             | 21.552                     | 400.8382                      | 4.3127               | 323.8     | 14.3      | 139.04                  |
| 15.10  | 12.748           | 0.626             | 23.839                     | 439.0438                      | 4.2915               | 323.8     | 15.9      | 153.80                  |
| 15.11  | 12.748           | 0.689             | 26.102                     | 476.0136                      | 4.2705               | 323.8     | 17.5      | 168.40                  |
| 15.12  | 12.748           | 0.752             | 28.339                     | 511.7748                      | 4.2496               | 323.8     | 19.1      | 182.83                  |
| 15.13  | 12.748           | 0.811             | 30.414                     | 544.2269                      | 4.2301               | 323.8     | 20.6      | 196.22                  |
| 15.14  | 12.748           | 0.843             | 31.512                     | 561.1173                      | 4.2198               | 323.8     | 21.4      | 203.30                  |
| 15.15  | 12.748           | 0.874             | 32.604                     | 577.7217                      | 4.2095               | 323.8     | 22.2      | 210.35                  |
| 15.16  | 12.748           | 0.937             | 34.768                     | 610.0853                      | 4.1889               | 323.8     | 23.8      | 224.31                  |
| 15.17  | 12.748           | 1.000             | 36.908                     | 641.3433                      | 4.1686               | 323.8     | 25.4      | 238.11                  |
| 15.18  | 12.748           | 1.063             | 39.022                     | 671.5212                      | 4.1483               | 323.8     | 27        | 251.75                  |
| 15.19  | 12.748           | 1.126             | 41.112                     | 700.6443                      | 4.1283               | 323.8     | 28.6      | 265.24                  |
| 16.1   | 14.000           | 0.343             | 14.696                     | 342.8706                      | 4.8302               | 355.6     | 8.7       | 94.81                   |
| 16.2   | 14.000           | 0.374             | 16.011                     | 371.8609                      | 4.8193               | 355.6     | 9.5       | 103.29                  |
| 16.3   | 14.000           | 0.406             | 17.319                     | 400.4407                      | 4.8085               | 355.6     | 10.3      | 111.73                  |
| 16.4   | 14.000           | 0.437             | 18.621                     | 428.6140                      | 4.7977               | 355.6     | 11.1      | 120.13                  |
| 16.5   | 14.000           | 0.469             | 19.916                     | 456.3845                      | 4.7870               | 355.6     | 11.9      | 128.49                  |
| 16.6   | 14.000           | 0.500             | 21.206                     | 483.7562                      | 4.7762               | 355.6     | 12.7      | 136.81                  |
| 16.7   | 14.000           | 0.563             | 23.766                     | 537.3183                      | 4.7549               | 355.6     | 14.3      | 153.33                  |
| 16.8   | 14.000           | 0.594             | 25.037                     | 563.5164                      | 4.7442               | 355.6     | 15.1      | 161.53                  |
| 16.9   | 14.000           | 0.626             | 26.301                     | 589.3309                      | 4.7336               | 355.6     | 15.9      | 169.68                  |

| Perfil | Diâmetro<br>(in) | Espessura<br>(in) | Área<br>(in <sup>2</sup> ) | Inércia<br>(in <sup>4</sup> ) | Raio Giração<br>(in) | D<br>(mm) | t<br>(mm) | A<br>(cm <sup>2</sup> ) |
|--------|------------------|-------------------|----------------------------|-------------------------------|----------------------|-----------|-----------|-------------------------|
| 16.10  | 14.000           | 0.689             | 28.811                     | 639.8240                      | 4.7125               | 355.6     | 17.5      | 185.88                  |
| 16.11  | 14.000           | 0.752             | 31.297                     | 688.8276                      | 4.6914               | 355.6     | 19.1      | 201.91                  |
| 16.12  | 14.000           | 0.811             | 33.604                     | 733.4420                      | 4.6718               | 355.6     | 20.6      | 216.80                  |
| 16.13  | 14.000           | 0.874             | 36.041                     | 779.6436                      | 4.6510               | 355.6     | 22.2      | 232.52                  |
| 16.14  | 14.000           | 0.937             | 38.453                     | 824.4416                      | 4.6303               | 355.6     | 23.8      | 248.09                  |
| 16.15  | 14.000           | 1.000             | 40.841                     | 867.8650                      | 4.6098               | 355.6     | 25.4      | 263.49                  |

Aço ASTM-36: Módulo de Elasticidade  $E = 29733 \text{ ksi}$

Densidade do aço  $\rho = 0.28 \text{ pci}$

## **AN-2. GAMS – Uma Noção**

### **AN-2.1. Introdução**

Durante as décadas de 50 e 60 foram feitos progressos substanciais no desenvolvimento de códigos computacionais para resolver grandes e complexos problemas de programação matemática, entretanto a década de 70 não manteve a escrita em virtude dos procedimentos formarem apenas uma pequena e dispendiosa parte do processo de modelagem. Cada modelo requeria muito tempo dos analistas e de programação para organizar os dados e escrever programas para transformar esses dados na forma exigida por cada otimizador de programação matemática; além disso, era difícil detectar e eliminar erros, pois os programas que realizavam as operações com os dados eram acessíveis apenas aos especialistas que o escreveram e não aos analistas responsáveis pelo projeto.

Desta forma, o *GAMS (General Algebraic Modelling System)* foi desenvolvido para fazer a construção e resolução de modelos matemáticos amplos e complexos, de quaisquer áreas afins, de forma mais direta para programadores e mais inteligível para usuários e modelistas, permitindo

realizar mudanças na especificação dos modelos de forma simples e segura, relações algébricas enunciadas de forma não ambígua e descrições de modelos independentes dos algoritmos de solução, além de tornar mais fácil a localização de erros dentro do programa.

O projeto do *GAMS* incorporou idéias tiradas da teoria dos bancos de dados relacionais e programação matemática, ajustando essas idéias às necessidades dos modeladores. Os bancos de dados relacionais fornecem uma estrutura para se desenvolver a organização dos dados e a capacidade para manipulação destes. A programação matemática fornece uma maneira de descrever o problema, bem como os métodos para resolvê-lo. Esta combinação permite separar os dados da lógica de solução, podendo o problema crescer em tamanho (aumento do número de dados) sem que haja um aumento na complexidade da representação.

Um problema de otimização modelado em *GAMS* é feito de forma que qualquer pessoa possa facilmente lê-lo e eventualmente alterar dados ou corrigi-lo para adequá-lo a uma nova necessidade, por exemplo. Assim, a codificação é concisa, a entrada de dados é feita direto na modelagem e de forma bem elementar, as transformações destes dados são feitas de forma algébrica; em suma, toda informação necessária para se entender o modelo está disponível em um único documento [BROOKE et al., 1997].

## **AN-2.2. Tutorial *GAMS***

De uma maneira geral, a terminologia adotada pela linguagem *GAMS* é feita de tal forma que índices são denominados *SETS*; dados são

chamados de PARAMETERS; variáveis de decisão são chamadas de VARIABLES; e as restrições do problema e a função objetivo são apresentadas como EQUATIONS. Outro princípio observado reza que todas as entidades do modelo são identificadas e agrupadas pelo seu tipo e que a ordem das entidades é escolhida de modo que nenhuma delas é mencionada antes de ter sido definida.

### **AN-2.2.1. Estrutura**

Os componentes básicos de um modelo GAMS são os seguintes:

- SETS
  - Declaração dos índices
  - Valoração dos membros
- Dados (PARAMETERS, TABLES, SCALARS)
  - Declaração
  - Atribuição de Valores
- VARIABLES
  - Declaração
  - Atribuição do Tipo
  - Atribuição de limites e/ou valores iniciais
- EQUATIONS
  - Declaração
  - Definição
- Enunciados pertinentes a SOLVE e MODEL
- Enunciados pertinentes a DISPLAY (opcionais)

O arquivo gerado pelo compilador apresenta a seguinte estrutura:

- Impressão *echo*
- Mapas de referência
- Listagem de equações
- Relatos do *status*
- Resultados

Para montagem do código, algumas observações de caráter geral devem ser consideradas:

- Os enunciados podem ser dispostos em linhas múltiplas entremeadas com linhas em branco, podem ser múltiplos por linha e receber comentários de maneira análoga.
- Após todos os enunciados são colocados ponto e vírgulas (;).
- O compilador *GAMS* não distingue entre letras maiúsculas e minúsculas; por uma questão estética, recomenda-se o uso de letras maiúsculas para as palavras que fazem parte da linguagem.
- Recomenda-se também fazer uma documentação a respeito do que está sendo elaborado, ou seja, um cabeçalho cujas linhas devem começar por asteriscos (que são ignoradas pelo compilador).
- A criação de entidades (variáveis de decisão, equações de restrição, matrizes, parâmetros, entre outras) é feita sempre em dois estágios: primeiro a declaração (dar um nome a entidade), depois definir ou atribuir valor(es). Todo nome de entidade deve obrigatoriamente começar por uma letra, podendo ser seguida de outras letras ou dígitos.

## AN-2.2.2. Conjuntos

*SETS* correspondem a índices na representação algébrica de um modelo *GAMS*. Observa-se que *GAMS* usa barras (/) ao invés de chaves para a listagem dos elementos de um conjunto.

```
SET I      índices de membros      /1 * 5/  
      J      índices de nos        /1, 2, 3, 4/;
```

As palavras minúsculas introduzidas entre os nomes dos *SETS* e a definição dos respectivos conjuntos são denominados texto. Ignorados pelo compilador, servem como documentação interna ao modelo.

Um aspecto útil aparece quando se deseja definir um *SET* com uma seqüência de elementos mais longa; para tanto, usa-se o asterisco conforme no exemplo acima.

## AN-2.2.3. Entrada de Dados

O modelo *GAMS* apresenta três modos para a entrada de dados: listas, tabelas e valoração direta.

### AN-2.2.3.1. Listas

```
PARAMETER
XI(I)      coordenada x do no inicial da barra i
           /1      0.0
           2      2.0
           3      4.0
           4      2.0/;
```

Observe o seguinte:

- Os pares da atribuição (índice e valoração) podem ser dispostos também em linha, separados por vírgulas, e devem estar delimitados por barras.
- Não existe ponto e vírgula separando a lista dos valores dos elementos do nome, domínio e do texto precedente; isso porque o mesmo enunciado está sendo usado para declarações e atribuições quando são usados os formatos de listas.
- O compilador verifica se cada elemento do domínio é de fato um membro apropriado, ou seja, devidamente declarado.
- Por *default*, os valores não declarados na lista ficam automaticamente valorados pelo zero.
- Um escalar pode ser visto como um parâmetro de domínio vazio; podemos então declará-lo como um enunciado *SCALAR* contendo uma lista de apenas um valor:

```
SCALAR E      modulo de elasticidade longitudinal      /10000/;
```

### AN-2.2.3.2. Tabelas

```
TABLE COORD(I,J)      coordenadas x e y dos nos
      x      y
1      0.0      0.0
2      2.0      0.0
3      4.0      0.0
4      2.0      1.5;
```

O efeito deste tipo de enunciado é declarar o parâmetro COORD e especificar seu domínio como o conjunto de pares do produto cartesiano I (representando os nós) e J (representando x e y) se porventura existirem entradas em branco na tabela, elas serão interpretadas como zeros.

### AN-2.2.3.3. Atribuição direta

```
PARAMETER
      SIGMA(I) tensao admissivel do metal utilizado;
      SIGMA(I) = F * S(I);
```

Se o mesmo parâmetro for atribuído mais de uma vez, a última atribuição tem primazia sobre todas as anteriores; no entanto, um mesmo parâmetro não pode ser declarado mais de uma vez.

O lado direito dos enunciados, que atribuem valores para os parâmetros, possuem uma gama bastante variada de expressões matemáticas e funções pré-construídas.

## AN-2.2.4. Variáveis

As variáveis de decisão de um modelo expresso em GAMS devem ser declaradas com o identificador VARIABLES.

```
VARIABLES      X(I,J)      coordenadas otimizadas dos nos
                Z          peso otimo da trelica;
```

Uma vez declarada, a cada variável deve ser atribuído um “tipo” (Tabela AN-2.1). A variável  $z$  é declarada sem nenhum domínio porque é uma quantidade escalar (domínio vazio) e do tipo *Free*. Todo modelo de otimização em GAMS deve obrigatoriamente conter uma tal variável para servir de quantidade a ser otimizada (no caso, minimizada). De maneira geral, se desejarmos que a variável  $X(I, J)$  seja vinculada a assumir somente valores positivos, poderemos declará-la do seguinte modo:

```
POSITIVE VARIABLE X;
```

Observe que o domínio de  $X$  não deve ser repetido no enunciado do tipo da variável.

Tabela AN-2.1 – Domínios de variáveis em GAMS

| Tipo                  | Valores permitidos      |
|-----------------------|-------------------------|
| <i>Free (default)</i> | $-\infty$ até $+\infty$ |
| POSITIVE              | 0 até $+\infty$         |
| NEGATIVE              | $-\infty$ até 0         |
| BINARY                | 0 ou 1                  |
| INTEGER               | 0, 1, 2, 3, ..., 100    |

*GAMS* foi projetado com um banco de dados adicional cujos registros, subdivididos em quatro campos, são mantidos para as variáveis e equações:

|     |                 |
|-----|-----------------|
| .LO | Limite inferior |
| .L  | Valor primário  |
| .UP | Limite superior |
| .M  | Valor marginal  |

O formato utilizado para referenciar essas quantidades é a variável seguida do nome do campo e do domínio (se necessário). Os limites inferior e superior de uma variável são colocados automaticamente de acordo com o tipo de variável em questão, mas seus valores podem ser sobrescritos pelo usuário conforme a necessidade. É muito útil quando se trabalha com programação não-linear especificar os limites inferiores e superiores das variáveis envolvidas, pois ajuda o *solver* a estreitar o intervalo de busca para a solução ótima.

```
X(I,J).UP = 5.0;  
AREA(K).LO = 25.0;
```

### **AN-2.2.5. Equações**

As equações devem ser declaradas em enunciados separados. O formato é análogo às outras declarações em *GAMS*: primeiro vem a palavra-chave *EQUATIONS*, em seguida do nome e texto explanatório, se convier.

## EQUATIONS

WEIGHT            define a funcao objetivo  
EULER(I)          limita a tensao atuante;

Em *GAMS*, a palavra *EQUATION* contabiliza além de equações propriamente ditas, também desigualdades.

Após declaradas as equações, faz-se as suas definições; é nessa etapa onde os enunciados aparecem mais complexos em termos de variedades. Os componentes são dados pela seguinte ordem:

- i) O nome da equação a ser definida
- ii) O domínio
- iii) O símbolo “..”
- iv) A expressão do lado esquerdo da equação
- v) O operador relacional
- vi) A expressão do lado direito da equação.

```
WEIGHT..        SUM((J), PESO(J) * COMPR(J) * AREA(J)) =E= Z;  
EULER(J,K)..   ((PI * PI * E(J) * IZ(J)) /  
                  (AREA(J) * COMPR(J) * COMPR(J)) =L= SIGMA(J);
```

Os operadores relacionais são três:

=L=        menor ou igual a  
=G=        maior ou igual a  
=E=        igual a

É importante entender a diferença entre “=” e “=E=”. O primeiro é utilizado nas expressões que alocam diretamente valores às variáveis, enquanto o segundo expressa a igualdade entre os dois termos da equação.

Outros pontos importantes rezam o seguinte: as variáveis podem

aparecer em ambos os lados de uma equação, podendo inclusive serem repetidas; uma equação pode aparecer em qualquer ponto de um modelo GAMS, contanto que suas variáveis e parâmetros tenham sido previamente definidos; e elas não precisam ser definidas na mesma ordem em que são declaradas.

### **Função Objetivo**

GAMS não possui nenhuma entidade explicitamente denominada de “função objetivo”. Para especificar uma função a ser otimizada, basta criar uma variável (sem domínio limitado) e que aparece na definição de uma equação, igualando-a à sua função objetivo.

### **AN-2.2.6. Conjunto de Equações para Solução**

Na modelagem de um problema, pode-se colocar equações refletindo toda e qualquer situação referente a uma estrutura, mesmo que não se utilize todas as variantes possíveis para o cálculo de otimização, como por exemplo: vibrações, combinação diversas de cargas (acidentais, permanentes e/ou eólicas), recalques, variações de temperatura, alinhamento de nós, entre outras.

O projetista pode optar pela montagem de diversos conjuntos de equações utilizando a palavra MODEL. Como toda entidade do GAMS, o formato da declaração é a palavra chave seguida do nome do modelo e de uma lista de equações (separadas por vírgula) entre barras. Se todas as equações definidas devem ser incluídas, podemos entrar com a expressão ALL no lugar da referida lista.

```
MODEL COMB1 /CARGA1, CARGA2, VENTO1/;
MODEL COMB2 /CARGA1, VENTO2, RECALQUE/;
MODEL COMB3 /ALL/;
```

Declarados os modelos, pode-se chamar o otimizador. O formato começa com a palavra-chave SOLVE, o nome do modelo previamente declarado, a segunda palavra-chave USING, um procedimento de resolução (Tabela AN-2.1), a terceira palavras-chave MINIMIZING ou MAXIMIZING, e nome da variável otimizada.

**Tabela AN-2.2 – Procedimentos de resolução em GAMS**

| Procedimento | Significado                        |
|--------------|------------------------------------|
| LP           | Programação linear                 |
| NLP          | Programação Não-linear             |
| MIP          | Programação Inteira Mista          |
| RMIP         | Programação Inteira Mista Relaxada |

```
SOLVE COMB1 USING NLP MINIMIZING Z;
```

Observe que a declaração de modelo, deve obrigatoriamente conter a função objetivo com a variável a ser otimizada. Outro fator é que, independente da quantidade de conjuntos, apenas uma solução pode ser obtida, isto é, o enunciado de outro SOLVE ocasiona erro no compilador. Também nota-se que o nome do *solver* não aparece em parte nenhuma do

código, ele é definido no momento da instalação do GAMS dentro das opções de cada pacote (*MINOS*, *CONOPT*, *LOQO*, *ZOOM*, entre outros); deve-se ter cuidado com a compatibilidade entre o solver utilizado e o procedimento de resolução, conforme a tabela, do enunciado SOLVE.

### **AN-2.2.7. Saída e Exibição de Resultados**

Todas essas operações anteriormente descritas criará a estrutura de dados e gerará os procedimentos específicos a ser otimizado pelo solver escolhido, o *solver* será invocado e a saída será organizada e impressa pelo GAMS em um arquivo de saída. Para se obter os valores ótimos podemos analisar esse arquivo ou utilizar-se da palavra `DISPLAY` para exibi-los de forma simples.

```
DISPLAY X.L, X.M, Z;
```

A variável de projeto `X` seguida de `.L` trará os diversos níveis do valor ótimo. O complemento `.M` trará os valores marginais, ou seja, valores que quando incrementados retorna o aumento de uma unidade no valor da função objetivo (`Z`).

## **AN-3. Norma AISC-ASD/1989**

Este apêndice descreve alguns dos principais detalhes da norma adotada para o dimensionamento ótimo das barras no projeto de treliças metálicas proposto nesta obra. O Código ASD (*Allowable Stress Design*) mais recente é datado de 1989 e foi desenvolvido pelo AISC (*American Institute of Steel Constructions*).

As unidades utilizadas no equacionamento são as unidades dimensionais britânicas, portanto todas as equações e descrições aqui apresentadas correspondem às unidades *kip-Inch-Second* do sistema inglês.

### **AN-3.1. Classificação das Seções**

As tensões admissíveis para a compressão axial são calculadas a partir de uma classificação da seção transversal do perfil, como sendo

Compacta, Não-compacta ou Esbelta. A determinação desta classificação e das propriedades da seção requerida é dada pelas Tabelas AN-3.1.a a AN-3.1.f, com referência na Figura AN-3.1, de acordo com um limite para razão entre a largura e a espessura das chapas que formam a seção do perfil.

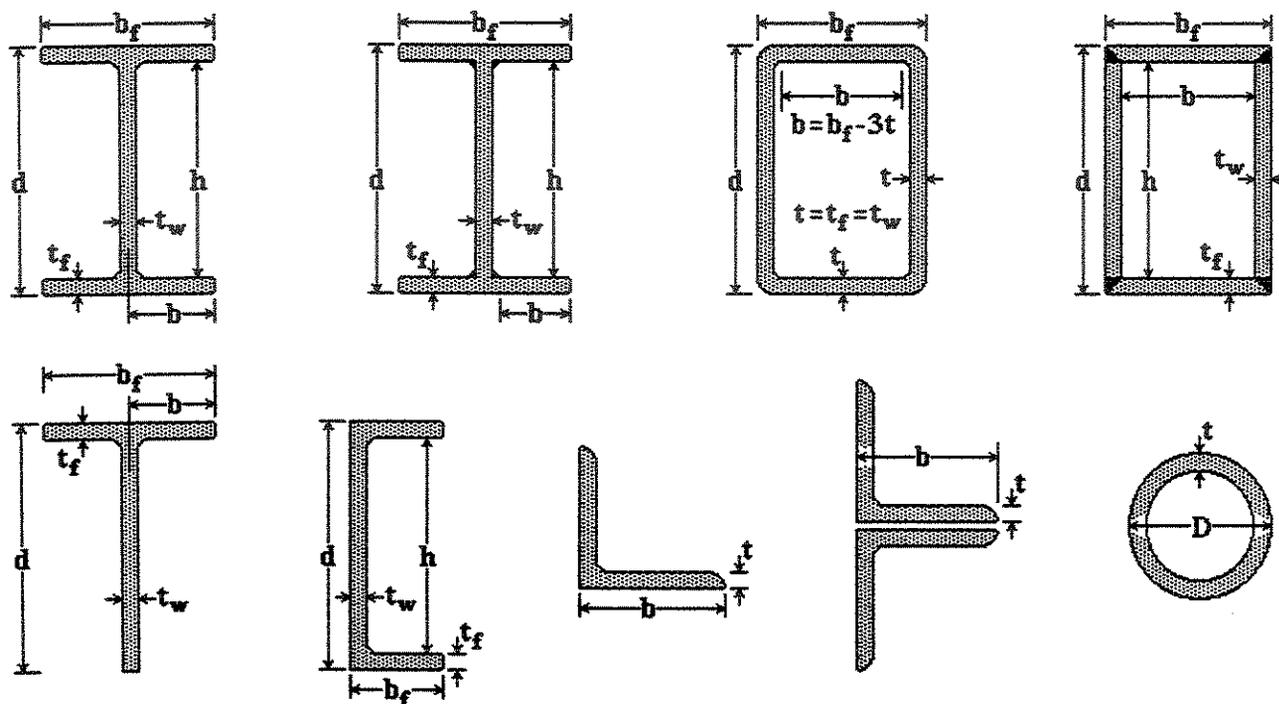


Fig. AN-3.1 – Definição das propriedades geométricas; acima tem-se perfil I laminado e soldado, *box* laminado e soldado; embaixo tem-se perfil T, C (canal), cantoneira simples, cantoneira dupla e perfil tubular.

**Tabela AN-3.1.a – Classificação dos perfis I baseado nos limites da razão largura-espessura.**

| <b>Perfil I (laminado ou soldado)</b> |   |   |   |
|---------------------------------------|---|---|---|
| <b>Verificar razão</b>                | <b>Seção</b>  |   |   |
|                                       | <b>Compacta</b>   | <b>Não-Compacta</b>   | <b>Esbelta</b>  |
| $b_f/2t_f$<br>(laminado)              | $\leq 65/\sqrt{F_y}$  | $\leq 95/\sqrt{F_y}$  | Sem limite  |
| $b_f/2t_f$<br>(soldado)               | $\leq 65/\sqrt{F_y}$  | $\leq 95/\sqrt{F_y/k_c}$  | Sem limite  |
| $d/t_w$                               | $\leq \frac{640}{\sqrt{F_y}} \left( 1 - 3.74 \frac{f_a}{F_y} \right)$<br>se $f_a/F_y \leq 0.16$<br>$\leq 257/\sqrt{F_y}$<br>se $f_a/F_y > 0.16$ | Sem limite  | Sem limite  |
| $h/t_w$                               | Sem limite  | $\leq 253/\sqrt{F_y}$<br>se comprimida (sempre)<br>$\leq 760/\sqrt{F_b}$<br>em caso contrário | $\leq \frac{14000}{\sqrt{F_y(F_y + 16.5)}}$<br>ou<br>$\leq 260$ |

**Tabela AN-3.1.b – Classificação dos perfis *Box* baseado nos limites da razão largura-espessura.**

| <b>Perfil <i>Box</i> (laminado ou soldado)</b> |   |   |   |
|--|---|---|---|
| <b>Verificar razão</b>                         | <b>Seção</b>  |   |   |
|  | <b>Compacta</b>   | <b>Não-Compacta</b>   | <b>Esbelta</b>  |
| $b/t_f$  | $\leq 190/\sqrt{F_y}$   | $\leq 238/\sqrt{F_y}$   | Sem limite  |
| $d/t_w$  | $\leq \frac{640}{\sqrt{F_y}} \left( 1 - 3.74 \frac{f_a}{F_y} \right)$<br>se $f_a/F_y \leq 0.16$<br>$\leq 257/\sqrt{F_y}$<br>se $f_a/F_y > 0.16$ | Sem limite  | Sem limite  |
| $h/t_w$  | Sem limite  | $\leq 253/\sqrt{F_y}$<br>se comprimida (sempre)<br>$\leq 760/\sqrt{F_b}$<br>em caso contrário | $\leq \frac{14000}{\sqrt{F_y(F_y + 16.5)}}$<br>ou<br>$\leq 260$ |
| Outros   | Se $t_w \geq t_f/2$ e $h \leq 6b_f$<br>(para perfil soldado)  | Não se aplica   | Não se aplica   |

**Tabela AN-3.1.c – Classificação dos perfis T baseado nos limites da razão largura-espessura.**

| <b>Perfil T (laminado ou soldado)</b> |                      |                       |  |
|---------------------------------------|----------------------|-----------------------|--|
| <b>Verificar razão</b>                | <b>Seção</b>         |                       |  |
|                                       | <b>Compacta</b>      | <b>Não-Compacta</b>   | <b>Esbelta</b>   |
| $b_f/2t_f$                            | $\leq 65/\sqrt{F_y}$ | $\leq 95/\sqrt{F_y}$  | Sem limite   |
| $d/t_w$                               | Não se aplica        | $\leq 127/\sqrt{F_y}$ | Sem limite   |
| Outras                                | Sem limite           | Sem limite            | $b_f/d_w \geq 0.5$ e $t_f/t_w \geq 1.25$<br>se soldado,<br>$b_f/d_w \geq 0.5$ e $t_f/t_w \geq 1.10$<br>se laminado |

**Tabela AN-3.1.d – Classificação dos perfis C (canal) baseado nos limites da razão largura-espessura.**

| <b>Perfil C (laminado ou soldado)</b> |   |   |  |
|---------------------------------------|---|---|--|
| <b>Verificar razão</b>                | <b>Seção</b>  |   |  |
|                                       | <b>Compacta</b>   | <b>Não-Compacta</b>   | <b>Esbelta</b>   |
| $b/t_f$                               | $\leq 65/\sqrt{F_y}$  | $\leq 95/\sqrt{F_y}$  | Sem limite   |
| $d/t_w$                               | $\leq \frac{640}{\sqrt{F_y}} \left( 1 - 3.74 \frac{f_a}{F_y} \right)$<br>se $f_a/F_y \leq 0.16$<br>$\leq 257/\sqrt{F_y}$<br>se $f_a/F_y > 0.16$ | Sem limite  | Sem limite   |
| $h/t_w$                               | Sem limite  | $\leq 253/\sqrt{F_y}$<br>se comprimida (sempre)<br>$\leq 760/\sqrt{F_b}$<br>em caso contrário | $\leq \frac{14000}{\sqrt{F_y(F_y + 16.5)}}$<br>ou<br>$\leq 260$  |
| Outras                                | Sem limite  | Sem limite  | $b_f/d_w \geq 0.5$ e $t_f/t_w \geq 1.25$<br>se soldado,<br>$b_f/d_w \geq 0.5$ e $t_f/t_w \geq 1.10$<br>se laminado |

**Tabela AN-3.1.e – Classificação dos perfis Cantoneira, simples e duplo (com abas paralelas), baseado nos limites da razões largura-espessura.**

| <b>Perfil Cantoneira (simples e duplo)</b> |                 |                      |                |
|--|-----------------|----------------------|----------------|
| <b>Verificar razão</b>                     | <b>Seção</b>    |                      |                |
|  | <b>Compacta</b> | <b>Não-Compacta</b>  | <b>Esbelta</b> |
| $b/t$                                      | Não se aplica   | $\leq 76/\sqrt{F_y}$ | Sem limite     |

**Tabela AN-3.1.f** – Classificação dos perfis Tubulares baseado nos limites da razões largura-espessura.

| Verificar razão | Perfil Tubular |                 |  |
|-----------------|----------------|-----------------|--|
|                 | Compacta       | Não-Compacta    | Esbelta                                    |
| $D/t$           | Não se aplica  | $\leq 3300/F_y$ | $\leq 13000/F_y$<br>se comprimida (sempre) |

### AN-3.2. Cálculo das Tensões

As tensões nas barras de seções não-esbeltas são calculadas para cada combinação de carregamento baseado nas propriedades da seção transversal bruta.

$$f = \frac{P}{A}$$

Se a seção é esbelta com elementos esbeltos, como as *almas* esbeltas da seção I, C (canal) e O (*box*) ou ainda as *mesas* esbeltas da seção O, o cálculo da tensão é baseado numa seção efetiva, determinada a partir da redução da dimensão da *mesa* e/ou da *alma*.

### **AN-3.3. Cálculo das Tensões Admissíveis**

#### **AN-3.3.1. Tensão Admissível de Tração**

A tensão admissível axial de tração  $F_a$  é assumida como sendo 60% da tensão de escoamento  $F_y$  do aço adotado.

$$F_a = 0.60 \cdot F_y$$

#### **AN-3.3.2. Tensão Admissível de Compressão**

A tensão admissível axial de compressão é o menor valor obtido da flambagem de flexão, determinado de acordo com os seguintes critérios.

Para barras sob compressão, o índice de esbeltez  $i$  tem que ser obrigatoriamente menor ou igual a 200.

$$i = \frac{K \cdot l}{r} \leq 200$$

Deve-se observar que para barras de treliças, onde não se considera a ação dos momentos fletores nas rótulas e todas as ações são axiais, o coeficiente de flambagem  $K$  é igual a 1.0. O raio de giração  $r$  é determinado a partir do menor Momento de Inércia  $J$  da seção transversal

do perfil adotado.

$$r = \sqrt{\frac{J}{A}}$$

A tensão admissível de compressão  $F_a$  depende do Índice de Esbeltez  $i$  e do correspondente valor crítico de esbeltez  $C_c$ , onde

$$i = \max\left\{\frac{K \cdot l}{r_{22}}, \frac{K \cdot l}{r_{33}}\right\} \quad e \quad C_c = \sqrt{\frac{2 \cdot \pi^2 \cdot E}{F_y}}$$

onde  $r_{22}$  é o raio de giração segundo o eixo que passa pelo centro geométrico da seção e é paralelo a alma (perfil I, C e O), ou a maior perna da cantoneira (perfil L), ou ainda paralelo as pernas vizinhas do perfil de cantoneiras dupla; e  $r_{33}$  é o raio de giração segundo um eixo ortogonal ao eixo 2-2.

Para seções compactas e não-compactas, as equações para determinação da tensão admissível de compressão são as mesmas, dadas por

$$F_a = \frac{\left[1.0 - \frac{1}{2} \cdot \left(\frac{i}{C_c}\right)^2\right]}{\frac{5}{3} + \frac{3}{8} \cdot \frac{i}{C_c} - \frac{1}{8} \cdot \left(\frac{i}{C_c}\right)^3} \cdot F_y \quad \text{se } i \leq C_c$$

$$F_a = \frac{12}{23} \cdot \frac{\pi^2 \cdot E}{i^2} \quad \text{se } i > C_c$$

Para os perfis com seções esbeltas, com exceção aos tubos, a tensão admissível de compressão é dado por

$$F_a = Q \cdot \frac{\left[ 1.0 - \frac{1}{2} \cdot \left( \frac{i}{C'_c} \right)^2 \right]}{\frac{5}{3} + \frac{3}{8} \cdot \frac{i}{C'_c} - \frac{1}{8} \cdot \left( \frac{i}{C'_c} \right)^3} \cdot F_y \quad \text{se } i \leq C'_c$$

$$F_a = \frac{12}{23} \cdot \frac{\pi^2 \cdot E}{i^2} \quad \text{se } i > C'_c$$

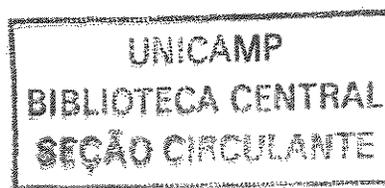
onde  $C'_c = \sqrt{\frac{2 \cdot \pi^2 \cdot E}{Q \cdot F_y}}$  e  $Q = Q_s \cdot Q_a$

Alguns profissionais simplificam a constante  $Q$  das expressões acima adotando o valor de 0,95. Esta constante é fruto do produto de dois fatores:  $Q_s$  é um fator de redução para elementos esbeltos não-enrijecidos; e  $Q_a$  é um fator de redução calculado pela razão entre a área efetiva da seção transversal (descontados chanfros, cortes, furos de parafusos e rebites ao longo do comprimento da barra) e sua área bruta.

Para tubos vazados, a tensão admissível é determinada, independente do índice de esbeltez crítico, por

$$F_a = 662 \cdot \frac{t}{D} + 0.40 \cdot F_y$$

onde  $D$  e  $t$  são o diâmetro e a espessura da parede do tubo, respectivamente.



### **AN-3.4. Considerações sobre a Norma AISC-ASD/1989**

Deve-se salientar que a norma americana para estruturas metálicas, que utiliza o Método das Tensões Admissíveis, não se resume apenas ao exposto neste trabalho. Aqui está descrito o necessário e suficiente para utilizar o programa proposto de otimizar treliças metálicas planas, com carregamento concentrado nos nós.