

SISTEMA PARA ANÁLISE DE ESTRUTURAS  
CONSIDERANDO INTERAÇÃO COM MEIO ELÁSTICO

AUTOR: PROF. MARCIO ANTONIO RAMALHO

ORIENTADOR: PROF. DR. WILSON SERGIO VENTURINI

UNIVERSIDADE DE SÃO PAULO  
ESCOLA DE ENGENHARIA DE SÃO CARLOS  
ÁREA DE ENGENHARIA DE ESTRUTURAS

SISTEMA PARA ANÁLISE DE ESTRUTURAS  
CONSIDERANDO INTERAÇÃO COM MEIO ELÁSTICO

PROF. MARCIO ANTONIO RAMALHO

Tese apresentada à Escola de Engenharia de São Carlos, da Universidade de São Paulo, como parte dos requisitos para a obtenção do título de "Doutor em Engenharia de Estruturas".

ORIENTADOR:

PROF. DR. WILSON SERGIO VENTURINI

SÃO CARLOS, JUNHO DE 1990

Para minhas filhas,

Paula e Júlia

## AGRADECIMENTOS

Agradeço a todas as pessoas que direta ou indiretamente tornaram possível a elaboração desta pesquisa.

Em especial, agradeço a Wilson S. Venturini e Márcio R.S. Corrêa, pela orientação, pelo apoio e pela dedicada amizade, absolutamente fundamentais para a realização deste trabalho.

## RESUMO

Este trabalho trata da montagem de um sistema computacional para análise elástica-linear de grandes estruturas em microcomputadores de 16 bits. É possível considerar-se a interação da estrutura analisada com um meio elástico semi-infinito, através de um procedimento baseado no método dos elementos de contorno. O sistema permite a definição de até 10 000 pontos nodais ou 33 000 graus de liberdade, oferecendo um total de oito diferentes tipos de elemento : treliça, barra, contorno, membrana, plano, sólido, placa/casca e sapata rígida.

No primeiro capítulo, situa-se a pesquisa realizada dentro do contexto dos trabalhos desenvolvidos nesta área até o momento. No segundo, traça-se um panorama geral da organização do programa, com todas as suas etapas de processamento e recursos oferecidos. Já o terceiro capítulo apresenta a biblioteca de elementos baseados no método dos elementos finitos, com a formulação e exemplos de aplicação. Finalmente, o quarto capítulo trata do elemento sapata rígida, desenvolvido com base no método dos elementos de contorno.

## ABSTRACT

This work deals with the elaboration of a computational system for linear elastic analysis of large structures in 16 bit microcomputers. It is possible to take into account the interaction of the structure under analysis with a semi-infinite elastic domain, using a procedure based on the boundary element method. The system allows up to 10.000 nodal points or 33.000 freedom degrees, with eight different types of elements: truss, beam, boundary, membrane, plane, solid, plate bending/shell and rigid footing.

In the first chapter, the work is situated in its context of developments in this area. In the second chapter, a general description of the program organization is made, showing all its processing parts and capabilities. The third chapter gives the element library concerning with the finite element method, with formulation and practical examples. Finally, the fourth chapter deals with the rigid footing element, based on the boundary element method.

## ÍNDICE

### CAPÍTULO 1 - Introdução

<b>1.1. Os Computadores e os Métodos Numéricos</b>	<b>1</b>
<b>1.2. Sistemas Computacionais Para Grandes Computadores</b>	<b>5</b>
1.2.1. Considerações Gerais	5
1.2.2. O Programa ASKA	5
1.2.3. O Programa NASTRAN	7
1.2.4. O Programa STARDYNE	8
1.2.5. O Programa STRUDL	9
1.2.6. O Programa ADINA	10
1.2.7. O Programa LEBRE	11
<b>1.3. Sistemas Computacionais Microcomputadores</b>	<b>13</b>
1.3.1. Considerações Gerais	13
1.3.2. O Programa SAP90	13
1.3.3. O Programa SUPERSAP	15
1.3.4. Comparação com Sistemas Para Grandes Computadores	17
<b>1.4. Objetivos Gerais do Trabalho</b>	<b>19</b>
<b>1.5. Organização do Trabalho</b>	<b>21</b>
<b>1.6. Referências Bibliográficas</b>	<b>23</b>

### CAPÍTULO 2 - Organização Geral do Programa

<b>2.1. Introdução</b>	<b>27</b>
<b>2.2. Conceitos Básicos do FORTRAN 4.01 da Microsoft</b>	<b>30</b>
2.2.1. Introdução	30
2.2.2. Endereçamento no Processador 8086	30
2.2.3. Modelos Básicos de Armazenamento	31
2.2.4. Armazenamento do Código e dos Dados	32
2.2.5. Técnica de "Overlay"	35

2.2.6. Estruturas de Arquivos em Disco	36
<b>2.3. Esquema Geral de Funcionamento do Programa</b>	<b>40</b>
2.3.1. Introdução	40
2.3.2. Divisão do Programa em Blocos de Comandos	41
2.3.3. Implementação dos Blocos em Esquema de "Overlay"	46
2.3.4. Fluxo de Processamento	48
<b>2.4. Entrada das Características Nodais</b>	<b>52</b>
2.4.1. Sistema de Referência e Graus de Liberdade	52
2.4.2. Parâmetros a Serem Lidos	52
2.4.3. Número das Equações Para Cada Grau de Liberdade	53
2.4.4. Exemplo	54
<b>2.5. Montagem das Matrizes e Vetores dos Elementos</b>	<b>57</b>
2.5.1. Considerações Gerais	57
2.5.2. Matriz de Rigidez e Vetor de Cargas	57
2.5.3. Vetor de Contribuição na Matriz Global	58
2.5.4. Matriz Tensão por Deslocamento e Vetor de Tensões Prescritas	61
<b>2.6. Minimização da Semi-Banda da Matriz de Rigidez Global</b>	<b>63</b>
2.6.1. Introdução	63
2.6.2. Princípios Básicos - Matriz de Conexão	66
2.6.3. A Estratégia Cuthill-McKee	68
2.6.4. A Estratégia R. Collins	71
2.6.5. Comparação de Resultados	74
<b>2.7. Montagem do Sistema Global e Cálculo dos Resultados</b>	<b>80</b>
2.7.1. Introdução	80
2.7.2. Armazenamento do Sistema de Equações Globais	81
2.7.3. Montagem do Sistema de Equações Globais	84
2.7.4. Solução do Sistema de Equações Globais	87
2.7.5. Saída dos Deslocamentos Nodais	91
2.7.6. Tensões e ou Esforços nos Elementos	92
<b>2.8. Área de Armazenamento dos Dados</b>	<b>97</b>
2.8.1. Introdução	97
2.8.2. Definição dos "COMMON's" do Programa	98



2.8.3. Variáveis Locais	102
<b>2.9. Arquivos em Disco Rígido</b>	<b>104</b>
2.9.1. Introdução	104
2.9.2. Arquivos Temporários	104
2.9.3. Arquivos Acessados Pelos Pós-Processadores	110
2.9.4. Arquivos de Saída de Dados e Resultados	114
<b>2.10. Arquivo Para Entrada dos Dados</b>	<b>115</b>
2.10.1. Introdução	115
2.10.2. Linguagem Computacional Adotada	116
2.10.3. Filosofia Geral do Programa	117
2.10.4. Organização do Programa	118
2.10.5. Organização dos Menus	119
2.10.6. Apresentação dos Menus	120
2.10.7. Rotinas Para Entrada dos Dados	122
2.10.8. Gravação do Arquivo de Dados	127
<b>2.11. Capacidade, Desempenho e Conclusões</b>	<b>137</b>
2.11.1. Introdução	137
2.11.2. Capacidade de Análise	138
2.11.3. Acuidade Numérica	141
2.11.4. Tempo de Processamento	143
2.11.5. Conclusões Gerais	147
<b>2.12. Referências Bibliográficas</b>	<b>149</b>

## **CAPÍTULO 3 - Biblioteca de Elementos**

<b>3.1. Introdução</b>	<b>152</b>
<b>3.2. Conceitos Básicos Sobre o Método dos Elementos Finitos</b>	<b>155</b>
3.2.1. Energia Potencial Total	155
3.2.2. Princípio dos Deslocamentos Virtuais	155
3.2.3. Subdivisão do Domínio	156
3.2.4. Montagem das Equações de Equilíbrio	157
<b>3.3. Elemento Tipo 1 - Treliça</b>	<b>161</b>
3.3.1. Caracterização do Elemento	161
3.3.2. Coordenadas Locais e Globais	162

3.3.3. Matriz de Rigidez Global	163
3.3.4. Vetor de Cargas do Elemento	165
3.3.5. Matriz da Relação Tensão/Deslocamento	165
3.3.6. Vetor de Esforços ou Tensões Iniciais	166
3.3.7. Desempenho	167
<b>3.4. Elemento Tipo 2 - Barra</b>	<b>169</b>
3.4.1. Caracterização do Elemento	169
3.4.2. Coordenadas Locais e Globais	171
3.4.3. Matriz de Rigidez em Coordenadas Locais	173
3.4.4. Liberação de Deslocamentos e Rotações	175
3.4.5. Matriz de Rigidez em Coordenadas Globais	175
3.4.6. Vetor de Cargas do Elemento	176
3.4.7. Matriz da Relação Tensão/Deslocamento	177
3.4.8. Vetor de Esforços ou Tensões Iniciais	179
3.4.9. Constrangimento de Graus de Liberdade	180
3.4.10. Desempenho	181
3.4.11. Exemplos	182
<b>3.5. Elemento Tipo 3 - Contorno</b>	<b>187</b>
3.5.1. Caracterização do Elemento	187
3.5.2. Elemento Atuando Segundo Eixos Globais	188
3.5.3. Elemento Atuando Segundo Direção Qualquer	189
3.5.4. Desempenho	191
<b>3.6. Elemento Tipo 4 - Membrana</b>	<b>193</b>
3.6.1. Caracterização do Elemento	193
3.6.2. Coordenadas Locais e Globais	195
3.6.3. Funções de Interpolação	196
3.6.4. Matriz da Relação Deformações/Deslocamentos	198
3.6.5. Matriz de Constantes Elásticas	199
3.6.6. Matriz de Rigidez em Coordenadas Globais	200
3.6.7. Vetor de Cargas	201
3.6.8. Matriz da Relação Tensão/Deslocamento	206
3.6.9. Vetor de Tensões Iniciais	206
3.6.10. Modos Incompatíveis	207
3.6.11. Elementos de Forma Triangular	209
3.6.12. Desempenho	210
3.6.13. Exemplo	213

<b>3.7. Elemento Tipo 5 - Plano</b>	222
3.7.1. Caracterização do Elemento	222
3.7.2. Coordenadas Locais e Globais	224
3.7.3. Funções de Interpolação	224
3.7.4. Matriz da Relação Deformações/Deslocamentos	225
3.7.5. Matriz de Constantes Elásticas	226
3.7.6. Matriz de Rigidez em Coordenadas Globais	228
3.7.7. Vetor de Cargas	228
3.7.8. Matriz da Relação Tensão/Deslocamento	230
3.7.9. Vetor de Tensões Iniciais	231
3.7.10. Modos Incompatíveis	231
3.7.11. Elementos Triangulares	231
3.7.12. Desempenho	232
3.7.13. Exemplo	235
<b>3.8. Elemento Tipo 6 - Sólido</b>	240
3.8.1. Caracterização do Elemento	240
3.8.2. Funções de Interpolação	242
3.8.3. Matriz da Relação Deformações/Deslocamentos	244
3.8.4. Matriz de Constantes Elásticas	246
3.8.5. Matriz de Rigidez em Coordenadas Globais	247
3.8.6. Vetor de Cargas	248
3.8.7. Matriz da Relação Tensão/Deslocamento	252
3.8.8. Vetor de Tensões Iniciais	253
3.8.9. Modos Incompatíveis	254
3.8.10. Desempenho	255
3.8.11. Exemplo	257
<b>3.9. Elemento Tipo 7 - Placa/Casca</b>	263
3.9.1. Caracterização do Elemento	263
3.9.2. Considerações Gerais Sobre a Formulação do Elemento	265
3.9.3. Coordenadas Homogêneas Triangulares	266
3.9.4. Matriz de Constantes Elásticas	268
3.9.5. Elemento Triangular de Membrana	269
3.9.6. Elemento Triangular de Placa	272
3.9.7. Matrizes e Vetores em Coordenadas Globais	283
3.9.8. Desempenho	285

3.9.9. Exemplos	289
<b>3.10. Referências Bibliográficas</b>	<b>299</b>

## **CAPÍTULO 4 - Elemento Sapata Rígida**

<b>4.1. Introdução</b>	<b>301</b>
<b>4.2. Método dos Elementos de Contorno</b>	<b>302</b>
4.2.1. Teoria da Elasticidade - Equações Governantes	302
4.2.2. Soluções Fundamentais	304
4.2.3. Equações Integrais para Pontos Interiores e do Contorno	310
<b>4.3. Formulação do Elemento Sapata Rígida</b>	<b>320</b>
4.3.1. Introdução	320
4.3.2. Forma e Função Aproximadora	322
4.3.3. Parâmetros Internos em Função dos Valores Nodais	323
4.3.4. Discretização da Equação Integral	326
4.3.5. Cálculo da Rigidez das Sapatas	329
4.3.6. Translação da Matriz de Rigidez	332
4.3.7. Simetrização da Matriz de Rigidez	336
<b>4.4. Análise de Resultados</b>	<b>342</b>
4.4.1. Introdução	342
4.4.2. Integração Numérica e Tempo de Processamento	344
4.4.3. Sapatas Isoladas	347
4.4.4. Sapatas com Interação	355
4.4.5. Influência das Sapatas na Superestrutura	358
<b>4.5. Conclusões</b>	<b>383</b>
<b>4.6. Referências Bibliográficas</b>	<b>387</b>

# CAPÍTULO 1

## Introdução

### 1.1. OS COMPUTADORES E OS MÉTODOS NUMÉRICOS

O aparecimento dos computadores, no final da década de 40 e início da década de 50, provocou uma verdadeira revolução nos processos de análise de estruturas. Essa revolução deu-se basicamente em dois níveis : maior rapidez na execução de cálculos e aprimoramento dos modelos matemáticos utilizados para as análises.

Numa etapa inicial, os maiores benefícios advindos do seu aparecimento estiveram concentrados, a nível prático, numa agilização significativa dos procedimentos de cálculo. Assim sendo, num primeiro momento, os modelos matemáticos utilizados sofreram poucas alterações, tornando-se a computação uma ferramenta de otimização dos tempos gastos nessas análises. Por exemplo, o cálculo de um sistema de equações lineares de poucas incógnitas, que podia consumir horas de trabalho para sua solução, passou a ser feito em poucos minutos ou mesmo segundos.

Então, como consequência dessa agilidade nos cálculos, puderam ser desenvolvidos, ou simplesmente aproveitados, processos que até essa data tinham sido inviáveis, devido ao trabalho numérico envolvido. Dentro desses processos, destaca-se o Método dos Elementos Finitos.

Esse método de análise, cuja origem histórica normalmente aceita é o trabalho de um matemático [1.1], consiste em se dividir o domínio a ser considerado em um número finito de subdomínios (elementos), adotando-se para esses subdomínios funções aproximadoras. Isso faz com que o comportamento do domínio analisado possa ser descrito através de um sistema de equações lineares, que resolvido produz como resultado os parâmetros desejados, ponto por

ponto, elemento por elemento.

Obviamente, as análises de estruturas correntes requerem a solução de sistemas de equações lineares com centenas ou milhares de incógnitas, não sendo possível sua utilização sem o concurso de ferramentas computacionais. Entretanto, assim que essas ferramentas foram colocadas à disposição dos pesquisadores, os trabalhos fluíram em ritmo relativamente intenso, principalmente pela necessidade de um maior apuro no cálculo das estruturas de aviões [1.2], [1.3], [1.4], [1.5].

Outro marco importante no desenvolvimento dos métodos numéricos aplicados às estruturas foi o aparecimento de uma linguagem computacional de alto nível. Até 1957 as dificuldades de utilização dos computadores eram relativamente grandes, pois o pesquisador tinha a necessidade de programar o seu algoritmo em linguagem de máquina. Nesse ano surgiu o FORTRAN (FORMula TRANslation), linguagem computacional de alto nível desenvolvida para o IBM 704, que simplificou muito essa utilização. Com o desenvolvimento dessa linguagem, já no início da década de 60, qualquer pesquisador, utilizando qualquer computador, podia escrever seus programas em uma linguagem única e padronizada, o que propiciou um desenvolvimento rapidíssimo a essa área de pesquisa.

Nesse ponto, o advento dos computadores já não significava apenas uma redução do tempo necessário para o desenvolvimento de uma determinada análise, mas sim a possibilidade de se realizar cálculos utilizando-se modelos matemáticos mais complexos. Esse é sem dúvida o grande benefício das técnicas computacionais de análise estrutural: permitir que os resultados obtidos possam representar com muito maior acuidade o comportamento real das estruturas, desde que o modelo matemático corresponda ao real desempenho estrutural.

A partir dessas condições surgiram os grandes sistemas de análise de estruturas por computador. Inicialmente

pode-se citar o STRESS (STRuctural Engineering System Solver) desenvolvido pelo Massachussets Institute of Technology em 1963 e largamente utilizado por mais de uma década para análise estática de estruturas de barras em regime elástico. Depois o STRUDL, também desenvolvido pelo M.I.T., que surgiu em 1966, já como um sistema geral em elementos finitos. Em seguida, inúmeros programas foram sendo desenvolvidos, dentre os quais pode-se citar o ASKA, o STARDYNE, etc. Alguns desses sistemas gerais de elementos finitos são analisados com mais detalhes no próximo item.

Em fins da década de 70 e início de 80, um outro fator de enorme importância veio novamente produzir mudanças significativas nesse campo : o microcomputador. Até essa data o acesso aos computadores, e portanto aos benefícios descritos nos parágrafos anteriores, esteve limitado aos profissionais ligados aos institutos de ensino e pesquisa e às grandes empresas. Com os microcomputadores essa situação foi radicalmente alterada, tornando-se possível que qualquer pequena firma ou mesmo o engenheiro independente pudesse ter acesso a esses recursos computacionais, e portanto também às vantagens propiciadas por esses recursos.

Novamente ocorreu uma verdadeira revolução, já agora de forma muito mais disseminada, nos padrões habituais da análise estrutural. E novamente essa revolução apresentou dois estágios evolutivos : inicialmente apenas a otimização dos tempos necessários às análises e posteriormente a utilização de modelos matemáticos mais complexos e eficientes.

Assim, num primeiro momento os microcomputadores foram usados como ferramentas que agilizavam os procedimentos de cálculo realizados manualmente com calculadoras e tabelas. No caso específico da engenharia civil, o cálculo independente de lajes, vigas e pilares passou a ser realizado em tempo reduzido, mas continuavam sendo lajes, vigas e pilares independentes e os modelos matemáticos não

foram alterados.

Apenas num estágio posterior é que foram utilizados esses recursos na melhoria da qualidade dos procedimentos, e não apenas do tempo gasto com as análises. Assim é que os pavimentos de edifícios passaram a ser calculados como grelhas ou mesmo através de elementos finitos, integrando-se todas as peças e passando-se a obter resultados mais próximos da realidade. Além disso, também os modelos matemáticos de dimensionamento puderam ser mais sofisticados, reduzindo-se de maneira significativa o consumo de materiais.

No estágio atual de desenvolvimento dos microcomputadores, pode-se afirmar que praticamente qualquer análise estrutural pode ser neles realizada, desde que se tenha o programa adequado. Entretanto, é necessário reconhecer que o desenvolvimento das máquinas, "hardware", superou em muito o desenvolvimento dos programas, "software". Mesmo a nível internacional, não são comuns os bons programas de cálculo de estruturas em microcomputadores. Exatamente sobre essa lacuna é que se posiciona este trabalho, cujos objetivos estão explicitados em um dos itens subsequentes, logo após a breve revisão feita sobre alguns programas desenvolvidos para "mainframes" e microcomputadores.



## 1.2 SISTEMAS COMPUTACIONAIS PARA GRANDES COMPUTADORES

### 1.2.1. Considerações Gerais

O principal objetivo deste item é realizar uma breve revisão sobre os mais conhecidos sistemas de análise de estruturas reticuladas desenvolvidos para grandes computadores. Para não se estender em demasia esta revisão, deve-se limitá-la à análise estática de estruturas em regime elástico-linear, que é o objetivo do programa a ser elaborado nesta pesquisa.

É importante ressaltar que os programas desenvolvidos para grandes computadores podem apresentar estruturas de funcionamento bastante diversas das encontradas nos programas desenvolvidos para microcomputadores. Entretanto, não se deseja aqui estabelecer apenas uma comparação entre essas estruturas, mas sim realizar também uma análise dos recursos disponíveis em cada grupo. Desse modo, espera-se poder verificar eventuais diferenças qualitativas entre os programas desenvolvidos para esses dois tipos de equipamento.

### 1.2.2. O Programa ASKA

ASKA (Automatic System for Kinematic Analysis) é um programa baseado no Método dos Elementos Finitos e começou a ser desenvolvido pelo Prof. Argyris e seus colaboradores na Universidade de Stuttgart, Alemanha.

Importantes informações para um conhecimento mais aprofundado do ASKA I, que trata dos problemas estáticos em regime elástico linear, encontram-se nas referências [1.6], [1.7], [1.8], [1.9] e [1.10]. Essas informações estão

resumidas nos parágrafos seguintes.

O ASKA é composto de diversos segmentos que devem estar carregados na memória durante a execução. Esses segmentos são divididos em um módulo gerente e outros módulos primários e secundários que são carregados para a memória quando necessário. O bloco de dados é dividido em páginas de comprimento fixo. Quando necessário as páginas são carregadas na memória e a partir daí acessadas para a recuperação de informações.

O restante da área de trabalho é utilizada para alocação dinâmica de dados temporários e a gerência dessa área é realizada por subrotinas chamadas de "sistema de recuperação de dados". Para a solução de grandes estruturas, a matriz de rigidez do sistema é dividida em blocos, que são carregados na memória apenas no momento de serem trabalhados. Um recurso interessante é a possibilidade do usuário optar pela análise utilizando-se reais de simples ou dupla precisão. Assim, pode-se obter economia de memória e tempo de processamento, quando os reais de simples precisão puderem ser utilizados.

O programa ASKA coloca à disposição do usuário um total de 54 elementos diferentes. Elementos de treliça (2), elementos de membrana (8), elementos de placa (3), elementos de casca (9), elementos axissimétricos (9), elementos de barra (9) e elementos tridimensionais (14) completam esse total. Para todos esses elementos os resultados consistem nos esforços ou tensões nas extremidades do mesmo, não importando o número de pontos nodais internos que eventualmente ele possua.

Finalmente, é importante mencionar-se que o ASKA possui o recurso da sub-estruturação, ou seja, é possível analisar-se uma estrutura formada por módulos idênticos com grande economia de tempo de processamento. Nesse caso, um grande número de operações repetidas deixam de ser realizadas.

### 1.2.3. O Programa NASTRAN

NASTRAN (NAsa STRuctural ANalysis) é um programa baseado no Método dos Elementos Finitos e distribuído pela COSMIC (Universidade da Geórgia) mediante um pagamento simbólico. Como o próprio nome sugere, foi desenvolvido por diversos centros de pesquisa ligados à NASA. Esse desenvolvimento começou em 1964 e custou, em seus primeiros cinco anos, aproximadamente quatro milhões de dólares, produzindo um sistema computacional de mais de 150.000 comandos em FORTRAN.

Detalhes a respeito dos inúmeros recursos desse sistema podem ser encontrados nas referências [1.11], [1.12], [1.13], [1.14], [1,15] e [1.16]. Um pequeno resumo dessas informações é apresentado a seguir.

O sistema NASTRAN é composto por vários módulos funcionais independentes comandados por um sistema executivo. Sua capacidade de análise de grandes estruturas é muito boa, permitindo a solução de sistemas estruturais de mais de 100.000 graus de liberdade. As variáveis reais são de dupla precisão e a decomposição do sistema de equações é realizada por blocos com a semi-banda fixa.

A biblioteca de elementos possui os tipos básicos necessários a um programa geral de análise. Estão disponíveis elementos barra, treliça, estado plano de tensão (dois CST gerando um quadrilátero), placa (dois HCT gerando um quadrilátero), casca, casca cônica, elemento toroidal e tridimensional. Além desses tipos, que podem ser chamados de convencionais, estão disponíveis para uso dois elementos "painel", um deles trabalhando à força cortante e outro ao momento fletor, o elemento "escalar", para conectar pares de graus de liberdade, e um elemento "geral", que permite ao usuário montar uma matriz de rigidez qualquer e introduzi-la no sistema de equações da estrutura.

Também para este sistema é possível a definição de

super-elementos ou sub-estruturas, otimizando-se o cálculo de sistemas estruturais compostos por módulos repetidos.

#### 1.2.4. O Programa STARDYNE

O STARDYNE foi desenvolvido pela Mechanics Research Inc. (MRI) e seu uso disseminado pela Control Data Corporations - Cybernet System. Foi muito utilizado, inclusive no Brasil, exatamente por estar disponível em um grande "bureau" de processamento de dados, uma alternativa viável para pequenos usuários realizarem processamentos em elementos finitos.

As informações aqui colocadas a respeito do sistema baseiam-se, em quase sua totalidade, na referência [1.17]. Desse modo, essas informações não são tão completas como as obtidas para os outros programas citados, restando-se algumas lacunas sobre a sua organização geral e procedimento utilizado na solução do sistema de equações.

Um dos pontos mais elogiados do STARDYNE é sua entrada de dados, considerada de grande eficiência. Os dados relativos aos nós e elementos podem ser fornecidos em ordem aleatória através de títulos de identificação. Além disso, a consistência dos dados é excelente, evitando-se que o usuário possa perder tempo de processamento com um modelo matemático que não represente fielmente a estrutura que se deseja analisar.

Uma característica do programa que chama a atenção é sua relativamente pequena capacidade de análise. São permitidos apenas 999 nós e 9999 elementos, perfazendo um total de aproximadamente 6000 graus de liberdade. Sem dúvida é um limite pequeno, principalmente quando comparado a outros programas de análise estrutural para grandes computadores.

O STARDYNE coloca à disposição do usuário elemento de

barra, elemento triangular de placa, elemento retangular de placa, elemento sólido cúbico e elemento sólido tetraédrico. Também possibilita a definição de até 100 sistemas rígidos, com no máximo 18 nós cada um, para simulação de diafragmas.

Não é mencionado o recurso de sub-estruturação, mas o sistema provê um renumerador de nós para a diminuição da semi-banda da matriz de rigidez. Isso possibilita uma modelagem mais eficiente, pois o usuário não precisa se preocupar com as conexões resultantes.

#### 1.2.5. O Programa STRUDL

O STRUDL (STRUCTURAL Design Language), também conhecido como ICES-STRUDL, é um subsistema do "Integrated Civil Engineering System" (ICES) e foi desenvolvido pelo M.I.T. Civil Engineering System Laboratory.

No presente trabalho estaremos examinando rapidamente algumas características do ICES-STRUDL II, que pode ser conhecido com maiores detalhes através das referências [1.18], [1.19], [1.20], [1.21], [1.22].

Este programa tem como característica positiva sua entrada de dados, considerada muito eficiente. Trata-se da POL (Problem Oriented Language) que coloca à disposição do usuário uma série de recursos que facilitam bastante a comunicação com a máquina. De maneira semelhante ao programa do item anterior, aqui também pode-se definir os dados da geometria e carregamento do problema a ser analisado sem a necessidade de seguir-se uma ordem pré-estabelecida. Além disso, o programa possui valores padrão para os diversos parâmetros necessários a uma determinada análise, diminuindo significativamente a quantidade de dados a serem fornecidos.

Vários elementos estão disponíveis para os usuários do

programa. Podem ser destacados : elemento treliça plana e tridimensional, elemento de pórtico plano e tridimensional, elemento de grelha, elementos para estados planos de tensão e deformação (12), elementos de placa (4), elementos tridimensionais (3) e casca (3). Portanto, tem-se um total de 27 tipos, com destaque para os elementos obtidos através de simplificações do elemento barra, que no STRUDL II versão 2 podem ser usados simultaneamente com os outros elementos do sistema.

Por fim, menciona-se que uma característica interessante do programa aqui analisado é a possibilidade de fazer-se o dimensionamento de peças de aço e concreto armado com a utilização de rotinas embutidas no próprio sistema.

#### 1.2.6. O Programa ADINA

O ADINA é um programa de propriedade da ADINA Engineering AB, Suécia. Foi desenvolvido em FORTRAN, com base no método dos elementos finitos e a participação destacada do Prof. Klaus-Jurgen Bathe, do MIT.

Maiores informações sobre o sistema aqui analisado podem ser obtidas nas referências [1.23], [1.24], [1.25] e [1.26], sendo um breve resumo delas apresentado nos parágrafos seguintes.

Um grande destaque deve ser dado na solução do sistema de equações geral da estrutura a ser analisada. O ADINA monta e resolve esse sistema num esquema de armazenamento, conhecido como "Skyline", que consiste em diversos blocos compostos apenas pelos elementos não nulos ou que poderão tornar-se não nulos durante a solução [1.27]. Esse procedimento, também conhecido como "altura efetiva das colunas", costuma ser mais eficiente que aqueles que consideram a banda constante, principalmente quando não se

utiliza um renumerador de nós da estrutura antes da montagem da matriz.

Outro ponto interessante do ADINA é sua biblioteca de elementos, muito completa. Estão disponíveis elementos de treliça, cabo, barra com dois nós e isoparamétrico, elementos bidimensionais para estados planos de tensão e deformação, elementos tridimensionais, placa, casca e ainda isoparamétrico para cascas, permitindo que o usuário possa analisar de forma eficiente um variado conjunto de estruturas.

O programa permite ainda, e também com grande eficiência, o uso do recurso da sub-estruturação. Pode-se definir um conjunto, repetí-lo várias vezes e compô-lo com outras subestruturas definidas, criando a estrutura total a ser analisada.

#### **1.2.7. O Programa LEBRE**

O programa LEBRE (Linguagem Educacional BRasileira para Engenharia), iniciou-se a partir de uma idéia muito interessante : desenvolver um sistema para análise de estruturas com ênfase em aspectos educacionais. Da primeira fase do projeto, que iniciou-se em 1978, participaram pesquisadores da UFPR, da COPPE-UFRJ e do CPGEC-UFRGS, coordenados pelo Prof. Agustin J. Ferrante.

As informações aqui resumidas encontram-se, com maiores detalhes, apresentadas nas referências [1.28] e [1.29].

O LEBRE tem uma proposta de certo modo diversa das verificadas para a maioria dos programas aqui analisados. Trata-se de um sistema que tem como característica assumida o objetivo de ser uma estrutura básica para desenvolvimentos posteriores, a serem implementados por diversos pesquisadores, seguindo apenas uma padronização

geral.

Percebe-se que uma grande importância foi dada às rotinas de entrada de dados, de modo a torná-las acessíveis aos usuários não especialistas. Todos os comandos têm grande flexibilidade de montagem e procuram facilitar ao máximo a tarefa de definição da geometria e do carregamento, dispondo-se inclusive de recursos de geração dos dados.

O LEBRE, na sua versão I-A, apresenta um total de 16 tipos de elementos. São cinco elementos para estruturas de barras, treliça plana e tridimensional, grelha e pórtico plano e tridimensional; quatro elementos para estados planos de tensão e deformação, triângulo e quadrilátero de primeira e segunda ordem; dois elementos para sólidos axissimétricos, quadriláteros de primeira e segunda ordem; um elemento tridimensional, isoparamétrico de primeira ordem; dois elementos para análise de placas, retangular não-conforme e triangular híbrido; e dois elementos para análise de cascas, retangular não-conforme e triangular híbrido.

A montagem e a solução do sistema de equações geral da estrutura são feitas considerando-se o armazenamento da matriz em semi-banda. Não se obtém nas referências consultadas para a elaboração deste item informações sobre a capacidade do programa e detalhes a respeito da matriz de rigidez global. Entretanto pode-se supor que tanto a sua montagem, bem como a sua solução, sejam realizadas por blocos, sem o que a capacidade do programa seria muito reduzida.

Por fim, deve-se mencionar que como resultados do processamento pode-se obter deslocamentos e reações nodais e esforços e ou tensões nos elementos.



## 1.3. SISTEMAS COMPUTACIONAIS PARA MICROCOMPUTADORES

### 1.3.1. Considerações Gerais

Neste item apresentar-se-á um resumo de dois programas desenvolvidos para a utilização em microcomputadores de 16 bits. Essa apresentação, como já foi mencionado, tem como objetivo estabelecer algumas comparações com os programas desenvolvidos para grandes computadores, que foram discutidos no item anterior.

Para atingir-se o objetivo proposto, serão apresentadas informações tão completas quanto for possível sobre a estrutura de funcionamento, capacidade de análise, tipos de elementos disponíveis e recursos extras de otimização, além de comentários a respeito de entrada dos dados e saída dos resultados. Resta ressaltar que, a exemplo do item anterior, essas informações devem limitar-se à análise estática de estruturas em regime elástico-linear.

### 1.3.2. O Programa SAP90

O SAP90 é o sucessor de um dos programas de análise estrutural para microcomputadores mais conhecidos em todo o mundo : o SAP80. Esses programas são de propriedade da Computers & Structures Inc. e foram desenvolvidos por Edward L. Winson e Ashraf Habibullah.

As principais diferenças entre o SAP80 e o SAP90 dizem respeito à inclusão de um elemento tridimensional e desenvolvimento de rotinas de pós-processamento que permitem ao usuário o desenho de esforços e tensões da estrutura analisada. As demais características foram

mantidas, inclusive a entrada dos dados, como pode-se notar pelo exame das referências [1.30] e [1.31].

O SAP90 é formado de diversos módulos independentes que são processados em sequência e se comunicam através de arquivos de dados temporários montados no disco rígido. Inicialmente atua um módulo que interpreta o arquivo formatado que contém os dados fornecidos pelo usuário e monta outros arquivos não formatados que serão lidos pelos demais módulos do sistema. Logo após são carregados na memória os módulos que executam a montagem das matrizes de rigidez dos elementos. Em sequência, entra o módulo de montagem e solução do sistema de equações global da estrutura, inclusive com a saída dos deslocamentos nodais. Por fim atuam os módulos que calculam esforços e tensões nos elementos. Esse sistema de funcionamento utilizado tem como desvantagem um maior acesso ao disco rígido, pois todas as informações são passadas através de arquivos. Entretanto, permite que a capacidade do programa possa ser relativamente grande, sem a necessidade de se trabalhar com estruturas de dados mais complexas.

Em relação à capacidade de análise, as informações disponíveis não são claras, pois o manual do usuário não menciona esses limites. Entretanto, em folhetos de publicidade lançados no Brasil, é mencionado para o SAP80 o limite máximo de 10.000 graus de liberdade, sem qualquer explicação adicional sobre o número de elementos ou pontos nodais permitidos.

Quanto aos tipos de elementos disponíveis ao usuário, o programa define apenas três : um elemento de barra, um elemento de casca e um elemento sólido. A barra permite a modelagem de treliças e pórticos, planos ou espaciais, além de grelhas. O elemento de casca possibilita a análise de estados planos de tensão, placas, além das cascas propriamente ditas. Finalmente, o elemento sólido hexaédrico é utilizado preferencialmente para modelagem de estruturas tridimensionais.

A montagem do sistema de equações global é feita por blocos e por altura efetiva de colunas. Esse procedimento produz um bom resultado em termos de tempo de processamento, principalmente porque o primeiro módulo executado realiza uma renumeração dos nós da estrutura de modo a produzir uma otimização desse perfil. Quanto à solução do sistema propriamente dita, o processo utilizado é o da eliminação de Gauss.

A entrada dos dados do SAP90, que é idêntica à do SAP80, permite uma certa flexibilidade ao usuário pois são usados formatos livres com títulos identificadores. Alguns recursos de geração também estão disponíveis, se bem que de eficiência relativa quando se trata de elementos de casca ou sólidos, faltando ao programa alguns módulos pré-processadores que potencializem ainda mais as gerações dessas redes.

Finalmente, quanto à saída dos resultados, além dos já mencionados pós-processadores que permitem o desenho dos esforços e tensões nos elementos, o programa oferece também a criação de arquivos formatados com esses resultados. Nesses arquivos podem ser encontrados os deslocamentos nodais, esforços ou tensões nos elementos e ainda reações para todos os deslocamentos restritos na estrutura.

### **1.3.3. O Programa SUPERSAP**

O SUPERSAP é um sistema de propriedade da Algor Interactive Systems Inc..É composto por um grande número de módulos representados por pós e pré-processadores, além dos módulos de análise propriamente ditos. As informações constantes deste item foram obtidas mediante a utilização do sistema e da referência [1.32].

O módulo de análise para estruturas estáticas é montado numa filosofia bastante diversa do SAP90,

constituindo-se num único programa que ao ser carregado para a memória traz uma parte básica de gerenciamento, que administra a chamada das demais partes num esquema de "overlay". Desse modo, na versão 7.52 do mencionado programa, o módulo de análise ocupa aproximadamente 650 Kbytes para seu armazenamento.

Quanto à capacidade de análise, novamente as informações não são fornecidas no manual do usuário. Apenas pela utilização do programa, que apresenta algumas informações a respeito da percentagem de memória utilizada em cada passo do processamento, pode-se perceber que o número máximo de nós deve situar-se em torno de 4400. Outra conclusão que pode ser estabelecida, agora pela formatação dos resultados apresentados, é que podem ser usados no máximo 9999 elementos.

O programa coloca à disposição do usuário um total de 10 tipos de elementos. Perfazem esse total : um elemento de treliça tridimensional; um elemento de pórtico tridimensional; um elemento de membrana; um elemento para análise de estados planos de tensão, deformação e axissimétricos, um elemento sólido hexaédrico; um elemento de placa e casca; um elemento para imposição de condições especiais de contorno; um elemento de casca espessa de 8 a 21 nós; um elemento geral, para carregamento de uma matriz de rigidez qualquer diretamente na matriz de rigidez global da estrutura; e finalmente um elemento tubular tridimensional.

O SUPERSAP provê um renumerador de pontos nodais da estrutura definida, que pode ou não ser acionado, de acordo com a opção do usuário. Esse renumerador atua internamente, sendo que todos os dados e as saídas de resultados são em relação à numeração original.

A montagem do sistema de equações globais da estrutura é feita por blocos de semi-banda fixa. O número de equações em cada bloco é calculado para que sejam acomodados na memória central do computador dois blocos completos de cada

vez. A solução desse sistema é realizada pelo processo de Gauss, com todos os casos de carregamento considerados simultaneamente.

Quanto à entrada dos dados, o programa oferece vários pré-processadores que permitem a geração automática de redes dos diversos elementos disponíveis. Esses pré-processadores baseiam-se no conceito de pontos chave ou "key nodes". Através desses pontos toda a rede é gerada de modo a simplificar enormemente o trabalho de definição de geometria que teria que ser feito pelo usuário. Assim, a entrada dos dados torna-se muito mais simples que a tradicional, onde são fornecidos separadamente nós e elementos da estrutura, mesmo que através de recursos de geração.

Finalmente, para a saída dos resultados, é possível a utilização de pós-processadores que desenham as tensões e esforços nos elementos e os deslocamentos dos pontos nodais. Entretanto, além desse recurso, o programa monta arquivos que contém esses valores obtidos.

#### **1.3.4. Comparação Com Sistemas Para Grandes Computadores**

Talvez a mais evidente diferença de recursos entre esses dois grupos de programas seja a quantidade de tipos de elementos disponíveis para os usuários. Os programas desenvolvidos para microcomputadores têm como característica poucos tipos de elementos, sendo no máximo um para cada modelagem específica. Essa situação contrasta vivamente com os programas elaborados para os grandes computadores, que oferecem diversos tipos para uma mesma aplicação. Por exemplo, pode-se citar o programa ASKA, que possui um total de 54 tipos de elementos e apresenta 14 diferentes tipos de elementos tridimensionais. Além da quantidade, os elementos presentes nos sistemas

elaborados para microcomputadores são, em geral, mais pobres em termos de função aproximadora.

Quanto aos motivos que levam a essas diferenças, eles não são tão evidentes quanto pode-se imaginar à primeira vista. As diferenças de "hardware" não servem como justificativa, pois os grandes computadores da década de 70 não tinham recursos muito maiores que os microcomputadores de hoje. Parece que na verdade uma menor quantidade de tipos de elementos seja decorrente de uma necessária simplificação no uso dos programas. É muito difícil que o usuário possa ser beneficiado com um leque tão amplo de possibilidades de utilização, exceto se for um verdadeiro especialista em elementos finitos. Como já se mencionou, a grande vantagem dos microcomputadores foi exatamente a de permitir que uma quantidade muito maior de pessoas tivesse acesso a bons programas de análise de estruturas. Entretanto, é necessário que os programas não sejam de uso tão complexo.

Quanto ao fato das funções aproximadoras serem em geral mais pobres, trata-se de uma adaptação do processo de análise. Desde que se tenha uma boa geração para os dados da geometria, é preferível a utilização de redes mais refinadas e elementos mais simples. Tanto para a própria modelagem das estruturas quanto para a utilização dos resultados obtidos, as redes mais refinadas apresentam vantagens significativas. Caso clássico dessas vantagens é a correta colocação de cargas não uniformes atuantes sobre as estruturas. Além disso, a obtenção de resultados em grande número de pontos facilita o trabalho dos pós-processadores, que no caso disso não ocorrer necessitariam de interpolações complexas para bem executar seus objetivos.

#### 1.4. OBJETIVOS GERAIS DO TRABALHO

O principal objetivo deste trabalho é montar um sistema computacional, em elementos finitos, para realizar análises de estruturas em regime elástico-linear, com a utilização de microcomputadores de 16 bits. Esse sistema deve permitir inclusive a consideração da interação com a fundação, através de um elemento desenvolvido com base no Método dos Elementos de Contorno. A seguir são descritos os principais itens que compõem as especificações do programa.

a) O programa deve possibilitar a análise de grandes estruturas, no mínimo 30.000 graus de liberdade, mesmo para microcomputadores com apenas 640 Kbytes de memória central, sem qualquer tipo de expansão. Entretanto, é importante que seu esquema de funcionamento permita uma adaptação extremamente simples a maiores memórias. Desse modo, poder-se-á utilizar com facilidade um recurso que deve estar em breve disponível. Trata-se, para os computadores montados com base nos processadores 80286 e 80386, das memórias centrais de até 16 Mbytes.

b) A criação de arquivos temporários e os algoritmos utilizados nas análises devem ser otimizados, de modo a viabilizar análises de grandes estruturas mesmo com a utilização de discos rígidos de apenas 30 Mbytes e com tempos de processamento relativamente pequenos. Assim sendo é prioritária a utilização de arquivos sequenciais, que ocupam menor espaço no disco rígido e têm tempo de acesso reduzido em comparação com os randômicos. Também no sentido de otimizar os tempos de processamento, pode-se prever trechos em linguagem de máquina nos procedimentos cuja utilização repetida assim o recomende.

c) Para a discretização da superestrutura, o sistema computacional deve prover elementos que efetivamente

permitam a simulação adequada das mais diversas peças estruturais. Assim, os seguintes elementos devem estar disponíveis : treliça, barra, membrana (estado plano no espaço), sólido, placa e casca. Todos esses elementos devem ser compatíveis entre si, podendo portanto serem utilizados simultaneamente e constituindo um sistema computacional único.

d) Numa primeira etapa, a interação com a fundação deve ser realizada através de um elemento sapata, pois as fundações rasas normalmente apresentam uma menor rigidez e nesse caso a consideração da interação torna-se mais importante. É interessante ressaltar que a utilização do elemento deve ser muito simples, de modo que o usuário possa realizá-la de forma cômoda e eficaz.

e) De modo a facilitar a entrada dos dados das estruturas considerando a junção de subestruturas montadas de forma independente, é importante que o programa possua, como recurso extra e opcional, uma renumeração dos nós fornecidos. Essa renumeração deve estar ativa apenas nas etapas onde isso signifique redução da semi-banda da matriz de rigidez do sistema estrutural, sem que o usuário necessite conhecê-la.

f) Toda a estrutura do programa deve ser modulada e geral, prevendo-se as futuras modificações que o sistema deve sofrer para um constante aprimoramento ao longo do tempo. Assim sendo, entende-se esta etapa como uma primeira fase de um processo de aperfeiçoamentos sucessivos que devem ser implementados por diversos pesquisadores, que não precisam conhecer em detalhes toda a estrutura do programa.

Segundo esses parâmetros é que se deu o desenvolvimento da presente pesquisa, cuja organização geral apresenta-se no próximo item.



## 1.5. ORGANIZAÇÃO DO TRABALHO

O presente trabalho encontra-se organizado em 4 capítulos. Em cada capítulo, encarado como uma unidade relativamente autônoma em relação aos demais, encerra-se um assunto bem definido e de certo modo estanque. Assim, espera-se que as informações possam ser apresentadas de forma simples e organizada.

Todas as equações que aparecem no trabalho são numeradas através do símbolo "... (n.m) ". Por sua vez as referências bibliográficas aparecem como " [n.m] ". Finalmente, as figuras são numeradas por " n.m ". Em todos esses casos o número "n" é o número do capítulo considerado, enquanto "m" indica a ordem de aparecimento da equação, referência bibliográfica ou figura dentro desse capítulo.

O primeiro capítulo, que contém este item de organização do trabalho, apresenta uma introdução geral a respeito do tema em estudo. Inicialmente, traça-se um paralelo entre o desenvolvimento dos computadores e dos métodos numéricos. Em seguida, são apresentados alguns sistemas computacionais para análise de estruturas utilizando grandes e microcomputadores. Então, são apresentados os principais objetivos da pesquisa e a seguir esta organização do trabalho. Por fim, a bibliografia referenciada neste capítulo.

Já o capítulo número dois contém todo o corpo básico do programa. Após uma introdução onde situa-se o módulo de análise dentro da filosofia geral do sistema, é apresentado um item com as principais características da linguagem escolhida para o desenvolvimento do "software". Em seguida vem um esquema geral de funcionamento, com todas as rotinas do sistema, organização dos blocos de rotinas e etapas de funcionamento. Logo após, são apresentados os itens que detalham essas etapas de funcionamento : entrada dos nós, informações gerais sobre a montagem de matrizes e vetores

dos elementos, minimização da banda da matriz de rigidez global e ainda a obtenção dos resultados da análise. Depois, são apresentados detalhes que dizem respeito ao armazenamento dos dados e estrutura de arquivos em disco rígido, inclusive o arquivo de dados. Por fim, é feita uma comparação entre o programa desenvolvido e outros já existentes para microcomputadores, sendo que esses resultados são utilizados para conclusões gerais a respeito do desenvolvimento realizado.

O capítulo três serve para o detalhamento da chamada biblioteca de elementos. Começa com um breve resumo sobre a análise matricial de estruturas, em especial voltada para o Método dos Elementos Finitos. Essa técnica é que será usada para o desenvolvimento da maioria dos elementos presentes nesse capítulo. Logo após, são apresentados detalhes a respeito da montagem de matrizes de rigidez, vetor de cargas, matriz de relação tensões e esforços por deslocamentos e matriz de tensões iniciais para cada um dos sete tipos básicos de elementos disponíveis. Por fim, são apresentadas as referências bibliográficas dos assuntos estudados.

No quarto capítulo é apresentado o elemento sapata rígida, que possibilita a consideração da interação entre a superestrutura e o solo. Após uma breve introdução, coloca-se um resumo do Método dos Elementos de Contorno, que serve de base ao desenvolvimento do elemento. Logo após, é apresentada a formulação propriamente dita, com detalhes sobre o desenvolvimento da matriz de rigidez. Depois, são analisados resultados obtidos com a utilização do elemento, desenvolvendo-se considerações sobre suas características mais marcantes. Por fim, em seguida às conclusões gerais sobre o elemento, são apresentadas as referências bibliográficas mencionadas.

## 1.6 REFERÊNCIAS BIBLIOGRÁFICAS

- [1.1] Courant, R.; "Variational Methods for the Solutions of Problems of Equilibrium and Vibrations"; Bull. Am. Math. Soc.; vol 49; 1943.
- [1.2] Langefors, B.; "Analysis of Elastic Structures by Matrix Transformation with Special Regard to Semimonocoque Structures"; J. Aeron. Sci.; vol 19; 1952.
- [1.3] Levy, S.; "Structural Analysis and Influence Coefficients for Delta Wings"; J. Aeron. Sci.; vol 20; 1953.
- [1.4] Argyris, J.H.; Kelsey, S.; "Energy Theorems and Structural Analysis"; Aircraft Engineering; Oct-Nov 1954, Feb-May 1955.
- [1.5] Turner, M.J.; Clough, R.W.; Martin, H.C.; Topp, L.J.; "Stiffness and Deflections Analysis of Complex Structures"; J. Aeron. Sci.; vol 23; 1956.
- [1.6] Schrem, E.; Roy, J.R.; "An Automatic System For Kinematic Analysis ASKA Part I"; Proc. IUTAM Colloq. High Speed Comput. of Elastic Structures; Universidade de Liège; Belgica; 1970.
- [1.7] Argyris, J.H.; Bronlund, O.E.; "Computer Aided Structural Analysis The Machine-independent System ASKA"; Nord. Data 70 Conf.; Copenhagen; 1970.
- [1.8] Argyris, J.H.; Grieger, I.; Schrem, E.; "Structural Analysis by Problem Oriented Languages"; 23rd Ann. General Meeting of the Aeronaut. Soc. of India, Indian Inst. of Technol.; Kanpur, Índia; 1971.

- [1.9] Schrem, E.; "ASKA User's Reference Manual"; ISD-Rep no. 73; Stuttgart; 1971.
- [1.10] Bernhardt, K; Streiner, P.; "Glo. A Set of ASKA Processors Providing Direct Access to Internal Data"; Institut für Statik und Dynamik der Technische Hochschule Stuttgart; Stuttgart; 1971.
- [1.11] "The NASTRAN Theoretical Manual"; NASA SP-221; 1970.
- [1.12] "The NASTRAN User's Manual"; NASA SP-222; 1970.
- [1.13] "The NASTRAN Programmers Manual"; NASA SP-223; 1970.
- [1.14] "NASTRAN Demonstration Problem Manual"; NASA SP-224, 1970.
- [1.15] "A Technical Evaluation of the NASTRAN Computer Program "; The Boeing Co.; Seattle; 1971.
- [1.16] Beste, D.H.; Herness, E.D.; Ice, M.W.; "A Capabilities Guide to the NASTRAN Computer Code"; Scientific Systems Rep. Boeing Comput. Services; Seattle; 1971.
- [1.17] "Numerical and Computer Methods in Structural Mechanics"; Ed. S. Fenves, N. Perrone, A. Robinson e W. Schnobrich; Academic Press; Londres; 1973.
- [1.18] Logcher, R.D.; Sturman, G.M; "STRUDL - A Computer System for Structural Design", J. Structural Div. Proc. ASCE; vol 92; 1970.
- [1.19] "ICES System : General Description"; Report R67-94; Massachusetts Inst. of Technology; 1967.

- [1.20] "ICES-STRUDL II, Engineering User's Manual, Vol 1, Frame Analysis"; Report R68-91; Massachusetts Inst. of Technology; 1968.
- [1.21] "ICES-STRUDL II, Engineering User's Manual, Vol 2, Additional Design and Analysis Facilities"; Report R70-71; Massachusetts Inst. of Technology; 1970.
- [1.22] "ICES-STRUDL II, Engineering User's Manual, Vol 3, Reinforced Concrete Structure"; Report R70-35; Massachusetts Inst. of Technology; 1970.
- [1.23] "ADINA, ADINAT e ADINA-PLOT : User's Manual"; ADINA Engineering AB; Vasteras; Suécia.
- [1.24] "Proceedings of the ADINA Conference"; Report AVL 82448-6; Massachusetts Inst. of Technology; 1977.
- [1.25] "Proceedings of the ADINA Conference"; Report AVL 82448-9; Massachusetts Inst. of Technology; 1977.
- [1.26] "Proceedings of the ADINA Conference"; J. Computers and Structures; Vol 13; 1981.
- [1.27] Bathe, K.J.; "Finite Element Procedures in Engineering Analysis"; Prentice-Hall, Inc.; 1981.
- [1.28] Ferrante, A.J.; "Linguagem LEBRE I-A para Análise de Estruturas - Manual do Usuário"; Publ. COPPE/UFRJ ; Rio de Janeiro; 1981.
- [1.29] Ferrante, A.J.; de Carvalho, J.A.P. "Linguagem LEBRE I-A para Análise de Estruturas - Manual do Sistema"; Publ. COPPE/UFRJ ; Rio de Janeiro; 1981.

[1.30] "SAP80 : User's Manual"; Computers & Structures Inc.; Berkeley; 1984.

[1.31] "SAP90 : User's Manual"; Computers & Structures Inc.; Berkeley; 1988.

[1.32] "SUPERSAP : User's Manual"; Algor Interactives Systems, Inc.; 1987.

## CAPÍTULO 2

### Organização Geral do Programa

#### 2.1. INTRODUÇÃO

O sistema LASER (Linguagem para Análise de Sistemas Estruturais Reticulados) é um conjunto de programas voltados para as diversas etapas envolvidas em uma análise estrutural de grande porte. Compreende, além do programa de análise propriamente dito, módulos que podem ser classificados em dois grandes conjuntos : pré-processadores e pós-processadores. Compõem os pré-processadores programas de entrada e conferência de dados, onde são oferecidos grandes recursos de geração automática e permitido o uso de tela ou "plotter" para apresentação de desenhos. Além disso, na categoria de pós-processadores, existem módulos para apresentação de resultados, que fazem desenhos da estrutura deformada e curvas de tensões e ou esforços, também com a utilização de tela ou "plotter".

O objetivo deste capítulo é fornecer um panorama geral sobre a montagem do já referido módulo de análise. Dentro desse panorama pretende-se mostrar como estão organizadas as diversas etapas do processamento, seus interrelacionamentos, filosofia de armazenamento do código, dos dados e dos arquivos em disco rígido, além de outros detalhes de significância para o perfeito entendimento do programa.

Portanto, é um capítulo muito importante pois define toda a filosofia básica do sistema computacional. Muito mais que qualquer parte específica, essa filosofia geral é que vai ser responsável pela capacidade que o programa apresentará para futuros desenvolvimentos. Qualquer erro nessas definições iniciais comprometerá totalmente o desempenho do "software", mesmo que sejam a ele anexadas as

melhores rotinas.

Um dos pontos mais importantes a se analisar para a elaboração de um bom programa computacional é a linguagem a ser utilizada. Essa linguagem precisa ser escolhida através de uma análise consciente, que pese com cuidado as principais características necessárias ao desenvolvimento do programa. Neste caso, após considerações de diversas ordens, optou-se pela utilização do FORTRAN 4.01 da Microsoft Corporation

O principal motivo que levou a essa escolha é o fato do FORTRAN ser uma linguagem quase universal para a engenharia de estruturas. Esse fato é muito importante pois não se pretende com este trabalho gastar tempo e esforço de forma desnecessária, remontando-se algoritmos já desenvolvidos. Para tanto, tratou-se de escolher uma linguagem computacional de modo a ser possível um bom intercâmbio de informações, facilitando-se o aproveitamento de algoritmos eficientes já desenvolvidos. Além disso, como pretende-se que este programa sirva de esqueleto básico para numerosos aperfeiçoamentos futuros, é de interesse adotar-se uma linguagem conhecida por quase todos os pesquisadores da área. Dentro dessas considerações colocadas, um bom e completo compilador FORTRAN foi realmente a solução ideal.

Este capítulo está organizado de modo que, logo após esta introdução, vem uma compilação dos principais recursos do FORTRAN 4.01 da Microsoft. A seguir é apresentado o esquema geral de funcionamento do sistema computacional que compõe o módulo de análise, com apresentação das rotinas componentes, fluxo de processamento, etc. Depois são mostrados detalhes a respeito das diversas fases existentes, como entrada das características nodais, montagem dos elementos e do sistema global, minimização da semi-banda da matriz, solução do sistema e cálculo de esforços e tensões. Em sequência são discutidos o tratamento dado à área de dados e aos arquivos em disco



utilizados no processamento. Por fim, são apresentados detalhes a respeito da capacidade e eficiência de utilização do programa montado, com comparações em relação a outros programas similares existentes. Essas informações possibilitarão determinadas conclusões gerais sobre os tópicos discutidos.

## 2.2 CONCEITOS BÁSICOS DO FORTRAN 4.01 DA MICROSOFT

### 2.2.1. Introdução

Dentro deste item estarão colocadas algumas das principais características da linguagem FORTRAN adotada para uso no programa em questão. Estas características, encontradas com maiores detalhes nas referências [2.1] e [2.2], são importantes para que se possa compreender as soluções adotadas para determinados pontos específicos do algoritmo a ser apresentado.

### 2.2.2. Endereçamento no Processador 8086

Inicialmente deve-se fazer uma pequena explanação sobre a maneira como o processador 8086, e toda a sua família, trata o endereçamento da memória. Esse entendimento é fundamental para uma perfeita compreensão dos recursos do FORTRAN em questão, bem como das demais linguagens a serem utilizadas nos computadores de 16 bits.

Normalmente, o computador equipado com qualquer processador da família do 8086 pode endereçar apenas 64 Kbytes de memória. Como esse limite é extremamente restritivo, não permitindo o desenvolvimento de programas mais elaborados, a maneira encontrada para a superação dessa barreira foi a divisão da memória total em blocos de 64 Kbytes.

Esses blocos são chamados de "segmentos" e o ponto de partida de cada segmento na memória é um endereço de 16 bits. O processador 8086 reserva quatro registros para endereços de segmentos básicos : CS ("code segment"), DS ("data segment"), SS ("stack segment") e ES ("extra

segment").

Entretanto, o endereço de um segmento aponta apenas para a base do mesmo. Para que o processador refira-se a um item particular dentro de um determinado segmento é necessário mais um endereço de 16 bits, chamado endereço do "offset", o qual dá a posição do item em relação à base mencionada.

Portanto, para um endereço completo no 8086 é necessário um total de 32 bits, sendo 16 bits para o endereço do segmento e outros 16 bits para o endereçamento do chamado "offset".

### **2.2.3. Modelos Básicos de Armazenamento**

O FORTRAN 4.01 da Microsoft possui 3 modelos básicos de armazenamento: "Medium Model", "Large Model" e "Huge Model". O modelo "default", utilizado sempre que uma declaração explícita não estiver presente, é definido durante a instalação do compilador e pode ser "Medium" ou "Large". Entretanto, esse "default" pode ser alterado no instante da compilação através de uma diretiva geral ou através de comandos para variáveis em particular, valendo nesse caso qualquer um dos três modos anteriormente mencionados.

Quando um determinado programa possui apenas um segmento para o código e um segmento para os dados, o endereçamento da memória pode tornar-se mais simples e eficiente. Isso se dá pois o processador já possui o endereço básico para o segmento que contém os códigos (CS) e os dados (DS). Portanto qualquer item a ser referenciado necessita apenas de um endereço de 16 bits, o que significa uma vantagem em relação ao endereço completo de 32 bits, pois demanda menos espaço de memória e tempo de processamento. Nesse caso, o modelo de armazenamento mais

adequado é o "Medium Model". Com ele, o compilador vai gerar automaticamente endereços de apenas 16 bits, incorporando ao programa as vantagens anteriormente mencionadas.

Entretanto, a maioria dos programas necessita de mais de um segmento tanto para o código como para os dados. Isso faz com que o compilador tenha que gerar pelo menos alguns endereços completos de 32 bits. Então, o modelo adequado, apesar de tornar o programa menos eficiente, é o "Large Model". Apenas a destacar que, para este modelo, apesar da possibilidade do total do código e dos dados ultrapassarem o limite de 64 Kbytes, cada módulo do programa e cada variável individual devem estar dentro desse limite (entende-se por módulo de programa um arquivo fonte compilado separadamente). Assim sendo, o programa total deve ser particionado em módulos que ocupem no máximo 64 Kbytes e cada variável não poderá ultrapassar em nenhuma hipótese esse limite.

Caso seja necessário que uma determinada variável ultrapasse o limite de 64 Kbytes, o modelo de armazenamento a ser utilizado será o "Huge Model". Somente ele permite que uma variável individualmente ocupe mais do que um segmento de memória. Entretanto, convém ressaltar que para cada módulo do programa continua sendo necessário respeitar-se o limite de 64 Kbytes.

#### **2.2.4. Armazenamento do Código e dos Dados**

Chama-se código de um programa o conjunto de instruções executáveis, em linguagem de máquina, produzidas pelo compilador a partir do programa fonte. Já os dados não são instruções a serem executadas, mas simplesmente posições da memória utilizadas pelo programa para armazenamento de variáveis. Genericamente falando todos os

processadores fazem distinções entre as áreas de código e dados. Especificamente no caso do processador 8086, e sua família, é necessário que estes conjuntos estejam alocados em segmentos diferentes. Assim sendo, num programa compilado pelo FORTRAN 4.01, código e dados são armazenados em áreas separadas da memória do computador e além disso existem algumas outras características próprias do compilador em questão.

Para o código, qualquer que seja o modelo de memória a ser utilizado, cada módulo não pode ultrapassar o limite de um segmento. Isso significa que os módulos não poderão ter mais de 64 Kbytes e serão armazenados em seus próprios segmentos, isto é, não existem módulos que começam a ser armazenados em um determinado segmento e terminam em outro. Desse modo, o compilador pode gerar endereços de 16 ou 32 bits, dependendo do fato do módulo em questão estar sendo alocado dentro ou fora do segmento "default". Quando se tem apenas um segmento de código, obviamente a situação fica simplificada pois somente existirá código alocado no segmento "default" e todos os endereços gerados serão de 16 bits.

Para as chamadas de sub-rotinas e funções o compilador gera, normalmente, endereços de 32 bits, exceto para o caso de se estar trabalhando com "Medium Model". Entretanto, o usuário pode forçar que um determinado conjunto de rotinas, que se interrelacionam, ocupe um mesmo segmento (basta que elas estejam num mesmo módulo, ou seja, num mesmo arquivo fonte). Desse modo, através de comandos específicos, pode-se informar ao compilador essa situação conseguida e pela geração de endereços de 16 bits conseguir-se uma redução da quantidade de memória e tempo de execução. Portanto, cabe ao programador estudar convenientemente os módulos a serem montados de modo a se conseguir uma maior eficiência para o programa.

Quanto aos dados, o tratamento varia bastante, de acordo com o armazenamento definido. Para o "Medium Model"

a situação é a mais simples e eficiente. Todas as variáveis são alocadas no segmento "default" DS. Desse modo, o compilador pode simplesmente gerar endereços de 16 bits para todas essas posições.

Quando se trata do armazenamento "Large" a situação altera-se um pouco. Nesse caso, de modo semelhante ao que ocorre com os códigos, o total da área de dados pode ultrapassar 64 Kbytes. Entretanto, também aqui cada variável, individualmente, precisa estar alocada dentro de um mesmo segmento. Isso restringe o tamanho de cada variável a 64 Kbytes, mas permite ao compilador, apesar de gerar endereços de 32 bits para as variáveis alocadas fora do segmento "default", trabalhar com uma aritmética de 16 bits para encontrar a posição de um item dentro de uma variável. Isso se dá pois todas os itens de uma mesma variável ocupam o mesmo segmento.

Caso o modo de armazenamento seja "Huge", existem menos restrições, mas também menos eficiência. Neste caso, cada variável pode ultrapassar, individualmente, as fronteiras de um segmento. Desaparece então a limitação de 64 Kbytes para uma determinada matriz ou vetor. Entretanto, isso faz com que o compilador, além de ter a necessidade de gerar endereços completos de 32 bits, passe a trabalhar também com uma aritmética de 32 bits. Isso porque os itens de uma mesma variável podem ocupar dois ou mais segmentos diferentes. Então, apesar de ser o modo de armazenamento menos restritivo é também o menos eficiente, se bem que em testes realizados essa diferença revelou-se, para os casos usuais, pouco significativa. Entretanto, recomenda-se que a utilização de variáveis "Huge" seja feita com cuidado, limitando-se o seu emprego apenas aos casos em que sejam realmente necessárias.

Como última informação importante a respeito do armazenamento dos dados, deve-se ressaltar os casos de dimensionamentos auto-ajustados e variáveis passadas como parâmetros. Quanto aos dimensionamentos auto-ajustáveis,

deve-se verificar se a variável pode ultrapassar o limite de 64 Kbytes. Caso isso aconteça, é necessário informar ao compilador a possibilidade dessa ocorrência definindo-se a variável como "Huge". Caso contrário, o FORTRAN 4.01 definirá o armazenamento no modo "default", que depende de sua instalação. Já para os parâmetros passados como argumentos de sub-rotinas e funções, é importante que os argumentos formais sejam definidos com o mesmo tipo de armazenamento dos parâmetros que são passados. Caso algum engano seja cometido pelo programador a respeito dos detalhes acima colocados, as consequências serão extremamente danosas, pois os erros daí advindos são assistemáticos e não avisados.

#### 2.2.5. Técnica de "Overlay"

Um das mais interessantes maneiras de diminuir o efeito danoso de um código extenso para a limitação da capacidade de um programa é a técnica de "overlay". Normalmente, quando monta-se um programa destinado a uma tarefa simples e única, a quantidade de código gerada pelo computador não costuma atrapalhar de maneira significativa a capacidade do mesmo. Entretanto, para a execução de tarefas mais complexas a quantidade de memória utilizada para armazenamento do código pode ultrapassar em muito o limite do suportável.

Para se resolver, ou minorar, esse problema é que se criou esta técnica de grande utilidade. Através dela pode o programador definir que partes do código devem estar presentes na memória central do computador num determinado momento do processamento. As partes restantes continuam armazenadas em memórias periféricas, como discos rígidos, flexíveis, etc, aguardando o momento oportuno de serem carregadas para a memória central. Para um programa

complexo, que envolve a execução de variadas tarefas não simultâneas, o ganho é muito grande.

Essa vantagem é tão significativa que programas de análise estrutural que não se utilizam dessa técnica precisam ser repartidos em módulos estanques, comandados por um gerenciador. Entretanto, essa separação acarreta que toda e qualquer transmissão de dados entre os diversos módulos componentes precisa ser feita através das referidas memórias periféricas, pois ao ser carregado um novo módulo toda a área de dados do anterior estará perdida. Evidentemente, essa não parece ser uma situação das mais interessantes para a eficiência do processamento.

Com o uso do "overlay", a área de dados permanece inalterada, havendo troca do código presente na memória central e por consequência em execução num determinado momento. Se for feito com eficiência, o uso dessa técnica pode ser equivalente à troca de módulos mencionada no parágrafo anterior, obviamente com a vantagem de não haver perda da área de dados.

O FORTRAN 4.01 possui uma maneira muito eficiente de utilizar-se a técnica aqui descrita. Desse modo, o programa desenvolvido nesta pesquisa procurará, no limite do possível, lançar mão desse recurso, sendo que maiores detalhes a respeito dessa utilização estarão em item subsequente deste capítulo.

#### **2.2.6. Estruturas de Arquivos em Disco**

Para um programa de grandes dimensões e relativa complexidade, é fundamental a definição de formas corretas de armazenamento de dados em disco, pois é evidente que nem todos os dados gerados pelo programa podem coexistir na memória central do computador. Assim sendo, variáveis que serão utilizadas por partes subseqüentes do código, muitas



vezes precisam ser armazenadas para essa posterior utilização. Considerando-se que a eficiência desse armazenamento varia muito em função da estrutura definida, pode-se concluir que a escolha de uma estrutura inconveniente é extremamente danosa à própria eficiência do programa como um todo.

O FORTRAN 4.01 possui um total de seis estruturas para a utilização de arquivos em disco. Quanto ao formato, é possível definir até três situações distintas para os dados a serem armazenados : formatado, não formatado e binário.

No primeiro caso, os dados são gravados em disco, em caracteres ASCII, através de um formato definido pelo usuário. Isso significa que dentro desse arquivo estão dados que podem ser lidos diretamente através de um editor de textos qualquer. Esse formato só deve ser utilizado em casos onde seja realmente necessário, pois é uma estrutura que requer um tempo de processamento relativamente grande para leitura ou escrita.

No segundo caso, os dados são armazenados sem formato, em números hexadecimais. Obviamente, é uma estrutura adequada à transmissão de dados exclusivamente numéricos e que vão ser acessados por um outro programa, ou parte subsequente do mesmo, também escrito em FORTRAN.

Já no formato binário, o armazenamento é feito byte a byte, em ASCII ou hexadecimal. O acesso deve ser realizado da mesma maneira em que se deu a gravação, também em ASCII ou hexadecimal, dependendo do caso específico.

Já quanto ao acesso, pode-se escolher entre duas formas possíveis : sequencial e randômico, também chamado de acesso direto.

O arquivo de acesso sequencial é um conjunto de registros de comprimentos variáveis, colocados, como o próprio diz, sequencialmente no disco. É um acesso com menor tempo de processamento que o randômico, mas não permite que um dado registro seja lido, sem que todos os anteriores o tenham sido. Assim sendo, é um modo de acesso

adequado a conjuntos de dados que serão gravados e lidos de acordo com a mesma sequência.

Para o acesso direto ou randômico, existem algumas diferenças a serem consideradas em relação ao sequencial. Em primeiro lugar, ao contrário do caso anterior, os registros não podem apresentar comprimentos variáveis. Serão sempre comprimentos fixos definidos previamente pelo usuário. Em compensação, o acesso de um determinado registro dentro do arquivo pode ser realizado de forma direta, sem que tenham que ser lidos os anteriores. É uma forma de armazenagem adequada a conjuntos de dados que precisam ser acessados eventualmente de maneira diferente da qual foram gravados.

Sendo tão importante uma correta avaliação sobre o desempenho dessas estruturas de arquivos, optou-se pela montagem de um pequeno programa onde essas condições pudessem ser analisadas a nível quantitativo. Consiste o referido programa num conjunto de instruções que realiza um mesmo trabalho básico de gravação e leitura de um arquivo de dados, utilizando-se as seis estruturas mencionadas. Para todas elas o programa executa mil vezes a gravação de um conjunto de vinte números (reais de quatro bytes). Em seguida, é realizada a leitura desses mil conjuntos, contando-se, inclusive, o tempo de retorno do ponteiro ao início do arquivo, para os casos de acesso sequencial. No caso de arquivos formatados, foi adotado o formato F4.1, que deveria resultar num tamanho correspondente ao dos arquivos não formatados. Quanto aos arquivos randômicos, o tamanho do registro definido foi de 80 bytes (20 vezes quatro bytes). Os resultados obtidos foram organizados na tabela 2.1

Pela observação dos tamanhos de arquivos obtidos, pode-se concluir que não existe uma diferença significativa de eficiência entre as diversas estruturas, quanto a esse particular. Apenas a mencionar que os arquivos formatados normalmente resultariam maiores, pois o formato utilizado

neste teste dificilmente seria satisfatório para os casos práticos.

Estrutura	Tamanho (bytes)	Tempo (seg)
Sequencial/Não Form.	82.002	3.68
Randômico/Não Form.	80.000	6.27
Sequencial/Binário	80.000	3.57
Randômico/Binário	80.000	6.15
Sequencial/Formatado	82.000	38.40
Randômico/Formatado	82.000	42.35

tabela 2.1

Já quanto ao tempo de processamento, as diferenças são extremamente significativas. Pode-se perceber que os arquivos sequenciais não formatados e binários foram escritos e lidos em pouco mais da metade do tempo dos seus respectivos arquivos diretos. Já quanto aos formatados, o tempo de processamento é muito superior aos obtidos para os casos não formatados e binários.

Algumas conclusões podem ser tiradas dos resultados obtidos. Em primeiro lugar, os arquivos formatados devem ficar restritos às entradas e saídas do programa e outros usos onde seja absolutamente inviável a utilização de estruturas não formatadas (comunicação com programas compilados em outras linguagens, por exemplo). Entre as estruturas não formatadas e binárias, as últimas apresentam uma certa vantagem, tanto em tamanho quanto em tempo de processamento. Já quanto ao acesso, o sequencial é mais adequado para os casos onde possa ser usado sem grandes prejuízos, até porque a possibilidade de uso de registros de tamanho variável é muitas vezes bastante interessante.

## 2.3. ESQUEMA GERAL DE FUNCIONAMENTO DO PROGRAMA

### 2.3.1. Introdução

O módulo de análise do programa LASER é composto, atualmente, por um programa principal e 69 sub-rotinas e funções, perfazendo um total de 7818 linhas de programação em FORTRAN. Para facilitar a referência a esse conjunto, usar-se-á no decorrer deste trabalho o nome de chamada desse módulo, que é "LS".

Todo sistema estrutural apresenta algumas características comuns e outras próprias. Dentre aquelas que dizem respeito a um determinado sistema em particular, destaca-se a chamada biblioteca de elementos. O programa LS, em sua versão atual, possui oito diferentes tipos de elementos implementados. É a seguinte a relação desses tipos :

- tipo 1 - elemento treliça
- tipo 2 - elemento barra
- tipo 3 - elemento contorno
- tipo 4 - elemento membrana
- tipo 5 - elemento plano (estado plano de tensão, deformação e axissimétrico)
- tipo 6 - elemento tridimensional
- tipo 7 - elemento placa ou casca
- tipo 8 - elemento sapata

Além disso, também como característica especial, o programa coloca à disposição do usuário um minimizador da semi-banda da matriz de rigidez global, que atua opcionalmente sobre as estruturas.

## 2.3.2. Divisão do Programa em Blocos de Comandos

### 2.3.2.1. Considerações Iniciais

Os comandos do programa LS estão divididos em dez blocos distintos. Todos os blocos tem nomes de apenas dois caracteres e reúnem um conjunto de sub-rotinas e funções que tem objetivos relacionados. Cada sub-rotina ou função tem como primeiros caracteres de seu nome o nome do bloco que a contém. Desse modo, mesmo com um grande número de segmentos, torna-se mais nítida a própria estrutura do programa. A seguir estão resumidas algumas informações a respeito do conteúdo de cada um desses blocos, suas funções e outras características de interesse.

### 2.3.2.2. Bloco PR

Contém o programa principal e mais doze rotinas chamadas de utilitárias, ou seja rotinas usadas por todos os outros blocos do programa, num total de 1159 linhas de programação ( 15% do total ). O programa principal tem como função gerenciar todo o processamento, realizando a inicialização de variáveis, abertura de arquivos em disco, controle da capacidade de processamento, chamada das rotinas envolvidas em cada etapa e ainda controle do tempo de processamento total e por etapas. Quanto às doze rotinas, a seguir é apresentada uma relação com suas principais funções :

PRCALB : calcula a semi-banda necessária à montagem de um determinado elemento na matriz de rigidez global da estrutura.

PRCALT : calcula o tempo decorrido entre dois instantes

distintos do processamento do programa. A aquisição dos valores necessários a esse cálculo é realizada por uma sub-rotina do FORTRAN, diretamente do relógio interno da máquina.

PRELT : identifica e faz a chamada de um determinado tipo de elemento, tanto para montagem das matrizes e vetores iniciais quanto para cálculo das tensões e ou esforços finais.

PRCALC : verifica, para todas as etapas, se a capacidade do programa não está sendo ultrapassada pela estrutura que se está analisando. Caso isso ocorra emite mensagem e aborta o processamento.

PRCNOD : faz a leitura das características nodais da estrutura, ou seja, coordenadas, restrições e temperaturas nodais

PRTREL : comanda a chamada das rotinas de montagem de um elemento tipo treliça. Calcula tensão e esforço normal nesses elementos.

PRBARR : comanda a chamada das rotinas de montagem de um elemento tipo barra. Calcula tensões e esforços nesses elementos.

PRCONT : comanda a chamada das rotinas de montagem de um elemento tipo contorno. Calcula o esforço nesses elementos.

PRPLAN : comanda a chamada das rotinas de montagem dos elementos tipo membrana ou plano. Calcula tensões nesses elementos.

PRSOLD : comanda a chamada das rotinas de montagem de um elemento tipo sólido. Calcula tensões nesses elementos.

PRCASC : comanda a chamada das rotinas de montagem dos elementos tipo placa ou casca. Calcula tensões e esforços momentos nesses elementos.

PRFUND : comanda a chamada das rotinas de montagem de um elemento tipo fundação.

#### 2.3.2.3. Blocos TR, BR, CT, PL, SL, PC e FU

Nesses blocos, encontram-se agrupadas as rotinas que realizam os cálculos da matriz de rigidez, vetor de cargas consistente, matriz de relação tensão e ou esforço com deslocamentos e vetor de contribuições no sistema global para cada tipo de elemento. São as seguintes suas destinações específicas :

TR - elemento treliça  
BR - elemento barra  
CT - elemento contorno  
PL - elementos membrana e plano  
SL - elemento sólido  
PC - elementos placa ou casca  
FU - elemento sapata

Todos esses tipos de elementos serão estudados com detalhes nos dois próximos capítulos. Assim sendo, seria cansativo e pouco útil que dentro deste item se fizesse uma descrição mais detalhada de todas as rotinas que compõem esses blocos. Menciona-se apenas que elas representam a maior parcela do código, abrangendo um total de 5302 comandos ( aproximadamente 68 % do total ).

#### 2.3.2.4. Bloco MB

Neste bloco estão agrupadas as rotinas de minimização da semi-banda da matriz de rigidez. Engloba um total de 792 linhas de programa ( 10% do total ), distribuídas em 9 sub-rotinas. Realiza não só as operações de minimização propriamente dita, mas também tanto a preparação dos dados necessários como a implementação das modificações devidas à minimização conseguida.

Maiores informações a respeito do algoritmo utilizado para a execução da minimização estarão presentes em item posterior. Entretanto, neste ponto mencionar-se-ão as rotinas utilizadas, ressaltando-se suas principais funções.

MBCOMD : Comanda a execução da minimização, trabalhando com um verdadeiro programa principal do bloco.

MBMMMC : Monta a matriz de conexões, chamada MC. Essa matriz contém para cada nó a relação de todos os nós a ele conectados.

MBMVIG : Realiza o cálculo do "grau" de cada nó, armazenando essa informação no vetor IG. Entende-se por grau de um nó o número de nós a ele conectados.

MBPMMC : Opera sobre a matriz de conexões nodais, MC, de modo a organizar as conexões de cada nó de acordo com uma ordem crescente de graus.

MBCOMP : Identifica os "componentes" de uma estrutura, criando um vetor onde o componente de cada nó é armazenado. Entende-se por componente um conjunto de nós conectados apenas entre si.

MBMBND : Calcula a máxima diferença de numeração entre dois nós conectados. A banda da matriz está relacionada a essa



diferença.

MBGMIN : Gera o esquema de renumeração propriamente dito, baseando-se em matrizes e vetores criados pelas rotinas anteriores.

MBPMIR : Lê, modifica e grava a matriz IR, que contém os números das equações para cada grau de liberdade. Atua somente quando se consegue a redução da banda original.

MBPVIC : Lê, modifica e grava o vetor IC, que contém os números das equações onde as rigidezes dos elementos vão contribuir na formação da matriz global. De forma semelhante à rotina anterior, atua somente quando se consegue a redução da banda da matriz.

#### 2.3.2.5. Bloco SO

O conjunto de rotinas agrupadas neste bloco diz respeito à montagem e solução do sistema global de equações e ao cálculo e apresentação dos resultados obtidos. O total de linhas de programa envolvidas nessas operações é de 831 (também 10% do total). A seguir, uma relação das rotinas presentes e um resumo das suas principais funções :

SOVRFC : verifica todos os graus de liberdade presentes na estrutura definida. Caso encontre algum no qual nenhuma rigidez tenha sido alocada atribui para a diagonal da equação correspondente o valor  $10^{20}$ .

SOMONT : realiza a montagem da matriz de rigidez global da estrutura e dos vetores de carga em função das coordenadas globais. Trabalha com a matriz e a vetor divididos por blocos, dois de cada vez, cujo tamanho é definido levando-se em conta a situação mais desfavorável a ser

encontrada em termos de utilização da memória central da máquina.

SOSOLC : resolve o sistema de equações gerais da estrutura. Também aqui a matriz de rigidez e os vetores de carga são particionados por blocos para possibilitar a solução de grandes estruturas. Produz como resultado de suas operações os deslocamentos nodais em função do sistema global de referência.

SOESL : rearranja e imprime os deslocamentos nodais para os casos de carregamento definidos na estrutura.

SOTENS : calcula as tensões e ou esforços para cada elemento existente na estrutura através da multiplicação de matrizes e vetores previamente preparados.

### 2.3.3. Implementação dos Blocos em Esquema de "Overlay"

Já foi mencionado, no item 2.2.5 deste capítulo, que a técnica de "overlay" é de fundamental importância para uma atuação mais eficiente de um programa de análise estrutural. Neste item será apresentado o modo pelo qual a referida técnica foi utilizada pelo programa LS.

Inicialmente deve-se mencionar que para uma boa utilização do "overlay" é necessário que as rotinas que compõem seus blocos estejam agrupadas por critérios muito claros de interrelacionamento. Isso se deve ao fato de que se uma determinada rotina é chamada para a memória central do computador, todo o bloco no qual ela está inserida a acompanha nesse carregamento. Assim sendo, caso rotinas interrelacionadas estejam em blocos diferentes, o computador terá a necessidade de alojar e desalojar esses blocos sucessivamente, importando esse processo numa

significativa perda de tempo de processamento, principalmente para pequenas e médias estruturas.

Desse modo, o programador necessita avaliar corretamente o custo e o benefício de uma determinada divisão, para que o ganho em termos de memória não seja suplantado pelo custo em termos de tempo de processamento. É interessante mencionar que, no limite de utilização desta técnica, cada rotina que compõe o programa poderia ser um bloco de "overlay". Naturalmente, o módulo executável assim obtido seria extremamente otimizado em termos de necessidade de memória, mas seguramente inviável por um excesso de transferência de código do disco para a memória e da memória para o disco.

No caso do programa LS, os blocos apresentados no item anterior já foram montados levando-se em conta tais considerações. A figura 2.1 esquematiza o procedimento adotado. Através dela pode-se perceber que o único bloco que permanece na memória durante todo o processamento é o bloco PR. Assim foi definido pois estão contidos nesse segmento o programa principal e as rotinas utilizadas em diversas partes do programa. Caso esse segmento tivesse que sair da memória em algum instante, logo adiante teria que ser recarregado. Desse modo, o processamento teria seu tempo aumentado sem que a memória economizada fosse tão significativa ( o bloco PR compreende apenas 15% dos comandos do programa ).

Todos os demais nove blocos definidos são carregados para a memória apenas no instante de sua execução. A figura 2.2 mostra esquematicamente como o compilador trata esse fato. Em termos de memória alocada ao carregamento do código, existe uma parte destinada ao bloco PR, que estará sempre na memória. Quanto aos demais, o compilador reserva apenas um espaço suficiente para o carregamento do maior dos outros blocos. Desse modo percebe-se que a quantidade total de memória alocada para armazenamento do código é equivalente ao tamanho do bloco PR e ao do maior bloco

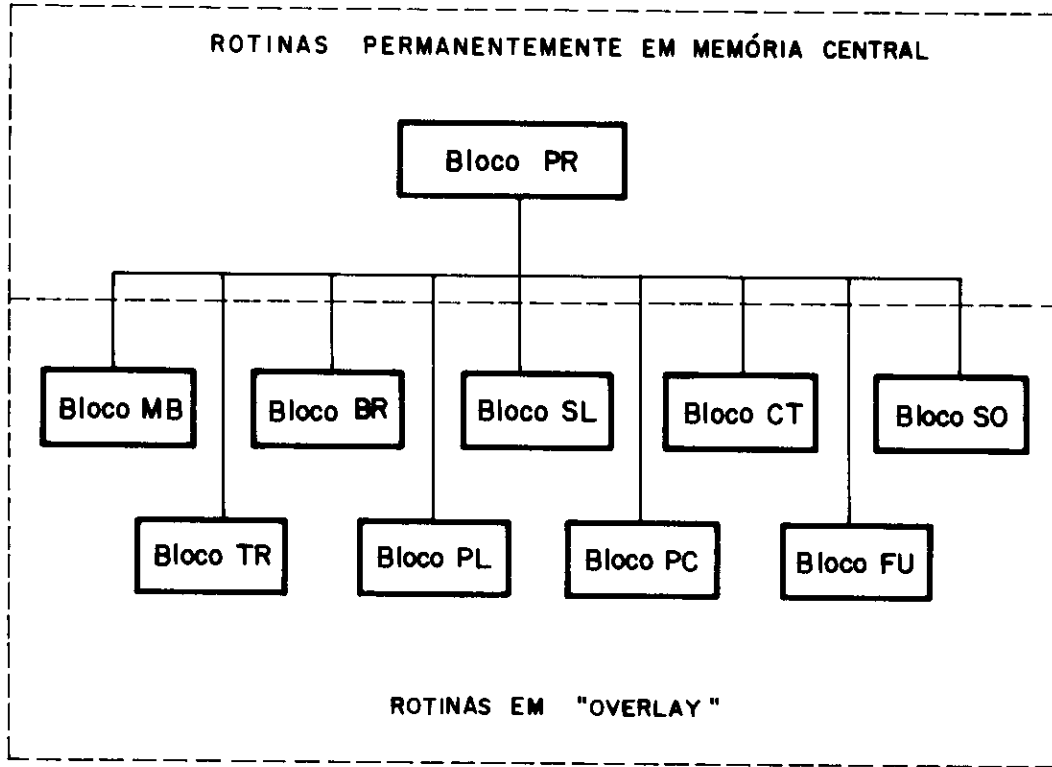


fig. 2.1 Organização dos blocos

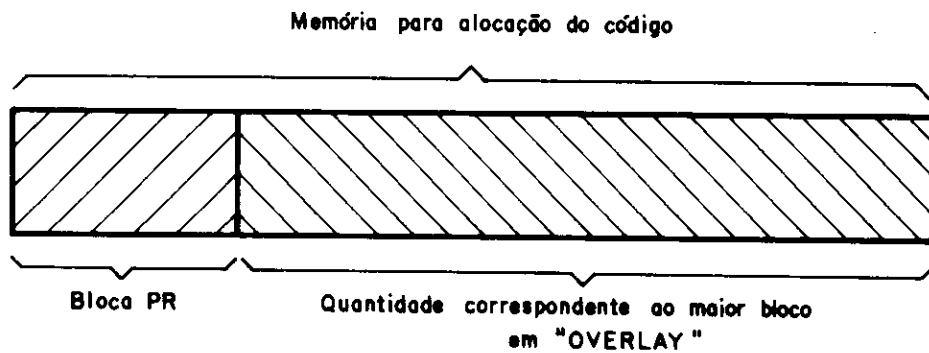


fig. 2.2 Esquema para alocação do código

entre os nove restantes, significando uma grande economia em relação à condição de carregamento simultâneo de todo o código. A memória não utilizada para armazenamento de código pode ser utilizada para a área de dados, aumentando-se a capacidade do programa.

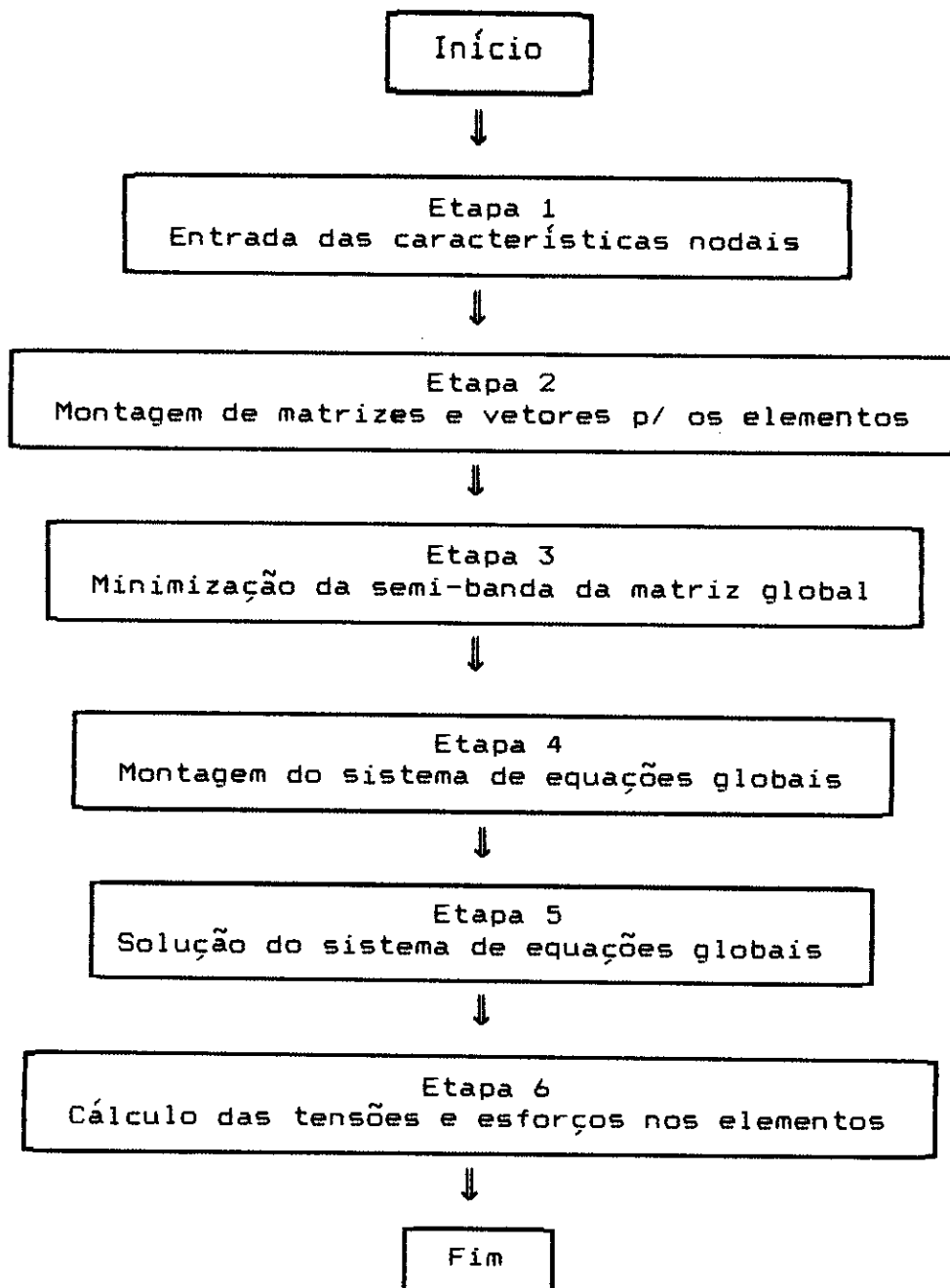
Se por um lado a vantagem em termos de memória economizada é evidente, o esquema utilizado não prejudica o tempo de processamento. Apesar de maiores detalhes a respeito do fluxo de processamento estarem colocados no próximo item, pode-se adiantar que, de acordo com o fluxo definido, todos os blocos em "overlay" são carregados para a memória apenas uma vez. Assim sendo, para os processamentos usuais, o tempo de carregamento do programa para a memória deve ser reduzido, em relação ao que seria verificado se todo o código fosse carregado. Isso se dá pois o bloco só é carregado no momento de sua utilização, se ela realmente ocorrer. Assim sendo, quando uma determinada estrutura não utiliza todos os tipos de elementos, ou mesmo o minimizador de banda, isso se traduz em uma vantagem: os módulos não utilizados não são em momento algum trazidos para a memória.

Outra vantagem significativa do procedimento adotado é quanto a futuros desenvolvimentos do programa, como por exemplo a introdução de novos elementos. Esses desenvolvimentos não irão alterar significativamente a capacidade do programa. Isso porque as rotinas que constituírem esse novo elemento, ou qualquer outro desenvolvimento a ser implementado, irão formar um novo bloco de "overlay", aumentando o total do código mas não a área necessária ao carregamento do programa na memória.

#### **2.3.4. Fluxo do Processamento**

Neste item a intenção é apresentar a maneira geral

como o programa organiza as suas diversas etapas de processamento para uma estrutura. O esquema abaixo reproduz esse caminho de forma simplificada.



Assim, em termos dos blocos definidos no item 2.3.2 e de acordo com o esquema de "overlay" do item 2.3.3, tem-se as seguintes configurações de código na memória central no

decorrer das diversas etapas :

Etapa 1 : somente bloco PR.

Etapa 2 : bloco PR mais um dos sete blocos correspondentes a um tipo de elemento (TR,BR,CT,PL,SL,PC ou FU).

Etapa 3 : bloco PR mais bloco MB.

Etapa 4 : bloco PR mais bloco SO.

Etapa 5 : bloco PR mais bloco SO.

Etapa 6 : bloco PR mais bloco SO.

Então, pode-se observar que realmente todos os blocos utilizados são carregados na memória central apenas uma vez, constituindo-se esse procedimento num dos pontos que ajudam a eficiência do programa.

Desse modo, em termos de esquema de funcionamento, resta apenas um detalhamento das fases presentes no fluxo de processamento, o que será feito em itens subsequentes.

## 2.4. ENTRADA DE CARACTERÍSTICAS NODAIS

### 2.4.1. Sistema de Referência e Graus de Liberdade

Para detalharem-se as características nodais de uma estrutura a ser analisada, é interessante tecer alguns comentários a respeito do sistema global de referência e dos graus de liberdade de um ponto nodal no espaço.

Quanto ao sistema de referência, a figura 2.3 mostra os três eixos cartesianos,  $X_1$ ,  $X_2$  e  $X_3$ , usados para a definição do domínio tridimensional. Nesse domínio, o nó genérico "N" é posicionado através de três coordenadas, respectivamente  $X_1(N)$ ,  $X_2(N)$ ,  $X_3(N)$ , conforme também se mostra na figura 2.3. Já os graus de liberdade de um ponto nodal no espaço, aparecem definidos na figura 2.4. São três translações, segundo  $X_1$ ,  $X_2$ ,  $X_3$ , e mais três rotações, em torno de  $X_1$ ,  $X_2$ ,  $X_3$ , respectivamente. Aliás, o programa sempre trata os graus de liberdade de um ponto nodal no espaço nessa ordem :  $TX_1$ ,  $TX_2$ ,  $TX_3$ ,  $RX_1$ ,  $RX_2$ ,  $RX_3$ .

### 2.4.2. Parâmetros a Serem Lidos

Nesta presente etapa a rotina envolvida é a PRCNOD. Para cada nó da estrutura a ser analisada é necessário fornecer as seguintes características :

N : número do nó  
IR(N,1) : restrição do nó segundo o grau de liberdade  $TX_1$   
IR(N,2) : restrição do nó segundo o grau de liberdade  $TX_2$   
IR(N,3) : restrição do nó segundo o grau de liberdade  $TX_3$   
IR(N,4) : restrição do nó segundo o grau de liberdade  $RX_1$   
IR(N,5) : restrição do nó segundo o grau de liberdade  $RX_2$



$IR(N,6)$  : restrição do nó segundo o grau de liberdade  $RX_3$   
 $X_1(N)$  : coordenada nodal segundo o eixo  $X_1$   
 $X_2(N)$  : coordenada nodal segundo o eixo  $X_2$   
 $X_3(N)$  : coordenada nodal segundo o eixo  $X_3$   
 $T(N)$  : temperatura do nó

Dessa lista, as coordenadas nodais  $X_1$ ,  $X_2$ ,  $X_3$  e a temperatura  $T$  são valores cujo significado é óbvio. Apenas sobre as restrições nodais seria necessário tecer algumas considerações adicionais. Esses seis números inteiros podem assumir valores de três tipos :

$0$  : o nó tem o grau de liberdade ativo, ou seja, é montada uma equação de equilíbrio para a coordenada considerada.

$1$  : o nó tem o grau de liberdade restrito, ou seja, não é montada a equação de equilíbrio que seria correspondente à coordenada considerada.

$NN > 1$  : o deslocamento nodal é constrangido, ou seja, a rigidez a ser alocada para o nó considerado, nesse grau de liberdade, é transferida para a equação do grau de liberdade correspondente do nó "NN". Esta opção é válida somente para nós ligados a elementos barra. Deve-se notar que o nó ao qual algum grau de liberdade será constrangido necessita ter uma numeração maior que 1. O resultado de tal procedimento será o mesmo deslocamento para os dois nós assim ligados, sempre no grau de liberdade considerado.

#### 2.4.3. Número das Equações para Cada Grau de Liberdade

Após a leitura dessas características, os vetores de coordenadas e temperaturas nodais, que são reais de quatro bytes, não são alterados e ficarão na área de dados para

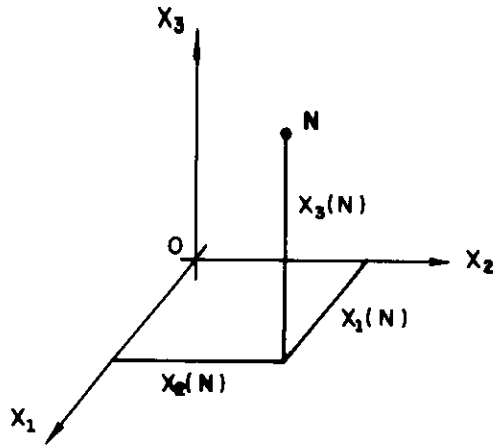


fig. 2.3 Coordenadas do nó "N"

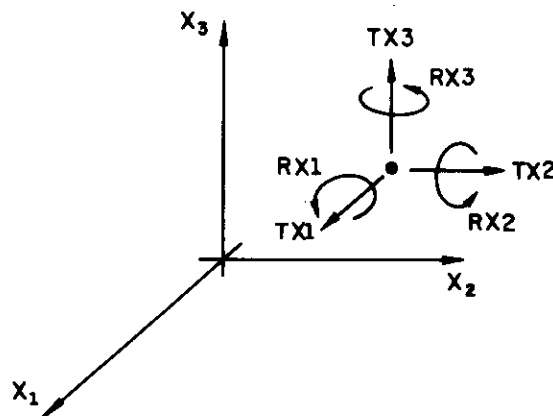


fig. 2.4 Graus de liberdade no espaço

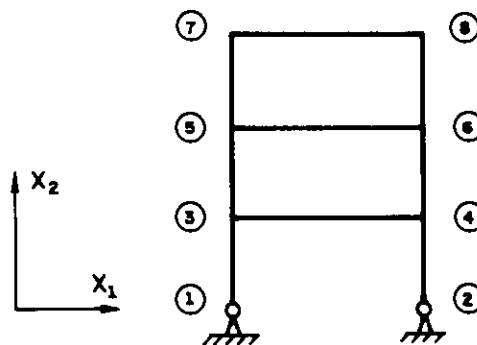


fig. 2.5 Pórtico no plano  $X_1 X_2$

utilização posterior. Já a matriz IR, composta de inteiros de quatro bytes, sofre um pós-processamento que é de certa forma padrão em quase todos os grandes sistemas de análise estrutural. É o seguinte o roteiro dessa alteração :

```
neq = 0
i de 1 até número de nós
j de 1 até 6
  se IR(i,j) = 0 então
    neq = neq + 1
    IR(i,j) = neq
  se IR(i,j) = 1 então
    IR(i,j) = 0
  se IR(i,j) > 1 então
    naux = IR(i,j)
    IR(i,j) = - IR(naux,j)
```

Isso significa que, para cada grau de liberdade da estrutura, já estará calculado o número da equação correspondente, sendo o zero indicador de grau de liberdade restrito e número negativo indicador de grau de liberdade constrangido. Após essas modificações, a matriz IR, além de permanecer na área de dados, é escrita em arquivo em disco para utilização posterior.

#### 2.4.4. Exemplo

Para um perfeito entendimento sobre a entrada das restrições nodais e posteriormente das modificações introduzidas pelo programa, vai-se tomar como base a estrutura mostrada na figura 2.5. Trata-se de um pórtico com geometria e carregamento contidos no plano  $X_1X_2$ . Além disso, os nós ligados por elementos barra horizontais devem apresentar o mesmo deslocamento na direção  $X_1$ , para simular

a existência de um diafragma rígido. Nesse caso as restrições nodais seriam as seguintes :

N	IR(N,1)	IR(N,2)	IR(N,3)	IR(N,4)	IR(N,5)	IR(N,6)
1	1	1	1	1	1	0
2	1	1	1	1	1	0
3	0	0	1	1	1	0
4	3	0	1	1	1	0
5	0	0	1	1	1	0
6	5	0	1	1	1	0
7	0	0	1	1	1	0
8	7	0	1	1	1	0

tabela 2.2

Após as modificações introduzidas pelo programa, na própria rotina PRCNOD, a matriz IR resultará :

N	IR(N,1)	IR(N,2)	IR(N,3)	IR(N,4)	IR(N,5)	IR(N,6)
1	0	0	0	0	0	1
2	0	0	0	0	0	2
3	3	4	0	0	0	5
4	-3	6	0	0	0	7
5	8	9	0	0	0	10
6	-8	11	0	0	0	12
7	13	14	0	0	0	15
8	-13	16	0	0	0	17

tabela 2.3

Portanto, todos os 17 graus ativos têm suas equações conhecidas e anotadas para uso nas demais rotinas.

## **2.5. MONTAGEM DE MATRIZES E VETORES DOS ELEMENTOS**

### **2.5.1. Considerações Gerais**

Cada tipo de elemento disponível para uso no programa LS tem obviamente seu próprio conjunto de código. Portanto, seus recursos e potencialidades são muitas vezes bastante diferentes, existindo entretanto determinadas características gerais que podem ser encontradas em todos esses tipos. Neste item serão estudadas exatamente essas características comuns, deixando-se as particulares para os dois próximos capítulos.

Mais especificamente, o interesse deste item estará concentrado em quais são e como serão utilizados os vetores e as matrizes montados para todos os tipos de elementos, não interessando no momento as particularidades dessa montagem. Antes de se passar a essas descrições, é importante mencionar que o número de graus de liberdade total de um elemento será representado por NGL e o número de tensões e ou esforços a serem calculados por NTE.

### **2.5.2. Matriz de Rigidez e Vetor de Cargas**

Em primeiro lugar, tratando-se de um programa baseado no processo dos deslocamentos, é fundamental que ao final das rotinas relativas aos elementos tenham sido calculados a matriz de rigidez e o vetor de cargas nodais consistentes. Esses dois conjuntos de dados serão utilizados na montagem do sistema global de equações.

No programa LS, para qualquer elemento, a matriz de rigidez será chamada RE e o vetor de cargas nodais de PE. Tanto a matriz RE, quadrada, como o vetor de cargas PE têm

dimensão NGL.

Considerando-se que a rotina que realiza a montagem do sistema de equações globais é única e trabalha com todos os tipos de elementos, é importante que tanto a matriz como o vetor sejam construídos em relação ao sistema global de referência. Assim sendo, para levar essas contribuições ao sistema de equações globais, basta saber em que graus de liberdade deve-se dar cada contribuição.

### 2.5.3. Vetor de Contribuição na Matriz Global

É fundamental que o montador do sistema de equações globais conheça em que graus de liberdade da estrutura a matriz de rigidez de um determinado elemento contribui. No esquema adotado por este programa, isso se dá através de um vetor onde esses graus de liberdade terão seus números armazenados. O vetor assim obtido é chamado de IC e tem dimensão NGL.

O vetor IC é montado a partir da matriz IR, mencionada no item 2.4, e dos nós que definem o elemento. Tome-se como exemplo a discretização, no plano  $X_1X_2$ , mostrada na figura 2.6. São dois elementos placa, P1 e P2, e quatro elementos barra, B1, B2, B3 e B4. As incidências desses elementos são as fornecidas na tabela 2.4. Já as restrições nodais para este problema aparecem na tabela 2.5. Após o processamento na rotina PRCNOD, a matriz IR resultante será a apresentada pela tabela 2.6.

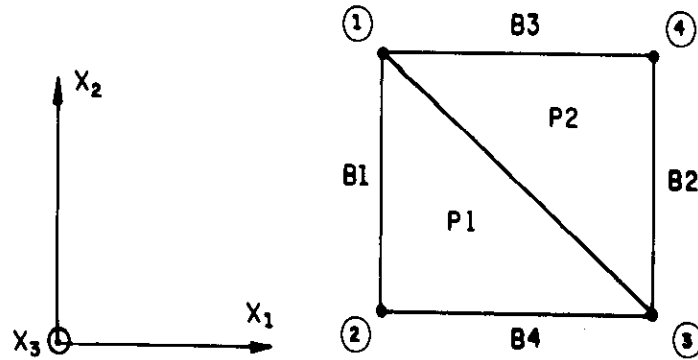


fig. 2.6 discretização no plano  $X_1 X_2$

Elemento	Nó 1	Nó 2	Nó 3
P1	1	2	3
P2	1	3	4
B1	1	2	
B2	4	3	
B3	1	4	
B4	2	3	

tabela 2.4

N	IR(N,1)	IR(N,2)	IR(N,3)	IR(N,4)	IR(N,5)	IR(N,6)
1	1	1	1	0	0	1
2	1	1	1	0	0	1
3	1	1	1	0	0	1
4	1	1	1	0	0	1

tabela 2.5

N	IR(N,1)	IR(N,2)	IR(N,3)	IR(N,4)	IR(N,5)	IR(N,6)
1	0	0	0	1	2	0
2	0	0	0	3	4	0
3	0	0	0	5	6	0
4	0	0	0	7	8	0

tabela 2.6

O elemento placa tem 3 graus de liberdade por nó. No caso considerado, respectivamente, a translação segundo  $X_3$  e as rotações em torno de  $X_1$  e  $X_2$ . Assim sendo, para o elemento P1, o vetor IC resultará no seguinte :

$$IC^T = \left\{ \begin{array}{ccc|ccc} 0 & 1 & 2 & 0 & 3 & 4 & 0 & 5 & 6 \end{array} \right\}$$

Já para o elemento P2, tem-se :

$$IC^T = \left\{ \begin{array}{ccc|ccc} 0 & 1 & 2 & 0 & 5 & 6 & 0 & 7 & 8 \end{array} \right\}$$

O elemento tipo barra, ao contrário, possui seis graus de liberdade por nó, ou seja, translações e rotações segundo  $X_1, X_2$  e  $X_3$ . Portanto, para o elemento B1, pode-se escrever :

$$IC^T = \left\{ \begin{array}{cccc|cccc} 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 3 & 4 & 0 \end{array} \right\}$$

Já para o elemento B2, tem-se :



$$IC^T = \left\{ \begin{array}{cccccc|cccc} 0 & 0 & 0 & 7 & 8 & 0 & 0 & 0 & 0 & 5 & 6 & 0 \end{array} \right\}$$

No caso do B3, resulta :

$$IC^T = \left\{ \begin{array}{cccccc|cccc} 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 7 & 8 & 0 \end{array} \right\}$$

Finalmente, para o B4, escreve-se :

$$IC^T = \left\{ \begin{array}{cccccc|cccc} 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 3 & 4 & 0 \end{array} \right\}$$

Dentro do vetor IC cada número representa a equação onde a rigidez correspondente do elemento deve contribuir. Os valores nulos indicam à rotina montadora que a rigidez do elemento correspondente àquele grau de liberdade não precisa ser carregada, pois na verdade a equação não existe.

#### 2.5.4. Matriz Tensão por Deslocamento e Vetor de Tensões Prescritas

As tensões e ou esforços para os elementos são obtidos pela simples multiplicação de uma matriz que relaciona essas grandezas a serem calculadas com os deslocamentos nodais obtidos no processamento e posterior adição de eventuais tensões e ou esforços iniciais.

Para esse procedimento, é necessário que as rotinas dos elementos construam a matriz que relaciona tensões e ou esforços com deslocamentos, chamada TD, e o vetor de tensões e ou esforços iniciais, TI. A matriz TD, genericamente falando, é retangular de dimensões NTE x NGL. Já o vetor TI tem dimensão NTE.

Informações sobre a montagem dessa matriz e desse

vetor podem ser encontradas no capítulo 3, que detalha as rotinas para cada tipo de elemento. Além disso, detalhes dos cálculos a serem realizados para a obtenção dos esforços e ou tensões finais aparecem no item 2.4.5 deste capítulo.

## 2.6. MINIMIZAÇÃO DA SEMI-BANDA DA MATRIZ DE RIGIDEZ GLOBAL

### 2.6.1. Introdução

A matriz de rigidez de uma estrutura analisada pelo Método dos Elementos Finitos é quadrada e simétrica. Além disso, em situação normal, os elementos diferentes de zero distribuem-se preferencialmente em torno da diagonal principal, conforme mostra-se na figura 2.7. Dessa maneira, é muito interessante que o armazenamento da matriz seja realizado de forma a evitar-se esse grande número de elementos nulos, e ainda aproveitando-se da simetria. Uma dessas formas de armazenamento é considerar apenas a faixa da matriz mostrada na própria figura 2.7 através da área escurecida.

A semi-banda da matriz, quantidade de elementos não nulos a partir da diagonal principal, depende do interrelacionamento entre os graus de liberdade da estrutura. Tome-se como exemplo uma estrutura com "n" graus de liberdade, cuja matriz de rigidez "R" encontra-se estilizada na figura 2.8. Vai-se admitir que existe um relacionamento entre os graus de liberdade "i" e "j", sendo  $j > i$ . Isso trará como consequência um valor diferente de zero, à direita da diagonal principal, na posição  $R_{ij}$ . Então, exclusivamente por causa desse relacionamento mencionado, a semi-banda da matriz de rigidez será  $b = j - i + 1$ .

Numa estrutura real, onde existam diversos graus de liberdade "i" e "j" interrelacionados, a semi-banda da matriz será o maior valor  $b = j - i + 1$ . Admitindo-se que cada nó da estrutura tenha "gl" graus de liberdade e que esses graus sejam numerados na sequência de numeração dos nós, então a semi-banda pode ser calculada pela expressão  $b = ( \text{dif} + 1 ) * \text{gl}$ . Nesse caso, "dif" é a maior das

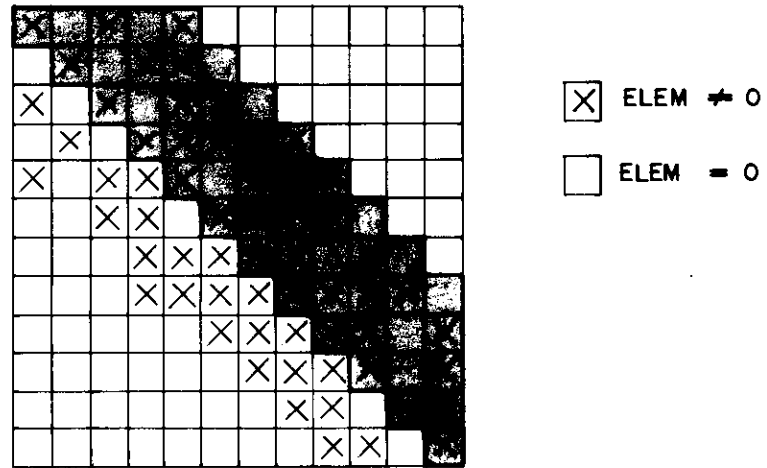


fig. 2.7 Matriz de rigidez quadrada e simétrica

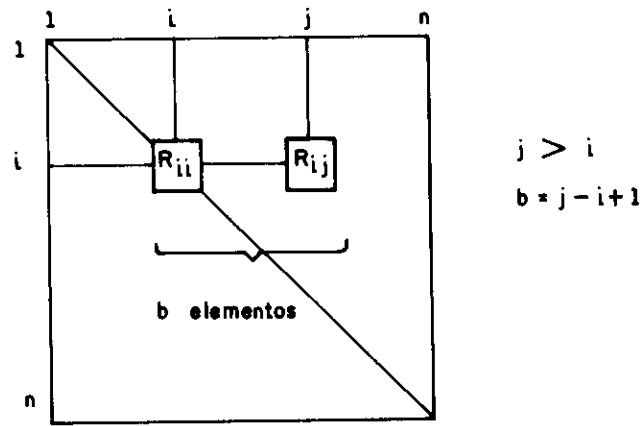


fig. 2.8 Matriz de rigidez R e semi-banda b

diferenças entre números de nós relacionados, ou seja, ligados através de um elemento qualquer.

Como já foi mencionado no primeiro capítulo deste trabalho, a minimização da semi-banda da matriz de rigidez global das estruturas a serem analisadas é considerado um ponto de grande importância para um bom programa de análise estrutural. Essa importância deve-se ao fato de que a geração de grandes estruturas não pode estar condicionada por detalhes como a numeração de nós de forma conveniente à obtenção de uma banda relativamente pequena. Pelo contrário, a numeração tem que atender às conveniências do usuário, não às do programa de análise.

Entretanto, uma banda relativamente grande para a matriz de rigidez de uma estrutura tem como consequência um significativo aumento do tempo de resolução do sistema de equações (o tempo de solução de um sistema de equações é proporcional ao quadrado da banda observada). Por sua vez, esse tempo de solução é o maior entre aqueles devidos à todas as etapas de processamento de uma grande estrutura. Desse modo, é necessário que a matriz de rigidez da estrutura apresente uma banda pequena, sem que o usuário seja o responsável por essa providência.

É interessante ressaltar que mesmo que o armazenamento da matriz de rigidez da estrutura seja em perfil (por exemplo altura efetiva de coluna, o que leva a uma banda variável) o minimizador continua tendo uma função fundamental. Isso porque se não houver um controle da numeração dos nós, também esses esquemas de armazenamento estarão inviabilizados. Deve-se lembrar que procurar a obtenção da banda mínima ou otimizar o perfil a ser obtido para a matriz armazenada com banda variável são procedimentos normalmente equivalentes.

É necessário que fique muito claro que quando se fala em renumeração de nós para obtenção da banda mínima da matriz de rigidez, supõe-se que inicialmente as equações do problema sejam numeradas em sequência, do nó 1 ao nó "n".

No caso, renumerar nós significa apenas uma maneira de executar um rearranjo nas equações, sem que na verdade a situação dos nós seja alterada. Assim sendo, renumerar nós é apenas uma forma prática de visualizar-se o esquema de renumeração das equações.

Alguns programas não tem rotinas para renumeração dos nós, mas definem as equações já considerando a conectividade dos elementos. É o caso dos programas SAP80 e SAP90. Os dois procedimentos mencionados, em última análise, são totalmente equivalentes, possuindo um em relação ao outro pontos vantajosos e desvantajosos que são óbvios. O primeiro gera uma numeração inicial para as equações, alterando posteriormente esse esquema, se isso for realmente desejável. Já o segundo, em qualquer hipótese, parte para a numeração das equações visando a otimização do armazenamento da matriz de rigidez.

Existe um grande número de procedimentos para a minimização da banda de uma matriz de rigidez. Inicialmente a maioria dos pesquisadores que preocuparam-se com esse assunto propuseram processos que trabalhavam diretamente na matriz, realizando permutações de linhas e colunas. As referências [2.3], [2.4], [2.5] e [2.6] são trabalhos representativos dessa fase.

Entretanto, após um trabalho pioneiro de Cuthill e McKee [2.7], o desenvolvimento do tema deu-se por procedimentos baseados na teoria dos grafos. Isso porque realizar permutações de linhas e colunas de uma matriz, corresponde a renumerar os vértices de um grafo. Desse modo, o desenvolvimento dos processos de minimização passou a ter uma base teórica mais consistente, a teoria dos grafos, ao mesmo tempo que algoritmos mais eficientes foram gerados. Como exemplos desse desenvolvimento têm-se as referências [2.8], [2.9], [2.10], [2.11] e [2.12]. Neste trabalho em especial, quando se menciona a minimização de banda de uma matriz, se está pensando em processos baseados na teoria dos grafos.

## 2.6.2. Princípios Básicos - Matriz de Conexão

Os algoritmos de renumeração de nós partem sempre de uma filosofia básica praticamente comum. A partir de um determinado nó, chamado semente, o novo esquema de numeração começa a ser construído, nó por nó, conexão por conexão, até que toda a estrutura esteja numerada. O problema é saber se a renumeração que começou numa determinada semente foi ou não eficiente para a obtenção de uma banda realmente pequena.

Os próximos sub-itens deste trabalho tecerão muitas considerações a respeito dessas estratégias de renumeração. Entretanto, existe um ponto comum a todas elas. Trata-se de uma matriz que contém os interrelacionamentos entre os nós da estrutura a ser analisada, aqui chamada matriz de conexão. Essa matriz contém, para cada nó da estrutura, todos os nós com que se relacionam, verdadeiramente traçando o caminho da renumeração que se deseja executar a partir de uma determinada semente.

Tome-se o exemplo mostrado na figura 2.9. As incidências dos oito elementos triangulares lá existentes podem ser resumidas na tabela 2.7.

Elemento	Nó 1	Nó 2	Nó 3
1	1	9	8
2	1	2	9
3	2	4	9
4	2	3	4
5	8	6	7
6	8	9	6
7	9	5	6
8	9	4	5

tabela 2.7

De alguma maneira essas incidências são passadas à rotina de minimização, que através delas pode montar a matriz de conexão. No caso do programa LS, essas conexões são lidas de um arquivo em disco. As rotinas que realizam a montagem de vetores e matrizes dos elementos passam para o arquivo os pares de nós conectados. No presente exemplo ter-se-ia o seguinte conjunto de dados :

```

elemento 1 : 1 9 / 1 8 / 9 8 /
elemento 2 : 1 2 / 1 9 / 2 9 /
elemento 3 : 2 4 / 2 9 / 4 9 /
elemento 4 : 2 3 / 2 4 / 3 4 /
elemento 5 : 8 6 / 8 7 / 6 7 /
elemento 6 : 8 9 / 8 6 / 9 6 /
elemento 7 : 9 5 / 9 6 / 5 6 /
elemento 8 : 9 4 / 9 5 / 4 5 /

```

A matriz de conexão, matriz MC, tem NN linhas e NMC colunas, sendo NN o número de pontos nodais e NMC o número máximo de conexões, inicialmente definido como 30. Caso esse número inicial não seja suficiente ele será aumentado automaticamente pelo programa LS, até que a memória central do computador seja esgotada. Lidos esses conjuntos de nós conectados, eles são utilizados pelo programa para a montagem de MC, que para o presente exemplo resulta na seguinte matriz de 9 linhas por 30 colunas :

$$MC = \begin{bmatrix}
9 & 8 & 2 & 0 & \dots & & & & & 0 \\
1 & 9 & 4 & 3 & 0 & \dots & & & & 0 \\
2 & 4 & 0 & \dots & & & & & & 0 \\
2 & 9 & 3 & 5 & 0 & \dots & & & & 0 \\
9 & 6 & 4 & 0 & \dots & & & & & 0 \\
8 & 7 & 9 & 5 & 0 & \dots & & & & 0 \\
8 & 6 & 0 & \dots & & & & & & 0 \\
1 & 9 & 6 & 7 & 0 & \dots & & & & 0 \\
1 & 8 & 2 & 4 & 6 & 5 & 0 & \dots & & 0
\end{bmatrix}$$



A linha "n" da matriz MC armazena as conexões do enézimo nó da estrutura. Por exemplo, tomando-se a linha quatro verifica-se que o nó 4 está conectado aos nós 2, 9, 3 e 5.

Um conceito de grande importância ligado às conexões nodais é o "grau". Define-se como grau de conexões de um nó, ou simplesmente grau de um nó, o número de nós a ele conectados. Assim sendo, para os nós da estrutura mostrada na figura 2.9, o nó número 1 tem grau 3, o 2 grau 4 e assim por diante. Os graus de todos os nós de uma estrutura estão, no programa LS, armazenados em um vetor chamado IG. Para o exemplo em questão o vetor IG resulta :

$$IG = \begin{bmatrix} 3 \\ 4 \\ 2 \\ 4 \\ 3 \\ 4 \\ 2 \\ 4 \\ 6 \end{bmatrix}$$

Após a montagem da matriz MC, resta discutir a estratégia de minimização a ser adotada. Os próximos sub-itens tratarão exatamente desse particular.

### 2.6.3. A Estratégia Cuthill-McKee

#### 2.6.3.1. Considerações Iniciais

Esta estratégia de minimização aparece originalmente concebida em [2.7]. Posteriormente sofreu algumas alterações, visando especificamente a minimização de perfís, [2.9]. Talvez seja a estratégia de minimização mais

utilizada e testada desde que esses algoritmos passaram a ser desenvolvidos.

A estratégia de Cuthill-Mckee basea-se na teoria dos grafos. Em poucas palavras, consiste em determinar uma certa quantidade de nós que tenham boas condições de tornarem-se sementes de esquemas eficientes. Logo em seguida esses esquemas são gerados e o melhor dentre eles, seja para a obtenção da banda mínima ou perfil mínimo, utilizado para a definição dos números das equações.

#### 2.6.3.2. Determinação das Sementes

Na presente estratégia, os nós a serem usados como sementes são aqueles que possuem a menor quantidade de nós da estrutura a igual distância. Isso quer dizer que são os nós de extremidades da estrutura, ou observando-se o grafo correspondente, aos vértices de extremidade. Tomando-se o referido exemplo da figura 2.9, pode-se definir como grafo correspondente o apresentado na figura 2.10.

Através da observação desse grafo, onde cada vertice é um nó da estrutura, pode-se definir a distância entre dois nós como sendo o menor número de segmentos entre vértices que ligam esses dois nós, ou os vértices correspondentes a esses nós. No caso a distância entre os nós 4 e 7 é três, pois três é o menor número de segmentos a serem percorridos para ir-se do nó, ou vértice, 4 ao nó, ou vértice, 7. Assim sendo, os nós que terão o menor número de nós à mesma distância serão os nós posicionados nas extremidades, no caso 3 e 7. Portanto, no caso considerado, as sementes para os esquemas a serem gerados serão os dois nós mencionados.

#### 2.6.3.3. Geração de Esquemas de Renumeração

A geração dos esquemas de renumeração de nós da

estrutura, ou vértices do grafo, é basicamente a mesma para qualquer algoritmo desse tipo. Trata-se de, adotado o nó ou vértice de partida, seguir as conexões armazenadas na matriz MC. A numeração é feita em seqüência, de um nó para as sua conexões imediatas, até que todos os nós estejam numerados. No caso da estratégia Cuthill-McKee, a seqüência de conexões para um determinado nó é montada seguindo uma certa hierarquia : inicialmente são considerados os nós de menor grau.

Considerando-se ainda o exemplo da figura 2.9, pode-se tentar explicar o algoritmo de geração mostrando-se os esquemas obtidos através da consideração das sementes 3 e 7. No caso, esses esquemas são apresentados na figura 2.11, a) para o nó 3 e b) para o nó 7. Em ambos os casos a maior diferença entre dois nós conectados foi de três. Assim sendo, qualquer dos dois esquemas pode ser utilizado para a renumeração dos nós da estrutura. Sobre esse esquema definido é que se daria a numeração das equações do problema.

#### 2.6.3.4. Pontos Negativos da Estratégia

O primeiro ponto negativo dessa estratégia de minimização é o grande trabalho computacional envolvido na determinação das sementes. Para que essa escolha seja realizada, é necessário que para cada nó da estrutura seja calculada a distância a todos os outros nós. A partir daí é que se pode definir o máximo número de nós a igual distância, sendo esse dado utilizado para a seleção do nó, ou dos nós, que possuem o menor número de nós a igual distância. Esses nós é que serão armazenados como as sementes dos esquemas a serem gerados.

Outro ponto negativo é que para cada semente é feita uma renumeração completa de todos os nós da estrutura. Após a geração de todos esses esquemas é que será escolhido o que

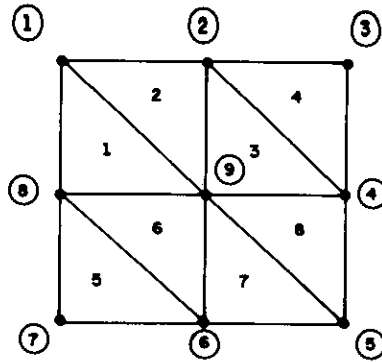


fig. 2.9 Estrutura com 9 nós e 8 elementos

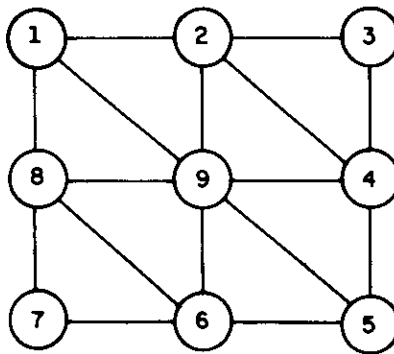
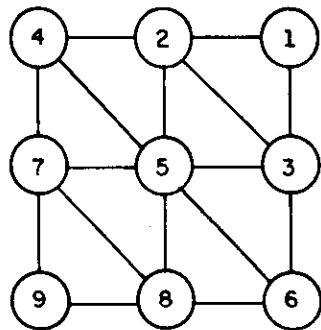
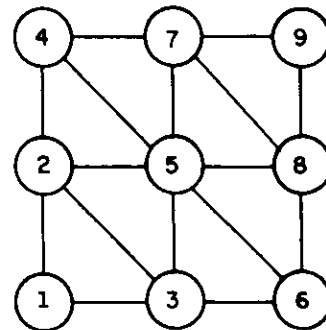


fig. 2.10 Grafo correspondente à estrutura definida



a) Semente 3



b) Semente 7

fig. 2.11 Esquemas de Renumeração Cuthill-McKee

melhor se adapta ao objetivo de gerar-se uma pequena banda na matriz de rigidez. Existem problemas para os quais podem ser selecionadas um grande número de sementes, causando um grande tempo de processamento.

Essas deficiências foram percebidas por alguns pesquisadores, que propuseram caminhos alternativos. Um exemplo disso é o trabalho de Gibbs, Poole e Stockmeyer [2.12]. Entretanto, mesmo que o tempo de processamento seja significativamente reduzido, sempre existe a possibilidade de uma falha na determinação das sementes, principalmente quando são consideradas algumas situações muito particulares. Desse modo, podem ser gerados esquemas que se afastem da banda mínima, tornando-se o algoritmo pouco eficiente.

#### **2.6.4. Estratégia R. Collins**

##### **2.6.4.1. Considerações Iniciais**

A presente estratégia encontra-se explicada com maiores detalhes em [2.10]. O procedimento de trabalho é extremamente simples. Trata-se de iniciar a renumeração dos nós tomando como semente cada nó da estrutura. Entretanto, essas renumerações podem não chegar ao final pois o algoritmo prevê o cálculo da diferença nodal obtida passo a passo. Caso tenha sido obtida uma diferença maior que a maior diferença de uma renumeração já pronta, o esquema é abandonado, passando-se a uma outra semente.

Como se pode perceber, a segurança do processo é bastante grande pois ao passar por todos os nós da estrutura, gerando esquemas de renumeração para cada um deles, o algoritmo garante a obtenção de um esquema de banda mínima, ou pelo menos muito próxima da mínima. Já o

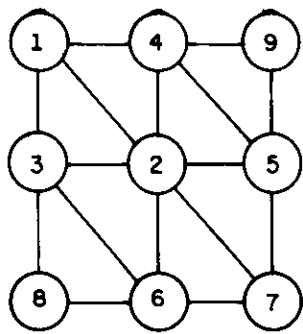
tempo de processamento, que à primeira vista deveria ser muito grande, revela-se bom em comparação com os obtidos por outros processos derivados da estratégia Cuthill-McKee. Isso porque aquelas estratégias demandam um grande trabalho computacional para determinar as sementes, enquanto o presente algoritmo passa imediatamente a gerar esquemas sem essa preocupação.

#### 2.6.4.2. Geração de Esquemas de Renumeração

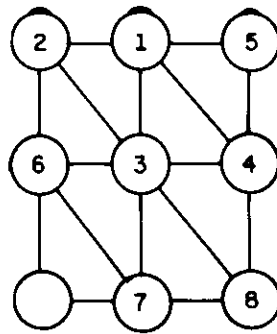
Para mostrar-se o procedimento da presente estratégia na geração de esquemas de renumeração vai-se tomar a mesma estrutura já considerada no método anterior e que se encontra mostrada na figura 2.9. Conforme foi mencionado, o algoritmo inicia renumerações para todos os nós da estrutura.

Essas renumerações seguem o procedimento básico já mencionado para o caso anterior. Adotado o nó de partida os novos números são colocados em sequência para as conexões do nó que se considera. Entretanto, ao contrário da estratégia Cuthill-McKee, não existe uma hierarquia entre as conexões do nó, sendo a numeração realizada pela ordem em que elas aparecem na matriz MC mostrada no item 2.6.2.

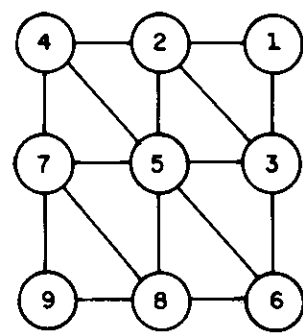
Tomando-se o exemplo mencionado, para o nó número 1 o esquema de renumeração seria completado, obtendo-se o resultado mostrado na figura 2.12 a). A máxima diferença entre número de nós obtida para esse esquema seria  $DIF = 5$ . Para o nó 2, o esquema seria abandonado na situação mostrada pela figura 2.12 b), ao ser obtida também uma diferença de 5. Já para o nó 3, o esquema chegaria ao final da renumeração, obtendo-se  $DIF = 4$ , conforme mostra-se em 2.12 c). A partir daí, todos os esquemas seriam abandonados antes de serem completados pois as diferenças obtidas igualam ou ultrapassam o valor 4. Para um perfeito entendimento da questão os esquemas parcialmente gerados



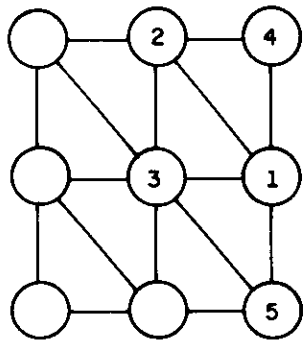
a) Semente 1



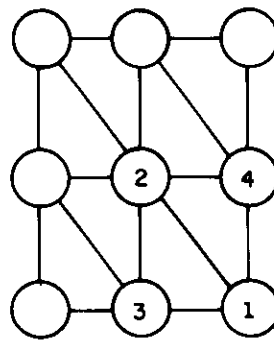
b) Semente 2



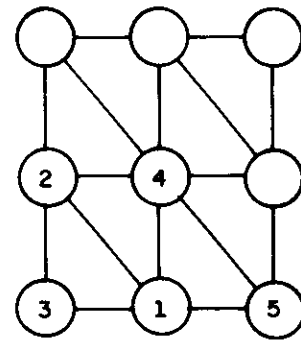
c) Semente 3



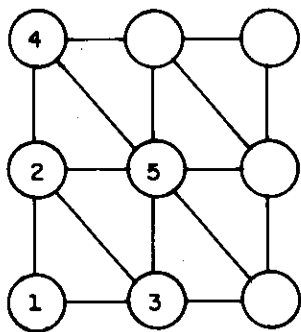
d) Semente 4



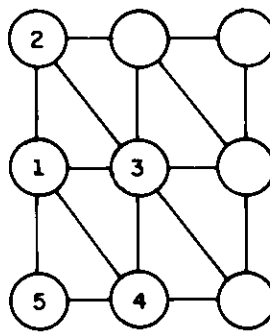
e) Semente 5



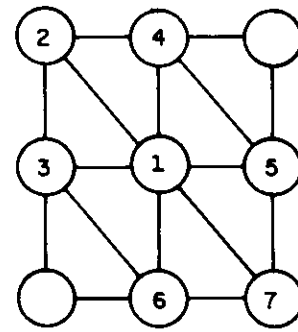
f) Semente 6



g) Semente 7



h) Semente 8



i) Semente 9

fig. 2.12 Esquemas de renumeração R. Collins

para os nós 4, 5, 6, 7, 8, e 9 também encontram-se na figura 2.12, letras d), e), f), g), h) e i), respectivamente.

#### 2.6.4.3. Ponto Negativo da Estratégia

Um ponto negativo da presente estratégia de renumeração é o fato das conexões de um determinado nó serem consideradas na ordem em que aparecem na matriz MC. Essa ordem obtida depende da numeração dos elementos que interligam os nós e portanto é aleatória. Assim sendo, o caminho de renumeração pode ser prejudicado, resultando em uma banda maior do que a que poderia ser obtida se as conexões fossem consideradas de acordo com algum procedimento lógico defensável.

#### 2.6.4.4. Adaptação Proposta Para a Renumeração

A partir da constatação da deficiência mencionada no item anterior, pode-se sugerir um procedimento alternativo. Trata-se de uma reorganização da matriz MC, colocando-se as conexões nodais lá armazenadas em ordem crescente de grau. Assim sendo nas primeiras posições apareciam os nós que possuem graus mais baixos, ou seja, que possuem um menor número de conexões. Essa providência é importante para melhorar o caminho da renumeração, tornando-o mais lógico e livre de condições aleatórias indesejáveis.

É relativamente simples justificar esse procedimento se se considerar que o grau de um nó indica o número de nós que se ligam a ele. Desse modo, organizar a matriz MC por ordem crescente de grau significa fazer com que durante a renumeração inicialmente sejam numerados os nós de menor grau. Portanto, em qualquer etapa do processo, serão inicialmente atribuídos números aos nós que se encontram



nas extremidades do grafo. Assim sendo é provável que a renumeração caminhe de uma forma mais consistente com a própria teoria adotada, resultando em menores valores para a banda.

### 2.6.5. Comparação de Resultados

Da leitura crítica da vasta bibliografia existente sobre o assunto de minimização da banda de uma matriz de rigidez, realmente pareceu que os dois métodos aqui apresentados podem ser considerados relativamente bons. Nesse caso, a avaliação deve envolver parâmetros como simplicidade, eficiência e também a segurança de que o processo não apresente resultados absurdos para determinados problemas específicos.

Assim sendo, foram testados os algoritmos mencionados para uma variada gama de problemas existentes. Nesses testes foram controlados o tempo de processamento e a máxima diferença nodal obtida no processo. Não se comparou diretamente a banda da matriz pois essa grandeza depende de alguns fatores que extrapolam a atuação dos minimizadores. Por exemplo, as restrições nodais de um problema tridimensional complexo influem nesse valor, mas os procedimentos de minimização não costumam levar em conta esses detalhes. Desse modo, o trabalho foi realizado controlando-se o parâmetro que realmente interessa às rotinas de minimização, que é a máxima diferença de numeração entre dois nós conectados.

Os resultados obtidos estão organizados na tabela 2.8. São 23 estruturas bastantes variadas (elementos com dois, três e quatro nós, consideração de diafragmas rígidos, etc), que devem produzir um universo de análise bastante significativo. Nessa tabela são apresentados : o número de nós do problema (Nós); a máxima diferença entre nós

conectados antes da atuação<sup>N</sup> do minimizador (Dif Orig); e para os três processos testados, a diferença conseguida (Dif) e o tempo de execução<sup>N</sup> em minutos e segundos (Tempo).

Est	Nós	Dif Orig	Cuthill		Collins 1		Collins 2	
			Dif	Tempo	Dif	Tempo	Dif	Tempo
1	40	37	33	00:03	36	00:01	33	00:01
2	44	41	37	00:03	40	00:01	37	00:01
3	34	15	11	00:01	14	00:01	11	00:01
4	42	39	31	00:01	32	00:01	30	00:01
5	90	87	82	00:24	86	00:06	82	00:07
6	56	53	48	00:07	52	00:02	48	00:02
7	379	116	20	00:24	21	00:17	20	00:16
8	379	365	24	00:29	24	00:20	24	00:20
9	413	386	27	00:37	27	00:26	27	00:26
10	483	440	22	00:38	23	00:26	22	00:25
11	515	438	28	00:57	30	00:45	28	00:44
12	548	507	32	00:58	33	00:43	32	00:44
13	577	391	28	00:53	29	00:39	28	00:39
14	592	501	26	01:35	27	00:41	26	00:42
15	696	524	37	01:56	37	01:16	37	01:16
16	728	622	32	01:29	32	01:02	32	01:02
17	753	658	23	02:07	22	01:17	22	01:20
18	933	720	40	02:48	39	02:11	39	02:08
19	1064	908	44	03:29	44	02:31	44	02:29
20	1115	1103	45	03:45	46	02:43	45	02:40
21	1172	1123	44	03:51	46	02:45	44	02:48
22	1179	652	43	04:46	46	02:48	43	02:35
23	1250	1079	54	05:56	54	04:05	53	04:10
Soma total			811	37:17	840	25:07	807	24:57

tabela 2.8

Resta ressaltar que "Cuthill" indica a estratégia apresentada no item 2.6.3. "Collins 1" o algoritmo original desenvolvido por R. Collins. Já "Collins 2" indica o

algoritmo de Collins, porém com as modificações propostas por este trabalho.

Pela observação dos resultados pode-se concluir que a diferença de nós obtida com os três processos apresenta poucas variações. Entretanto, a estratégia "Collins 2" apresentou em todos os exemplos processados a menor diferença, ora igualando-se a "Cuthill", na maior parte das vezes, ora igualando-se a "Collins 1", apenas duas vezes, ou mesmo conseguindo um valor inferior às duas, também duas vezes.

Quanto ao tempo de processamento, os valores verificados para "Collins 1" e "Collins 2" praticamente se equivalem. Entretanto, quando se compara esses valores com os obtidos para "Cuthill" verifica-se que a vantagem conseguida pelos primeiros é bastante significativa, situando-se em torno de 33 %.

Como conclusão geral menciona-se que a estratégia adotada foi a de R. Collins, sendo entretanto implementadas as modificações que constam no item 2.6.4.4.

## 2.7. MONTAGEM DO SISTEMA GLOBAL E CÁLCULO DOS RESULTADOS

### 2.7.1. Introdução

Neste item serão discutidas com maiores detalhes as etapas 4, 5 e 6 do fluxo de processamento apresentado no item 2.3. Trata-se da montagem e solução do sistema de equações globais e ainda cálculo e apresentação das tensões e ou esforços nos elementos.

É interessante observar que ao se chegar à etapa 4, montagem do sistema de equações globais, uma série de providências de grande importância terão sido cumpridas. Inicialmente, a entrada das características nodais e numeração das equações por grau de liberdade existente, armazenada na matriz IR.

Logo após, a formação de todas as matrizes e vetores dos elementos presentes. Desse modo, a matriz de rigidez RE, o vetor de cargas PE, o vetor de contribuição no sistema global IC, a matriz tensões e ou esforços por deslocamentos TD e ainda a matriz de tensões e ou esforços iniciais TI, terão sido montados e gravados para esta posterior utilização.

Por fim, caso essa tenha sido a opção do usuário, as rotinas de minimização de banda da matriz global já terão alterado a matriz IR e o vetor IC, de modo a expressar uma renumeração das equações previamente numeradas em sequência. Portanto, tudo está preparado para essas últimas etapas, que tratam da obtenção dos resultados propriamente ditos e que se encontram discutidas nos próximos sub-itens.

## 2.7.2. Armazenamento do Sistema de Equações Globais

Existem muitas formas de se armazenar a matriz de rigidez e vetor de cargas globais de uma estrutura. Normalmente esse armazenamento está intimamente relacionado ao processo adotado para a solução do sistema. A respeito desse tema encontra-se vasta literatura do mais alto nível. As referências [2.13], [2.14], [2.15], [2.16], [2.17] e [2.18] são apenas alguns exemplos a serem mencionados.

No programa LS o sistema de equações globais é montado considerando-se a matriz de rigidez simétrica, em faixa de banda fixa e armazenada por blocos. O vetor de cargas também é armazenado por blocos, sendo o número de equações de cada uma deles igual aos da matriz de rigidez. A figura 2.13 mostra esquematicamente essa técnica de armazenamento. Para maior clareza e concisão do assunto aqui apresentado, a nomenclatura dos parâmetros importantes é a que se segue :

neq : número de equações do problema  
nc : número de casos de carregamento  
lb : largura da semi-banda da matriz de rigidez  
neqb : número de equações por bloco  
nb : número de blocos

A técnica utilizada para a montagem e solução do sistema global de equações no programa LS é baseada na dos programas SAP4 e SAP5, desenvolvidos para grandes computadores. Obviamente que foram realizadas algumas modificações para tornar os algoritmos eficientes para utilização em microcomputadores. Mesmo assim, essa providência economizou um tempo considerável no desenvolvimento do sistema.

A verdade é que a procura de um bom procedimento para essas etapas demandaria um grande trabalho de pesquisa. Já o mencionado algoritmo encontra-se à disposição para uso e

sem dúvida nenhuma pode ser considerado eficiente e seguro, mesmo lembrando-se que técnicas mais modernas e refinadas devem produzir melhores resultados em termos de tempo de processamento. Entretanto, julgou-se que esse detalhe deveria ser deixado para um desenvolvimento posterior específico, onde um amplo estudo pudesse realmente indicar qual o procedimento mais adequado para a montagem e solução do sistema de equações globais do programa LS.

Uma providência de grande importância para a utilização de uma técnica de particionamento da matriz de rigidez por blocos é exatamente o cálculo do tamanho de cada um desses blocos. Ou seja, considerando-se a semi-banda fixa, quantas equações estarão armazenadas em cada bloco. O item 2.8 trata da área de dados do programa LS. Entretanto será importante adiantar que todo o armazenamento de grandes vetores e matrizes é feito numa área de trabalho de 400 Kbytes. Ressalta-se ainda que na montagem e solução do sistema de equações globais são utilizados reais de 4 e 8 bytes e inteiros de 4 bytes, sendo que o cálculo do número de equações de cada bloco deve levar em consideração esses detalhes.

Pela descrição do bloco S0 no item 2.3, percebe-se que duas rotinas do programa LS são as responsáveis pela montagem e solução do sistema global de equações : SOMONT e SOSOLC, respectivamente. No caso da SOMONT estarão ocupando a área de trabalho de 400 Kbytes, simultaneamente, dois blocos da matriz de rigidez, dois blocos dos vetores de carga e ainda NC vetores de quatro posições. No caso a matriz de rigidez e os vetores de cargas são reais de 8 bytes, sendo os NC vetores de quatro posições reais de 4 bytes. Assim sendo, a expressão que permite o cálculo do número de equações por bloco é a seguinte :

$$neqb_1 = \frac{100.000 - 4 nc}{4 (lb + nc)} \quad \dots(2.1)$$

Já no caso da rotina SOSOLC, deve-se considerar duas situações. Durante a triangularização estarão na memória central dois blocos da matriz de rigidez, dois blocos dos vetores de carga, reais de 8 bytes, e ainda um vetor auxiliar de "lb + neqb" inteiros de quatro bytes. Nesse caso o valor de neqb resulta :

$$\text{neqb}_2 = \frac{100.000 - \text{lb}}{4 (\text{lb} + \text{nc}) + 1} \quad \dots(2.2)$$

Para a retrossubstituição devem ser acomodados na área de trabalho um bloco da matriz de rigidez e um bloco do vetor de cargas, reais de 8 bytes, o vetor auxiliar de "lb + neqb" inteiros de 4 bytes e finalmente um vetor de "(lb - 2) nc + 2 neqb nc" reais de 8 bytes. Desse modo o número de equações por bloco pode ser escrito como :

$$\text{neqb}_3 = \frac{100.000 - \text{lb} - 2 (\text{lb} - 2) \text{nc}}{2 \text{lb} + 6 \text{nc} + 1} \quad \dots(2.3)$$

Obviamente o valor adotado pelo programa será o menor dos obtidos pelas equações 2.1, 2.2 e 2.3. Desse modo, considerando-se que deve-se respeitar um mínimo de duas equações por bloco, o problema será resolvido normalmente, qualquer que seja a banda, o número de equações e o número de vetores de carga.

Por fim, é importante ressaltar que o algoritmo adotado para a solução do sistema de equações globais do programa não opera em todos os elementos armazenados na faixa da matriz de rigidez. Isso porque normalmente existem muitos elementos zero numa coluna da matriz de rigidez antes que o primeiro elemento não nulo seja encontrado [2.13]. A operação que envolve esses elementos pode e deve ser descartada.

No procedimento aqui utilizado isso é feito através de

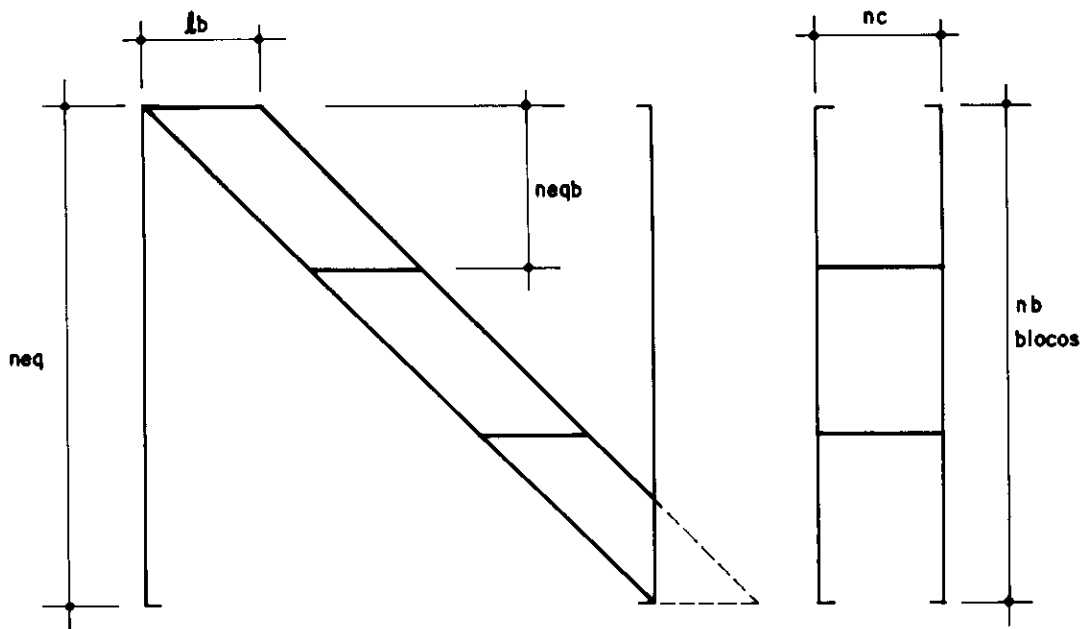


fig. 2.13 Esquema da matriz de rigidez e vetor de cargas

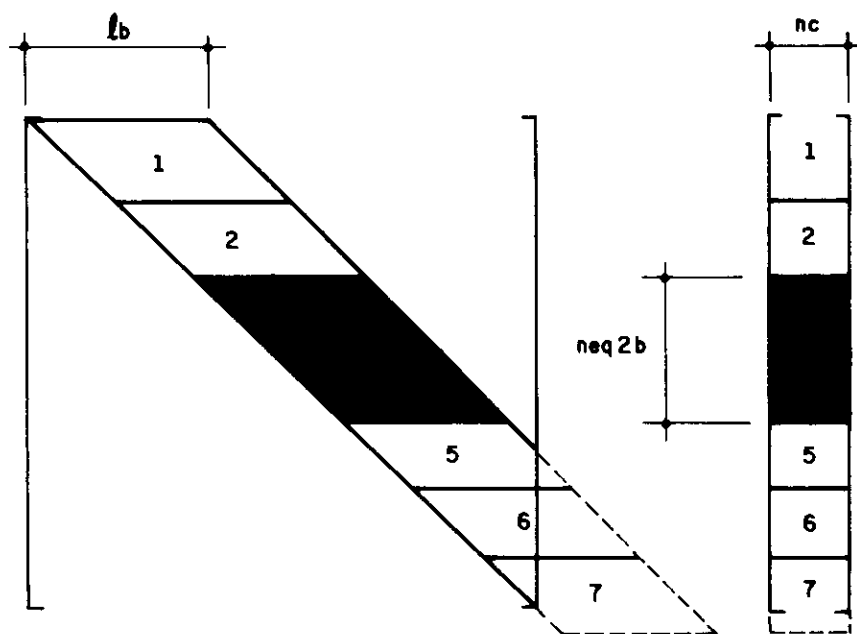


fig. 2.14 Blocos 3 e 4 da matriz e do vetor



um vetor auxiliar, calculado para cada bloco, e que contém a última posição não nula dentro de uma determinada coluna, a partir da diagonal. Desse modo, o algoritmo armazena toda a faixa, mas consegue uma redução no número de operações executadas, reduzindo o tempo de processamento.

### 2.7.3. Montagem do Sistema de Equações Globais

O montador a ser utilizado é perfeitamente geral em termos de utilização para os elementos presentes no programa. Ele trabalha com dois arquivos em disco, gravados durante o processamento das rotinas dos elementos. Trata-se dos arquivos ARQ2 e ARQ3, descritos com mais detalhes no item 2.9. Eles contém a matriz de rigidez e o vetor de cargas, RE e PE, de cada elemento e ainda o vetor de contribuições na matriz de rigidez, IC.

Conforme mencionado, a matriz de rigidez é montada por blocos, sendo que dois deles estarão simultaneamente na memória central. Além disso, também dois blocos de vetores de carga estarão sendo montados. A figura 2.14 mostra esquematicamente as partes da matriz e do vetor carregadas na memória central durante a montagem dos blocos 3 e 4 de uma determinada matriz de rigidez. De maneira simplificada, pode-se apresentar o algoritmo básico do montador pelas seguintes linhas de comando :

```

ne2b = 2 * neqb
i de 1 até ngle
    icn = 1 - IC(i)
    ii = IC(i) - neqb0
    se ii > 0 e ii < ne2b então
        j de 1 até nc
            B(ii,j) = B(ii,j) + PE(i,j)

```

...continua

j de 1 até ngle

jj = IC(j) + icn

se jj > 0 então

A(ii,jj) = A(ii,jj) + RE(i,j)

onde, além dos parâmetros já apresentados tem-se :

ngle : número de graus de liberdade do elemento

neqb0 : número da última equação do bloco anterior ( igual a zero se os blocos montados são os primeiros )

A : dois blocos da matriz de rigidez ( ne2b x lb )

B : dois blocos do vetor de cargas ( ne2b x nc )

Logicamente que o conjunto de comandos apresentados é uma simplificação do cerne do algoritmo aqui estudado. Ele representa o processamento a ser executado para cada elemento do problema resolvido, dentro de cada conjunto de dois blocos a serem montados. Assim sendo, admitindo-se um problema com "nb" blocos, todos os elementos definidos seriam lidos um número de vezes que corresponde à parte inteira da expressão  $nb/2 + 1$ .

Tentando minimizar todo esse acesso a disco, o algoritmo utilizado toma uma providência que pode ser importante no caso de problemas com um grande número de elementos e um grande número de blocos. Em primeiro lugar define-se como "próximos" conjuntos de dois blocos a serem montados a parte inteira da expressão  $\sqrt{nb} / 2 + 1$ . Assim sendo, ao identificar elementos que contribuem para blocos abrangidos por essa regra de proximidade, o programa grava esses elementos num arquivo temporário rascunho, no caso ARQ7. Depois, enquanto a rotina estiver montando esses blocos "próximos", as leituras serão feitas do arquivo rascunho ARQ7, reduzindo-se o acesso ao disco rígido e ganhando-se tempo de processamento.

Está claro que todo esse procedimento poderia ser

evitado se os arquivos que contém as matrizes e vetores dos elemento fossem de acesso direto ou randômico. Nesse caso poder-se-ia montar vetores que definissem quais elementos contribuem em quais blocos e o acesso ao disco rígido poderia ser bastante reduzido.

Entretanto, esse procedimento tem grandes dificuldades para ser implementado. Em primeiro lugar o tempo de acesso a arquivos randômicos é praticamente o dobro do verificado para arquivos de acesso sequencial, ver item 2.2. Para os problemas menores, que normalmente constituem maioria dos processamentos de um programa desse tipo, isso representaria uma perda importante.

Além disso, o arquivo randômico possui o comprimento de registro fixo. No caso de uma sistema que permite a utilização de vários tipos de elementos, isso é um ponto muito negativo. Ocorre que os registros precisariam ser dimensionados para os elementos que possuem as maiores matrizes e vetores e muito espaço em disco estaria perdido para os elementos que possuem pequenas matrizes e vetores. A única solução que poderia evitar esse desperdício seria a definição de um arquivo para cada grupo de elementos, com registro dimensionado especificamente para as matrizes e vetores desse tipo de elemento.

Por todas as colocações feitas é que se optou pela manutenção do arquivo sequencial, encarando-se como mal menor o acréscimo de acesso a disco assim obtido. Entretanto, esse ponto voltará a ser analisado com muito cuidado, pois os benefícios de se encontrar um procedimento ótimo para essa questão são realmente muito grandes.

Como última informação a respeito da rotina SOMONT, menciona-se que a gravação dos blocos da matriz de rigidez e do vetor de cargas é feita no arquivo temporário ARQ4. Inicialmente é gravado o primeiro bloco da matriz de rigidez, logo após o primeiro bloco dos vetores de carga, posteriormente o segundo bloco da matriz de rigidez e por fim o segundo bloco do vetor de cargas. Além disso, a

gravação desses blocos é feita por colunas, deixando-se o sistema pronto para ser acessado pela rotina de solução.

## 2.7.4. Solução do Sistema de Equações Globais

### 2.7.4.1. Considerações Iniciais

A técnica utilizada para a solução do sistema de equações baseia-se no processo de Gauss. Inicialmente, a matriz de rigidez e vetores de cargas são triangularizados e, posteriormente, através de um processo de retrossubstituição, os valores das incógnitas são determinados.

O algoritmo utilizado, quando na fase de triangularização, trabalha por linha da matriz de rigidez. Considerando-se um sistema de "n" equações e "n" incógnitas, com matriz simétrica, e ainda "c" vetores de carga, figura 2.15, as expressões gerais das modificações operadas podem ser escritas com sendo :

$$r_{ij} = r_{ij} - \frac{r_{li}}{r_{ll}} r_{lj} \quad \dots(2.4)$$

$$f_{ik} = f_{ik} - \frac{r_{li}}{r_{ll}} f_{lk} \quad \dots(2.5)$$

onde,

l : linha que está sendo triangularizada ( de 1 a "n-1" )

k : número do vetor de cargas operado ( de 1 a "c" )

Como pode-se observar na própria figura 2.15, quando uma determinada linha da matriz de rigidez está sendo triangularizada todo o triângulo composto pelos elementos abaixo da linha e à direita da diagonal principal precisa

ser alterado.

Já a segunda etapa do processo, a retrosubstituição, nada mais é que o cálculo das incógnitas "D" para um sistema  $R D = F$ , sendo a matriz R triangular superior. A expressão que permite esse cálculo é a que se segue :

$$d_{ik} = \left[ f_{ik} - \sum_{l=i+1}^n r_{il} \cdot d_{lk} \right] / r_{ii} \quad \dots(2.6)$$

onde,

i : linha da incógnita calculada ( de "n" a 1 )

k : número do vetor de cargas considerado ( de 1 a "c" )

Desse modo, o sistema estará resolvido, com todas as incógnitas calculadas.

#### 2.7.4.2. Implementação Considerando Matriz em Blocos

Evidentemente, também aqui todo o procedimento deve ser feito levando-se em conta que apenas uma parte da matriz de rigidez e do vetor de cargas estarão carregados em memória central. Mais especificamente dois blocos da matriz e dois blocos do vetor serão carregados simultaneamente.

Cada conjunto de blocos será lido do arquivo ARQ4 através de duas variáveis unidimensionais. Na primeira, variável A, ficam os primeiros blocos da matriz e do vetor, chamados de blocos básicos, e na segunda, variável B, os segundos blocos. Assim sendo, em cada variável, A ou B, as primeiras posições são ocupadas pela matriz de rigidez e as últimas pelo vetor de cargas. Esse armazenamento é diferente do esquema adotado pela rotina SOMONT, que apenas gravou os blocos de modo a serem assim acessados na presente rotina.

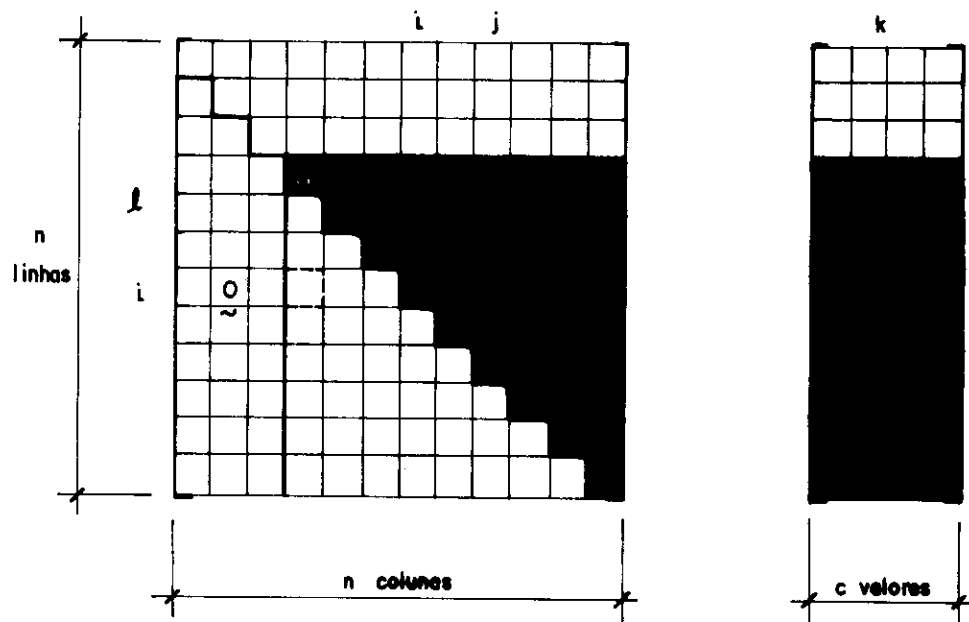


fig. 2.15 Sistema " $n \times n$ ", simétrico, com " $c$ " vetores

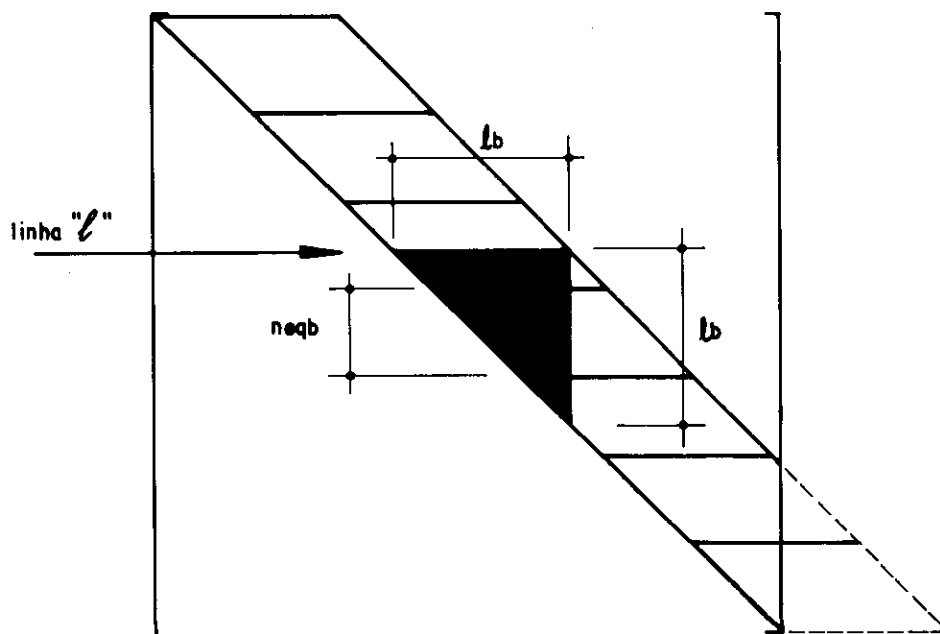


fig. 2.16 Alterações na triangularização da linha " $l$ "

#### 2.7.4.3. Etapa de Triangularização

Na verdade, o programa triangulariza um bloco de cada vez. O segundo bloco armazenado em memória central apenas recebe as modificações decorrentes da triangularização do primeiro. Esse processo pode ser melhor entendido quando observa-se, pela figura 2.16, a porção dos elementos que necessitam ser operados quando da triangularização da linha "1", no caso de uma matriz armazenada por faixa. Nesse caso, o número de blocos subsequentes a serem operados por efeito da triangularização da linha "1", nbs, pode ser calculado em função da largura da faixa, lb, e do número de equações em cada bloco, neqb, através da parte inteira da expressão :

$$nbs = \frac{lb - 2}{neqb} + 1 \quad \dots(2.7)$$

Como pode-se observar na expressão 2.7, se o número de equações por bloco for maior que a largura da faixa menos dois, "nbs" resultará igual a um. Nesse caso, apenas os dois blocos inicialmente carregados na memória necessitarão ser operados. Entretanto, caso "nbs" seja maior que um, o que ocorrerá quando a largura da faixa ultrapassar o número de equações por blocos, existirá um complicador adicional, pois apenas dois blocos podem ser armazenados simultaneamente na memória central. Nesse caso o algoritmo necessita salvar o primeiro bloco auxiliar em memória periférica, trabalhando então com o próximo, e assim sucessivamente até que os "nbs" blocos tenham sido modificados. A figura 2.17 apresenta um fluxograma simplificado para tornar mais claro esse procedimento.

#### 2.7.4.4. Etapa de Retrosubstituição

A retrosubstituição de valores para a determinação das

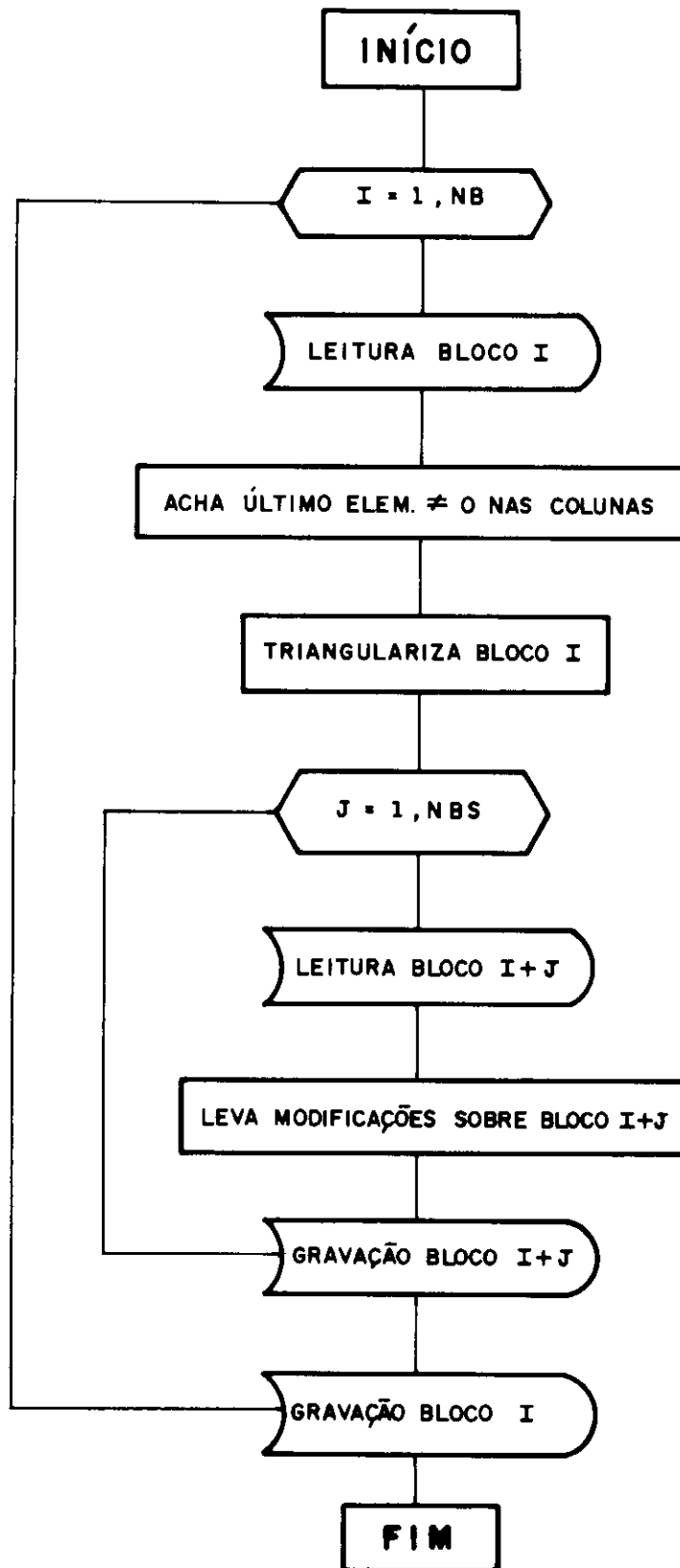


fig. 2.17 Fluxograma da etapa de triangularização



incógnitas, no caso os deslocamentos, é uma etapa que demanda um procedimento menor e mais simples do que a triangularização. Trata-se da aplicação da equação 2.6 da linha "neq" até a linha 1.

A única complicação adicional aqui verificada diz respeito ao fato do vetor de cargas estar particionado em blocos. De maneira semelhante à etapa de triangularização onde havia a necessidade de se alterar elementos que não faziam parte dos blocos em memória central, neste caso também ocorre de se necessitar de valores de deslocamentos já calculados e que se encontram em outros blocos do vetor de carga. Novamente essa situação ocorrerá quando, em termos aproximados, a largura da faixa for maior que o número de equações por bloco.

Para se contornar essa dificuldade, o programa carrega na memória central apenas um bloco da matriz de rigidez e do vetor de cargas. No espaço restante, é definida uma variável de "(nbs+1) neqb nc" posições onde estarão armazenados os deslocamentos calculados para blocos interrelacionados. Desse modo, a retrossubstituição será executada normalmente em memória central, para cada bloco, resultando num tempo de processamento relativamente reduzido.

Após concluída a retrossubstituição nos "nb" blocos do sistema, são conhecidos os deslocamentos de todos os "nc" casos de carregamento. Então, no arquivo ARQ2, esses deslocamentos estarão gravados para utilização na rotina que os organiza por nós da estrutura. Isso é feito por blocos, em ordem decrescente, de maneira que inicialmente são gravados os deslocamentos relativos a todos os carregamentos e que estejam armazenados no último bloco. Logo após os deslocamentos do penúltimo bloco e, assim por diante, até que todos os valores tenham sido gravados.

### 2.7.5. Saída dos Deslocamentos Nodais

A saída dos deslocamentos nodais é realizada pela rotina SODESL. É um procedimento bastante simples, baseado na leitura dos deslocamentos nodais, do arquivo ARQ2, e da matriz IR, que apresenta os números das equações do problema para cada grau de liberdade da estrutura e está armazenada no arquivo ARQ8.

Interessante de se mencionar é que o programa realiza um controle do número de casos de carregamento que será possível tratar simultaneamente, considerando-se a já mencionada área de trabalho de 400 Kbytes. Isso porque nas rotinas que realizam esses últimos procedimentos do programa, deve permanecer em memória central uma variável bidimensional, de reais 8 bytes, com tamanho relativamente grande. Essa variável, D, conterá os deslocamentos nodais, organizados por grau de liberdade e por caso de carregamento. Assim sendo, o número de linhas será "neqb x nb", número de equações em todos os blocos, e o número de colunas igual ao número de casos de carregamento cujas saídas estarão sendo providenciadas, "ncc".

Entretanto, a variável D não estará ocupando sozinha a área de trabalho de 100.000 posições de 4 bytes. Devem ser consideradas ainda mais duas variáveis. Elas dependem do número de casos de carregamento definido pelo usuário, "nc", que para estes casos não estará sendo alterado. Assim sendo, o limite para ncc será o seguinte :

$$ncc = \frac{100.000 - 4 nc - 2 neqb nc}{2 neq nb} \quad \dots(2.8)$$

Se o valor calculado pela equação 2.8 for menor que "nc", o programa não poderá processar todos os casos de carregamento simultaneamente. Nesse caso, inicialmente são considerados apenas os "ncc" primeiros casos de carregamento. Depois os próximos "ncc" casos e assim por

diante, até que todos tenham sido processados.

Simplificadamente, os comandos executados para a obtenção do vetor DD, de seis posições, que guarda os deslocamentos, já organizados para um determinado nó, são os seguintes :

```
i de 1 até nc
j de 1 até nn
  jj = 0
  k de 1 até 6
    DD(k) = 0
    l = IR(j,k)
    se l ≠ 0 então
      DD(k) = D(l,i)
      jj = 1
  se jj = 1 então imprime j,DD
```

Além da impressão do vetor DD, na unidade de saída dos resultados escolhida pelo usuário, a SODESL monta um arquivo com os valores dos deslocamentos para ser acessado por pós-processadores. Esse arquivo, com sufixo SD, está detalhado no item 2.9.

#### 2.7.6. Tensões e ou Esforços nos Elementos

Como última etapa de processamento do programa LS tem-se o cálculo das tensões e ou esforços nos elementos. De acordo com o esquema adotado, esse cálculo limita-se a algumas operações simples com matrizes e vetores previamente montados.

A operação começa no programa principal. Para cada grupo de elementos existente, é chamada a rotina de comando do tipo de elemento a ele correspondente, sendo indicado que se deseja o cálculo das tensões e ou esforços. Desse

modo, agora para cada elemento do grupo, a rotina de comando faz a chamada do procedimento SOTENS, que efetivamente executa o cálculo das tensões e ou esforços.

Essa rotina inicia-se com a leitura, para o elemento considerado, da matriz de relação tensões e ou esforços por deslocamentos, TD, e matriz de tensões e ou esforços iniciais, TI. Além disso, é acessado também o vetor de contribuição do elemento, IC, que relaciona o grau de liberdade local do elemento ao grau de liberdade geral da estrutura. A partir daí, considerando-se que os deslocamentos nodais já se encontram armazenados na área de trabalho desde a etapa anterior, basta realizar o cálculo propriamente dito, para cada caso de carregamento a ser considerado, através da expressão :

$$\underset{\sim}{TE} = \underset{\sim}{TD} \cdot \underset{\sim}{DE} + \underset{\sim}{TI} \quad \dots(2.9)$$

onde,

$\underset{\sim}{TE}$  : tensões e ou esforços calculados

$\underset{\sim}{DE}$  : deslocamentos dos nós do elemento

A saída dos valores calculados é realizada pela própria rotina de comando de cada tipo de elemento, através de formatação conveniente. Além disso, realiza-se a montagem da maior parte do arquivo que guarda os esforços dos elementos e será acessado pelos pós-processadores. Esse arquivo, de sufixo ST, é apresentado com detalhes no item 2.9.

Com esta última etapa, o processamento de uma estrutura que esteja sendo analisada está terminado. Resta apenas ao programa principal apagar os arquivos temporários do disco e tomar as últimas providências para o término do processamento.

## 2.8. ÁREA DE ARMAZENAMENTO DE DADOS

### 2.8.1. Introdução

A respeito da área de armazenamento de código deste programa, algumas informações relativamente completas foram apresentadas nos itens anteriores. Discutiu-se sua divisão em blocos e o carregamento do programa para a memória central do computador, utilizando-se uma técnica de "overlay".

Entretanto, muito pouco foi discutido até agora a respeito de uma área da memória que, em termos de extensão, seria ainda mais importante. Trata-se da área destinada ao armazenamento dos dados. Assim sendo, o objetivo deste item é apresentar toda a filosofia geral de uso da área de dados, inclusive fornecendo detalhes sobre as principais variáveis a serem utilizadas.

É interessante ressaltar que todo programa de computador, desenvolvido em linguagem estruturada, normalmente possui dois tipos de variáveis definidas: variáveis locais e variáveis globais. As variáveis locais são aquelas cuja existência limita-se a um determinado procedimento ou rotina. Já com as variáveis globais a situação é diferente, sendo seu conteúdo compartilhado entre várias rotinas.

Mais especificamente, quando se trabalha com FORTRAN, as variáveis locais são aquelas que, usadas dentro de uma determinada sub-rotina, não se encontram em "COMMON" nem passadas como parâmetro. As variáveis globais, por sua vez, são as que se encontram em "COMMON" ou passadas por parâmetros entre os módulos de programação. O importante de se observar é que uma variável local, apesar de ter sua existência restrita a apenas um módulo de programação, continua ocupando o mesmo espaço na memória durante toda a

execução do programa. Assim sendo, o programador experiente deve restringir o seu uso ao mínimo possível.

No caso do programa LS, o uso de variáveis locais praticamente está abolido. Além das matrizes e vetores utilizados ocuparem áreas comuns a todas as rotinas, até mesmo a grande maioria dos escalares segue esse princípio. Obviamente, que os nomes dessas variáveis são muitas vezes alterados de procedimento para procedimento, mantendo-se somente a mesma área de armazenamento. Esse tipo de ação leva a uma economia significativa da quantidade de memória a ser utilizada com os dados de caráter geral, sem nenhum prejuízo à clareza dos comandos.

## **2.8.2. Definição dos "COMMON's" do Programa**

### **2.8.2.1. Considerações Iniciais**

Como foi mencionado no item anterior, um perfeito gerenciamento da área reservada aos dados é fundamental para que o programa seja realmente eficiente. Nesse particular, acredita-se que grande parte da capacidade e desempenho do programa LS, que podem ser considerados excelentes em comparação com os programas similares existentes no mercado, devem-se ao esquema de utilização das variáveis.

Esse esquema baseia-se na definição de quatro áreas de "COMMON" onde a quase totalidade das variáveis envolvidas no processamento estão armazenadas. Essas áreas, rotuladas com nomes de três caracteres : PRC, PEL, PEI e CTL, estão definidas com maiores detalhes nos sub-itens a seguir.

### 2.8.2.2. Área com rótulo PRC

Este "COMMON" é uma grande área, de 400.008 bytes no total, onde três variáveis diferentes estão alocadas umas sobre as outras, através de uma declaração posterior de "EQUIVALENCE". As variáveis são as seguintes : um vetor AA, de 50.001 elementos reais de oito bytes ; um vetor A, de 100.002 reais de quatro bytes ; um vetor IA, de 100.002 inteiros de quatro bytes.

Nessa grande área estarão alocadas, cada uma a seu próprio tempo, a maioria das grandes matrizes e dos grandes vetores utilizados durante o processamento. De modo especial, nessa área encontram-se todas as variáveis que dizem respeito a dados de caráter global da estrutura. Por exemplo pode-se citar : coordenadas e restrições nodais, parâmetros dos elementos, blocos da matriz de rigidez global, deslocamentos nodais e mais um grande número de variáveis auxiliares ao cálculo executado.

A eficiência desse procedimento é enorme. Devido às definições sobrepostas de reais de oito e quatro bytes e ainda inteiros de quatro bytes, qualquer variável passada como parâmetro em qualquer das rotinas do programa pode ser armazenada sem que haja desperdício de memória. A única complicação adicional diz respeito aos limites dessas variáveis sobre a área total, cujo gerenciamento precisa ser feito pelo próprio programa.

Para maior clareza a respeito desse item, pode-se tomar um exemplo. Trata-se da chamada da rotina PRCNOD, onde são lidas as características nodais da estrutura, realizada pelo programa principal. Os primeiros comandos da mencionada rotina são :

```
SUBROUTINE PRCNOD ( NPN, IR, X1, X2, X3, T )  
INTEGER*4 IR(NPN,6)  
REAL*4 X1(NPN),X2(NPN),X3(NPN)
```

Como rápidas informações adicionais pode-se mencionar que IR é uma matriz que guarda as restrições nodais, X1, X2, e X3 armazenam as coordenadas, e T as temperaturas (ver item 2.4. para maiores informações). Além disso, NPN é o número de pontos nodais a serem lidos. A chamada, no programa principal, é a seguinte :

```
N1=1+6*NPN
N2=N1+NPN
N3=N2+NPN
N4=N3+NPN
CALL PRCNOD( NPN, IA(1), A(N1), A(N2), A(N3), A(N4) )
```

Como pode-se observar, todas as variáveis da sub-rotina que necessitam de uma grande capacidade de armazenamento estão alocadas dentro da área comum PRC. Inicialmente entram as 6 NPN posições de IR e depois as 4 NPN correspondentes a X1, X2, X3 e T. Assim sendo, a capacidade do programa, nessa etapa, estaria condicionada à relação :  $10 \text{ NPN} < 100.002$  , ou seja, no máximo 10.000 pontos nodais.

O procedimento mostrado nos parágrafos anteriores pode ser repetido para qualquer etapa do programa onde isso seja interessante. Como vantagens, pode-se citar o fato do armazenamento dos dados adquirir grande flexibilidade, sem que isso importe em dificuldades maiores para a clareza dos comandos a serem definidos nas subrotinas. Como ponto negativo, aliás perfeitamente aceitável, resta apenas que o gerenciamento dessa alocação precisa ser feito no próprio programa que a utiliza.

### 2.8.2.3. Área com rótulo PEL

Trata-se de uma área comum especialmente destinada ao armazenamento de variáveis que dizem respeito aos diversos tipos de elementos. Pode-se citar como exemplo a matriz de



rigidez, o vetor de cargas, o vetor de incidências, a matriz tensão/deformação, matrizes e vetores auxiliares do processamento, etc (para maiores informações a respeito ver item 2.5).

Para que essas matrizes e vetores, que existem basicamente em todos os tipos de elementos, não ocupem áreas diferentes da memória, onerando a quantidade total necessária para essa alocação, é que foi criada esta área com fim específico. Seu tamanho é de 20.000 bytes, atendendo assim às necessidades dos tipos de elementos atualmente implementados.

#### 2.8.2.4. Área com rótulo PEI

Essa área comum não aparece no programa principal. É utilizada apenas para transferência de informações entre as rotinas componentes do bloco de um determinado elemento. É usada nos seguintes elementos : barra, bloco BR ; membrana e estados planos , bloco PL ; sólido, bloco SL ; placa ou casca, bloco PC. Portanto não se utilizam desta área comum os elementos treliça, contorno e sapata, nos quais o presente "COMMON" sequer está definido.

Seu tamanho definido é de 10.000 bytes e, como se pode perceber pelas informações colocadas no parágrafo anterior, é um "COMMON" que tem duplo objetivo : salvar área de memória para outras utilizações e tornar o programa mais organizado.

#### 2.8.2.5. Área com rótulo CTL

No "COMMON" chamado CTL estão as variáveis escalares que precisam ter seus valores conhecidos, e às vezes alterados, em diversas etapas do processamento. Ao invés de

serem passadas como parâmetros para tal ou qual rotina que delas necessitasse, optou-se por este procedimento, que se acredita que melhore a organização do programa.

Através da área comum CTL, podem ser acessadas por todas as rotinas que necessitem variáveis como : número de pontos nodais, número de elementos de um determinado grupo, número de equações do problema, semi-banda da matriz de rigidez global, variáveis de controle de processamento, variáveis de controle de dispositivos de saída, opções de minimização, etc. Portanto, não é uma área definida com o intuito de economizar memória utilizada, mas sim de tornar o acesso às variáveis que o compõem possível de qualquer rotina do programa.

### 2.8.3. Variáveis Locais

Como já foi mencionado, praticamente não existem variáveis de caráter local no programa LS. Rigorosamente todas as matrizes e vetores presentes no programa estão alocadas em áreas comuns a todas as rotinas. Algumas poucas exceções dizem respeito aos índices de procedimentos repetidos, escalares para acumulação parcial de operações e variáveis auxiliares.

Mesmo assim, para conseguir-se uma organização mais adequada, definiu-se um certo roteiro para dar nomes a essas variáveis. Algumas das principais regras nesse sentido são destacadas a seguir :

**Índices de procedimentos repetidos :** do "DO" mais externo para o mais interno são usados índices de apenas um caracter, reservando-se para tanto as letras do alfabeto que o FORTRAN por "default" define como variáveis inteiras, ou seja, i, j, k, l, m e n, nessa ordem.

**Variáveis auxiliares** : para o caso de variáveis inteiras, são usadas IAUX ou IAUX1, IAUX2, etc. No caso de reais AUX ou AUX1, AUX2, etc.

**Índices auxiliares** : quando algum índice auxiliar é definido em função de dois ou mais índices já definidos anteriormente, procura-se combinar os nomes dos já existentes para formar o nome do auxiliar. Apenas como por exemplo pode-se citar :  $IJ = 2*I + J$ .

**Variáveis para acumulação parcial** : para um acumulador inteiro ISOM ou ISOM1, ISOM2, etc. Para o caso de reais SOM ou SOM1, SOM2, etc.

## 2.9. ARQUIVOS EM DISCO RÍGIDO

### 2.9.1. Introdução

O programa LS trabalha basicamente com dois tipos de arquivos em disco rígido : arquivos temporários e arquivos permanentes. Os arquivos temporários são aqueles criados pelo programa durante o processamento, para armazenamento de dados, e apagados automaticamente no final da análise. Já os arquivos chamados de permanentes, ou seja que não são apagados no final da análise, podem ser subdivididos em mais dois tipos : arquivos não formatados montados pelo programa com informações a serem acessadas pelos pós-processadores e arquivos formatados, a serem acessados pelo usuário para verificação dos dados fornecidos e resultados obtidos.

A seguir são fornecidas informações mais detalhadas, como nomenclatura, conteúdo e utilização, para cada um desses tipos mencionados. Ressalta-se que os dados a serem fornecidos para a análise também são lidos de um arquivo em disco. Entretanto, a respeito desse tópico em especial tratará o item 2.10.

### 2.9.2. Arquivos Temporários

#### 2.9.2.1. Considerações Iniciais

Os arquivos temporários, como foi mencionado no item anterior, são usados para armazenamento de dados durante a execução de uma análise, sendo criados e apagados automaticamente pelo programa, sem que o usuário deles tome

conhecimento. Entretanto, a eficiência desses acessos depende em grande parte a eficiência do programa como um todo pois a massa de dados envolvida nessas operações é bastante grande.

O programa LS utiliza um total de oito arquivos temporários, de ARQ1 a ARQ8. Todos foram definidos em relação ao formato e ao acesso de modo a trabalharem com o máximo de eficiência durante o processamento, de acordo com os conceitos apresentados no item 2.2.6. Os próximos itens detalham essas definições e também o conteúdo de cada um deles nas etapas do processamento onde são utilizados.

#### 2.9.2.2. Arquivo ARQ1

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : é gravado com a matriz de relação tensões e ou esforços por deslocamentos, TD, e matriz de tensões e ou esforços iniciais, TI.

**etapa 3** : inativo

**etapa 4** : inativo

**etapa 5** : inativo

**etapa 6** : é lido para obtenção das matrizes TD e TI, gravadas na etapa 2

### 2.9.2.3. Arquivo ARQ2

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : é gravado com a matriz de rigidez e com o vetor de cargas dos elementos, RE e PE, respectivamente.

**etapa 3** : inativo

**etapa 4** : é lido para obtenção das matrizes e vetores gravados na etapa 2.

**etapa 5** : é gravado com os deslocamentos nodais, à medida que a solução do sistema de equações vai-se processando. Logo após é lido para ser dada a saída desses deslocamentos.

**etapa 6** : inativo

### 2.9.2.4. Arquivo ARQ3

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : se a minimização da banda da matriz está ativada, é gravado com os pares de nós conectados pelos elementos. Caso contrário recebe o vetor de contribuição dos elementos, IC.

**etapa 3** : se a minimização está ativa, é lido para obtenção

dos pares de nós conectados e posteriormente gravado com os novos vetores de contribuição dos elementos, modificados pela minimização. Caso a minimização não esteja em ação, permanece inativo.

**etapa 4 :** é lido para obtenção do vetor de contribuição dos elementos, IC.

**etapa 5 :** inativo

**etapa 6 :** é lido para obtenção do vetor de contribuição dos elementos, IC.

#### 2.9.2.5. Arquivo ARQ4

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1 :** inativo

**etapa 2 :** se a minimização da banda da matriz está ativada, é gravado com o vetor de contribuição dos elementos, IC. Caso contrário, permanece sem utilização.

**etapa 3 :** se a minimização está ativa, é lido para obtenção dos vetores de contribuição dos elementos, IC, que serão modificados e gravados no ARQ3. Caso a minimização não esteja em ação, permanece inativo.

**etapa 4 :** é gravado com os blocos da matriz de rigidez e vetores de cargas globais.

**etapa 5 :** é lido para obtenção dos blocos da matriz de rigidez e do vetor de cargas globais.

**etapa 6** : inativo

#### 2.9.2.6. Arquivo ARQ5

Arquivo não formatado de acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : inativo

**etapa 3** : inativo

**etapa 4** : é gravado e lido, dentro da própria etapa, com as cargas aplicadas em nós da estrutura.

**etapa 5** : serve para gravação e leitura de vetor auxiliar à solução do sistema de equações globais.

**etapa 6** : inativo

#### 2.9.2.7. Arquivo ARQ6

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : é gravado com parâmetros relativos aos grupos de elementos, como tipo, número de elementos, quantidades de materiais e de seções, etc.

**etapa 3** : é lido para obtenção dos parâmetros gravados na etapa 2.



**etapa 4** : inativo

**etapa 5** : inativo

**etapa 6** : é lido para obtenção dos parâmetros gravados na etapa 2.

#### 2.9.2.8. Arquivo ARQ7

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : inativo

**etapa 2** : inativo

**etapa 3** : inativo

**etapa 4** : serve para gravação e leitura de vetor auxiliar à montagem do sistema de equações globais.

**etapa 5** : serve para gravação e leitura de vetor auxiliar à solução do sistema de equações globais.

**etapa 6** : inativo

#### 2.9.2.9. Arquivo ARQ8

Arquivo de formato binário e acesso sequencial. Sua utilização é a seguinte :

**etapa 1** : é gravado com a matriz que relaciona o número das equações com os graus de liberdade, IR.

**etapa 2 :** inativo

**etapa 3 :** se a minimização de banda está ativada, a matriz IR é lida e regravada com as alterações devidas. Caso contrário, permanece inativo.

**etapa 4 :** é lido para obtenção da matriz IR, a ser utilizada na montagem do sistema de equações globais.

**etapa 5 :** é lido para obtenção da matriz IR, a ser utilizada na impressão dos deslocamentos nodais.

**etapa 6 :** inativo

### **2.9.3. Arquivos Acessados Pelos Pós-Processadores**

#### **2.9.3.1. Considerações Gerais**

Existem 3 arquivos montados pelo programa LS para posterior utilização pelos pós-processadores. Todos eles possuem como prefixo o nome definido para o arquivo de dados da estrutura, aqui representado pela palavra PREFIX. Já o sufixo é determinado de acordo com o tipo de saída que eles contém. Os próximos itens fornecem informações mais detalhadas a respeito das características e conteúdos desses arquivos, ressaltando-se apenas que I4, R4 e R8 serão símbolos usados para indicar inteiros de 4 bytes, reais de 4 bytes e reais de 8 bytes, respectivamente.

### 2.9.3.2. Arquivo PREFIX.SG

O arquivo de sufixo SG guarda as informações relativas à geometria da estrutura. É um arquivo não formatado de acesso direto com tamanho de cada registro fixado em 44 bytes. Considerando cada linha como um registro, sua estrutura de gravação é a seguinte :

```
nn,nge
1,IR,X1,X2,X3,T
2,IR,X1,X2,X3,T
:
n,IR,X1,X2,X3,T
nte,ne,nmn
1,NI,NJ,...
2,NI,NJ,...
:
n,NI,NJ,....
```

} NGE vezes

onde,

nn : número de pontos nodais (I4).  
nge : número de grupos de elementos (I4).  
IR : matriz de restrições nodais (I4).  
X1,X2,X3 : coordenadas dos nós (R4).  
T : temperatura nodal (R4).  
nte : número do tipo dos elementos do grupo (I4).  
ne : número de elementos do grupo (I4).  
nmn : número máximo de nós por elemento (I4).  
NI,NJ,... : números dos nós que definem o elemento (I4).

Portanto, através do acesso aos dados armazenados no arquivo PREFIX.SG, a geometria da estrutura analisada pode ser conhecida e utilizada por qualquer programa pós-processador.

### 2.9.3.3. Arquivo PREFIX.SD

O arquivo de sufixo SD guarda os resultados dos deslocamentos nodais obtidos para cada caso de carregamento. É um arquivo não formatado de acesso direto com registros de 48 bytes. Conforme o adotado para o item anterior, considerando cada linha com um registro, pode-se definir sua estrutura como sendo a seguinte :

```
nn,nc
1,DX1,DX2,DX3,RX1,RX2,RX3
2,DX1,DX2,DX3,RX1,RX2,RX3
:
n,DX1,DX2,DX3,RX1,RX2,RX3
```

} nc vezes

onde,

nn : número de pontos nodais (I4).

nc : número de carregamentos (I4)

DX1,DX2,DX3 : translações nodais segundo X1,X2,X3 (R8).

RX1,RX2,RX3 : rotações nodais em torno de X1,X2,X3 (R8).

### 2.9.3.4. Arquivo PREFIX.ST

O arquivo de sufixo ST contém informações a respeito das tensões e ou esforços que ocorrem nos elementos. É um arquivo não formatado de acesso direto com registros de 104 bytes. Sua estrutura é um pouco mais complexa que a dos arquivos anteriores e por isso será apresentada por partes.

Os primeiros registros são montados no programa principal e têm a seguinte estrutura :

```
nge,nc
nr1
:
nrn
```

} nge números

onde,  
nge : número de grupos de elementos (I4).  
nc : número de casos de carregamento (I4).  
nr1 a nrn : número do registro onde tem início a gravação<sup>N</sup>  
das tensões ou esforços para cada grupo de elementos (I4).

Após esses registros iniciais, são gravadas as tensões e ou esforços propriamente ditos. Esses valores são gravados por elemento, para cada caso de carregamento e para cada localização existentes. Um registro genérico para essa parte do arquivo seria o seguinte :

ne,nc,nl,ntl,TE(1),TE(2),...,TE(ntl)

onde,  
ne : número do elemento (I4).  
nc : número do carregamento (I4).  
NL : número da localização (I4).  
ntl : número de tensões ou esforços (I4).  
TE : valores das tensões ou esforços (R8).

Tomando-se, como exemplo, uma situação específica de um grupo de apenas um elemento, com seis tensões e ou esforços em dois locais, e supondo a existência de dois casos de carregamento, os registros para esse grupo seriam os seguintes :

1,1,1,6,TE(1),TE(2),TE(3),TE(4),TE(5),TE(6)  
1,1,2,6,TE(1),TE(2),TE(3),TE(4),TE(5),TE(6)  
1,2,1,6,TE(1),TE(2),TE(3),TE(4),TE(5),TE(6)  
1,2,2,6,TE(1),TE(2),TE(3),TE(4),TE(5),TE(6)

#### 2.9.4. Arquivos de Saída de Dados e Resultados

O programa permite que as saídas dos dados fornecidos e resultados obtidos sejam feitas via console, impressora ou arquivo em disco, que aliás é considerado o modo "default". Caso a opção adotada seja realmente em disco, o programa LS montará na verdade dois arquivos : o primeiro de sufixo D, para conter as saídas referentes aos dados fornecidos para nós e elementos; e o segundo de sufixo R, com os resultados obtidos para os deslocamentos nodais e esforços e ou tensões nos elementos. Nos dois casos, o prefixo será sempre o nome do arquivo de dados da estrutura.

Tanto o arquivo PREFIX.D como o PREFIX.R são arquivos formatados e de acesso sequencial, podendo portanto serem acessados através de um editor de dados comum ou impressos por meio de um "COPY" ou comando equivalente. Cabe ressaltar que, para facilitar a observação através do editor e permitir a impressão em papel de 80 colunas, manteve-se o comprimento de cada linha em 78 caracteres.

## 2.10. ARQUIVO PARA A ENTRADA DOS DADOS

### 2.10.1. Introdução

O arquivo que contém os dados para análise é formatado e sequencial. Seu nome deve ser formado apenas por um prefixo, que pode ser escolhido livremente pelo usuário, apenas com o cuidado de não serem utilizados caracteres não permitidos pelo sistema operacional. O sufixo não deve ser definido. Sua montagem pode ser feita por um editor de dados comum ou através da utilização de um pré-processador do sistema LASER.

A montagem por um editor convencional é sempre muito trabalhosa, principalmente porque apenas as características nodais, seja de geometria ou carregamento, é que se mantém constante. Para os elementos, cada tipo a ser definido tem suas próprias características a serem fornecidas, e portanto necessariamente a sua própria formatação de entrada.

Assim sendo, pode-se admitir que o usuário padrão estará muito mais propenso a utilizar-se do módulo do sistema que realiza essa montagem, ou mesmo dos módulos que possuem geração automática para estruturas particulares. Então, considerando-se que seria cansativo, e de certo modo inútil, apresentar extensas explicações a respeito da formatação desse arquivo de dados, optou-se por um procedimento diferente. Trata-se de apresentar o módulo pré-processador, que realiza essa montagem com recursos gráficos acoplados.

O módulo aqui descrito tem seu esquema básico definido sobre o trabalho apresentado na referência [2.19]. Algumas modificações realizadas foram fruto de experiências adquiridas pela própria utilização do programa lá apresentado.

### **2.10.2. A Linguagem Computacional Adotada**

Para todos os pré e pós-processadores do sistema, ou seja, para os programas auxiliares que permitem geração de redes de forma automática e desenho das deformações, tensões e esforços na estrutura, em especial para o programa aqui exposto, a linguagem utilizada é o TURBO PASCAL 3.0, desenvolvido pela Borland, Inc..

Essa linguagem, muito utilizada nos meios científicos, possui recursos de grande valia para o tratamento de telas. Isso é fundamental para o desenvolvimento de um programa realmente eficiente que necessite trabalhar com os recursos gráficos do microcomputador.

Além disso, a linguagem dispõe de recursos de alocação dinâmica de variáveis, tratamento de áreas de dados em endereçamento absoluto, estruturação recursiva de rotinas e outros pontos que realmente permitem a construção de programas de grande desempenho.

Interessante ressaltar que as principais fontes de consulta à respeito da linguagem mencionada encontram-se apresentadas como referências [2.20] e [2.21].

### **2.10.3. Filosofia Geral do Programa**

Este programa pré-processador, ao final de sua execução, cria o arquivo formatado a ser acessado pelo módulo de análise LS. Essa filosofia de trabalho tem produzido ótimos resultados pois praticamente atende a todos os tipos de usuário do programa.

Para os iniciantes, ele praticamente dispensa a utilização de manuais complicados que muitas vezes desestimulam fortemente uma utilização mais frequente do programa. Esse desestímulo, ao longo do tempo, tende a



tornar-se uma verdadeira rejeição, optando o usuário por procedimentos mais simples.

Para aqueles que já possuem alguma experiência, ele continua sendo muito útil para uma primeira montagem da estrutura a ser analisada. Realizado o primeiro processamento, eventuais mudanças podem ser realizadas no arquivo de dados através de um editor comum, com grande facilidade e rapidez.

Finalmente, para o usuário de grande experiência sua utilização realmente tende a ficar mais restrita. Entretanto continua muito útil em casos de estruturas com possibilidades de geração e para detalhes não rotineiros. Por exemplo, elementos pouco usados, efeitos especiais de temperatura e muitos outros tópicos que não constituam o dia a dia do usuário.

Além disso, nem é necessário mencionar que essa armazenagem dos dados de entrada é fundamental para que não se limite a atuação do programa à de um procedimento interativo convencional, onde para cada execução torna-se necessário uma nova entrada de dados. Evidentemente, essa solução tornaria completamente inviável a utilização de um programa voltado para a análise de grandes estruturas.

A criação do mencionado arquivo texto ao final da entrada de dados chama a atenção para um detalhe de grande importância que é o armazenamento desses dados enquanto se processa o restante das entradas. O PASCAL oferece nesse ponto um recurso interessante que é a alocação dinâmica de variáveis. Através desse procedimento todos os dados já adquiridos se encontram em memória central, podendo ser acessados com grande rapidez e eficiência, sem que se necessite recorrer a periféricos. Ao final da entrada, ou em qualquer momento que se desejar, esses dados podem ser armazenados em disquetes ou discos rígidos através de um simples comando que indique a necessidade de se "salvar" as entradas já feitas.

Além dessas considerações, deve-se mencionar que para

se obter um processamento altamente eficiente torna-se necessária a subdivisão do programa em um determinado número de subprogramas comandados por um gerenciador. Isso proporciona um carregamento muito mais rápido para a memória central, pois os módulos não utilizados não são carregados, e permite que se adicione recursos extras ao programa, sem um comprometimento da quantidade de memória central disponível ao armazenamento dos dados. Essa substituição dos programas em execução se faz através da técnica de "Overlay", já mencionada em itens anteriores deste capítulo.

Como último detalhe a ser mencionado, tem-se a questão das mensagens e textos de ajuda que são opcionalmente oferecidas aos usuários. Essas mensagens se encontram armazenadas em arquivo e, se assim for determinado, são carregadas para a memória passando a ser armazenadas na área de "Heap", através de alocação dinâmica. Caso um comando de ajuda seja acionado essas mensagens são transferidas para a área reservada ao "buffer" de vídeo, aparecendo de modo instantâneo na tela do computador.

Assim sendo, pode-se resumir a estrutura de funcionamento proposta para esse programa através da figura 2.18, onde se mostra a interação entre o programa editor e os dispositivos de entrada, armazenagem e saída dos dados.

#### **2.10.4. Organização do Programa**

O presente programa está dividido em seis módulos, sendo um gerente do processamento e cinco dedicados a partes específicas do arquivo a ser montado. O módulo gerente, como seria de se esperar, realiza todo o comando do processamento, distribuindo tarefas e realizando chamadas dos módulos específicos.

O segundo módulo cuida da entrada de características

nodais. Essas características ( coordenadas, temperatura, restrições, e carregamentos ) podem ser fornecidas nó por nó ou através de rotinas de geração. Todos esses procedimentos encontram-se embutidos nesse único módulo.

Os quatro módulos restantes dizem respeito aos oito tipos de elementos cuja utilização é permitida pelo programa LS. Eles cuidam de gerenciar as entradas relativas a materiais, propriedades geométricas, incidências, carregamentos e demais parâmetros particulares que digam respeito a um tipo de elemento em particular. O terceiro módulo cuida das entradas relativas aos elementos unidimensionais, treliça e barra. O quarto diz respeito aos elementos bidimensionais, membrana, plano e placa/casca. O quinto cuida do único elemento tridimensional definido no programa, o sólido. Por fim, o sexto módulo é que permite a entrada dos elementos que podem ser vistos como especiais : contorno e sapata.

Essa organização se revelou eficiente pois permite ao usuário um acesso rápido a qualquer rotina de entrada sem que todas elas estejam presentes simultâneamente em memória central. Entretanto é importante ressaltar que durante o uso do programa praticamente não se percebe essa divisão, sendo que todo o procedimento de comando desta ou daquela rotina é feito a nível de módulo gerente, sem qualquer atuação do usuário. A figura 2.19 ilustra esquematicamente a organização adotada.

#### **2.10.5. Organização dos "Menus"**

Dentro do programa editor existem basicamente dois tipos de menus : os menus de opção, assim chamados pois permitem ao usuário optar por uma dentre as várias funções de um módulo ; e os menus de controle, assim chamados pois executam dentro de cada opção as funções de controle, como



fig. 2.18 Esquema de funcionamento

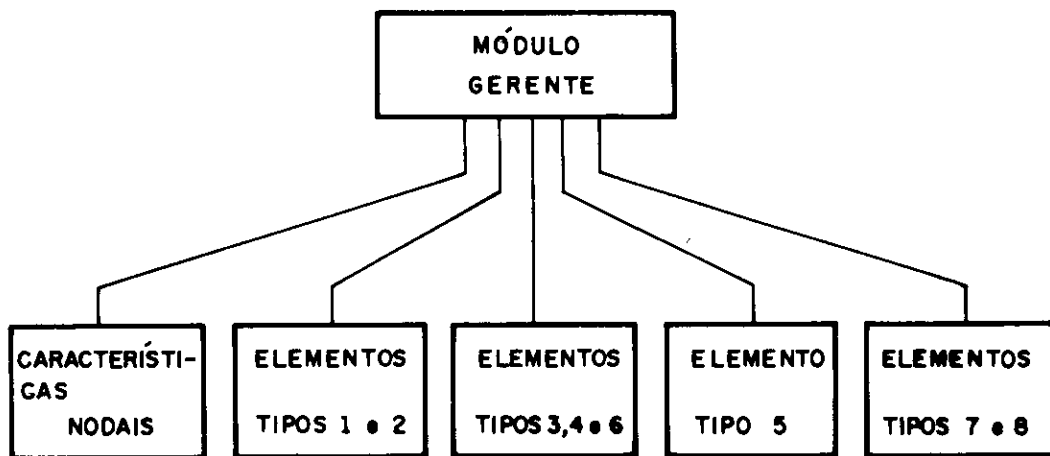


fig. 2.19 Organização do programa

por exemplo limpar um campo, salvar um campo, pedir uma mensagem explicativa, abandonar a edição, etc.

Os menus de opção são comandados através de caracteres comuns, letras ou números, sendo que os menus de controle são ativados por caracteres de controle, ou seja, caracteres comuns pressionados simultaneamente com a tecla CTRL ou teclas especiais. (neste trabalho a tecla CTRL será substituída pelo símbolo "^". Assim, a pressão simultânea das teclas CTRL e S será representada pelo símbolo "^S"). Isso ocorre pois as funções dos menus de controle precisam ser ativados enquanto se processa a entrada de dados. Portanto a ordem para execução dessas funções necessita de uma diferenciação em relação aos caracteres comuns que vão sendo digitados.

Para se conseguir esse efeito, os dados são adquiridos não da tela, como parece ao usuário, mas sim diretamente do teclado e só depois de analisados aparecem no vídeo. As rotinas que possibilitam esses recursos foram montadas com funções disponíveis no TURBO PASCAL e representam um grande avanço também na consistência dos dados, pois parte dela passa a ser feita simultaneamente com a digitação. Assim, por exemplo, se um determinado menu apresenta 3 opções que devem ser escolhidas através dos caracteres A, B ou C, apenas essas teclas poderão ser digitadas. Qualquer outra que seja pressionada provocará a emissão de um sinal sonoro de alerta e o cursor continuará solicitando a digitação de um caracter permitido, ou seja, A, B ou C.

#### **2.10.6. Apresentação dos Menus**

A partir deste ponto vai-se procurar apresentar alguns detalhes a respeito desses menus mencionados. Entretanto, devido ao espaço exíguo, tentar-se-á um resumo, que certamente deixará muitas informações em suspenso.

Ao ser carregado para a memória o programa gerente de edição, a tela é limpa e formatada segundo o esquema apresentado pela figura 2.20. Pode-se ver na referida figura algumas das características usadas para facilitar ao máximo a entrada dos dados. Nota-se, por exemplo, que o campo onde deve ser digitado o nome do arquivo de dados já está definido por traços horizontais. Dentro desse campo, qualquer caracter não permitido será recusado, soando um alarme para indicar o erro. O mesmo alarme soa quando se tenta ultrapassar o campo definido. Além disso, o programa coloca uma pequena explicação a respeito do dado a ser digitado, tornando muito improvável a ocorrência de erros, mesmo para o usuário inexperiente. Já na parte inferior do vídeo pode-se observar um dos chamados menus de controle, com funções para entrada do dado, volta de cursor com apagamento dos caracteres e limpeza do campo e abandono do programa.

Após a digitação do nome do arquivo a ser usado no armazenamento dos dados, pode ocorrer que o arquivo fornecido exista ou não no diretório corrente. Caso exista, o programa realiza a sua leitura carregando as informações contidas para serem modificadas. Se não existir, o programa procede normalmente, supondo que uma nova estrutura começa a ser definida. Em qualquer caso, logo após o nome, o usuário se defrontará com o menu apresentado na figura 2.21.

Trata-se, como já se mencionou, de um menu de opção. Na verdade o menu que permite o acesso a todas as funções de edição disponíveis. Duas dessas funções conduzem a rotinas de leitura imediata. Por exemplo, imaginando-se que dentro desse MENU PRINCIPAL tenha sido digitado o caracter "A", o usuário terá acesso à definição dos parâmetros de controle do processamento como número de pontos nodais, número de grupos de elementos, número de casos de carregamento e opção de minimização de banda da matriz. Caso o caracter digitado tenha sido "K", então o programa

LINGUAGEM ANALISE DE SISTEMAS ESTRUTURAIS RETICULADOS  
<< MODULO EDITOR >>

Nome do Arquivo de Dados : \_\_\_\_\_

O nome deve ser do tipo P P P P P P P P ( prefixo com ate 8 caracteres ). E permitida qualquer estrutura de subdiretorios, desde que o total de caracteres nao ultrapasse o campo definido.

<ENTER>: Entra Campo                    <BKSPC>: Retrocede Cursor  
<TAB> : Limpa Campo                    <ESC> : Volta ao Sistema

fig. 2.20 Tela Inicial

<< MENU PRINCIPAL >>

A - PARAMETROS DE CONTROLE  
B - CARACTERISTICAS NODAIS  
C - ELEMENTO TIPO 1 - TRELICA  
D - ELEMENTO TIPO 2 - BARRA  
E - ELEMENTO TIPO 3 - CONTORNO  
F - ELEMENTO TIPO 4 - MEMBRANA  
G - ELEMENTO TIPO 5 - PLANO  
H - ELEMENTO TIPO 6 - SOLIDO  
I - ELEMENTO TIPO 7 - PLACA/CASCA  
J - ELEMENTO TIPO 8 - SAPATA  
K - MULTIPLICADORES DE ACOES

O - ORIENTACAO AO USUARIO  
S - SALVA DADOS  
X - SISTEMA OPERACIONAL

SUA OPCAO : \*

fig. 2.21 Menu Principal

realizará a leitura de multiplicadores para as ações do casos de carregamento definidos. Esses multiplicadores são utilizados, por exemplo, para definir combinações de carregamento.

Entretanto, todos os outros caracteres, de B a J, conduzirão a novos menus. Uma observação desses menus permitirá uma verdadeira visão panorâmica a respeito da diversidade de funções embutidas no programa. Segue uma relação dos menus e do número das figuras que os apresenta, sempre em relação ao caracter digitado :

B - Menu de Características Nodais	- figura 2.22
C - Menu Elemento Tipo 1 - Treliça	- figura 2.23
D - Menu Elemento Tipo 2 - Barra	- figura 2.24
E - Menu Elemento Tipo 3 - Contorno	- figura 2.25
F - Menu Elemento Tipo 4 - Membrana	- figura 2.26
G - Menu Elemento Tipo 5 - Plano	- figura 2.27
H - Menu Elemento Tipo 6 - Tridimensional	- figura 2.28
I - Menu Elemento Tipo 7 - Placa/Casca	- figura 2.29
J - Menu Elemento Tipo 8 - Sapata	- figura 2.30

As opções oferecidas podem ser usadas em qualquer ordem com alguns cuidados. Por exemplo, não se pode definir as restrições de um nó sem que anteriormente esse nó tenha sido definido. Da mesma maneira, se o usuário tentar definir um gradiente de temperatura para um elemento ainda não definido, o programa acusará um erro.

## **2.10.7. Rotinas para Entrada dos Dados**

### **2.10.7.1. Considerações Iniciais**

Em todas as rotinas para entrada dos dados a



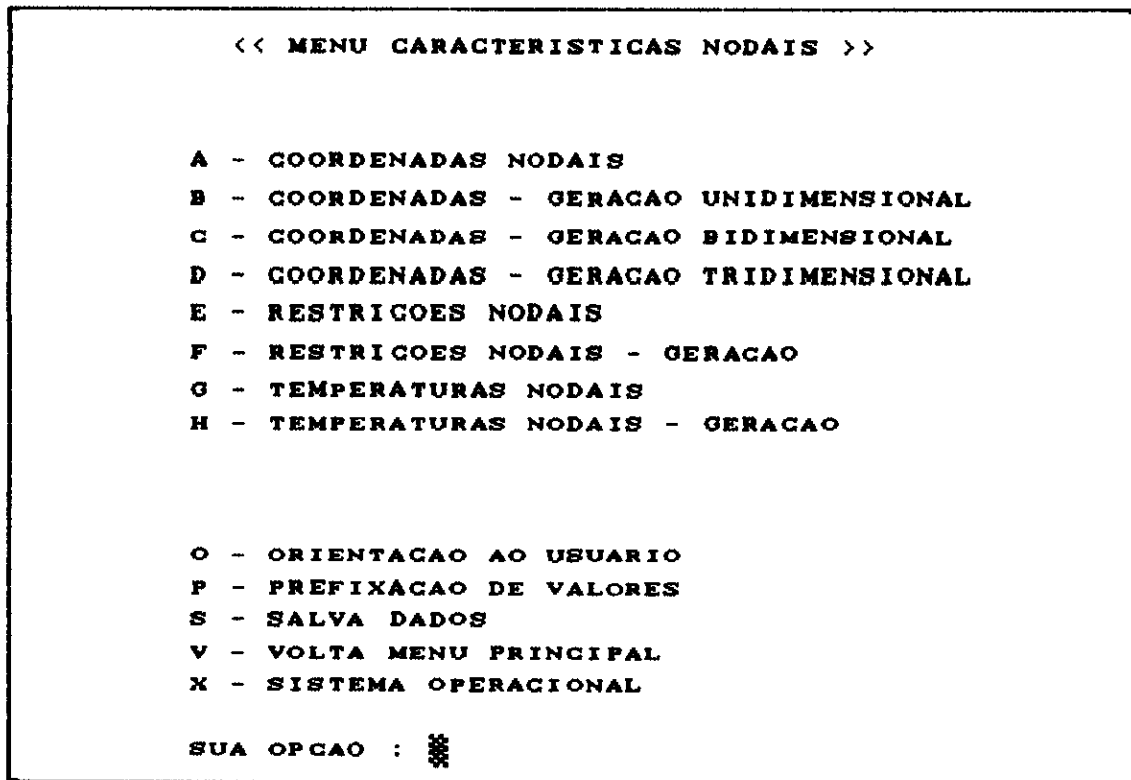


fig. 2.22 Menu de Características Nodais

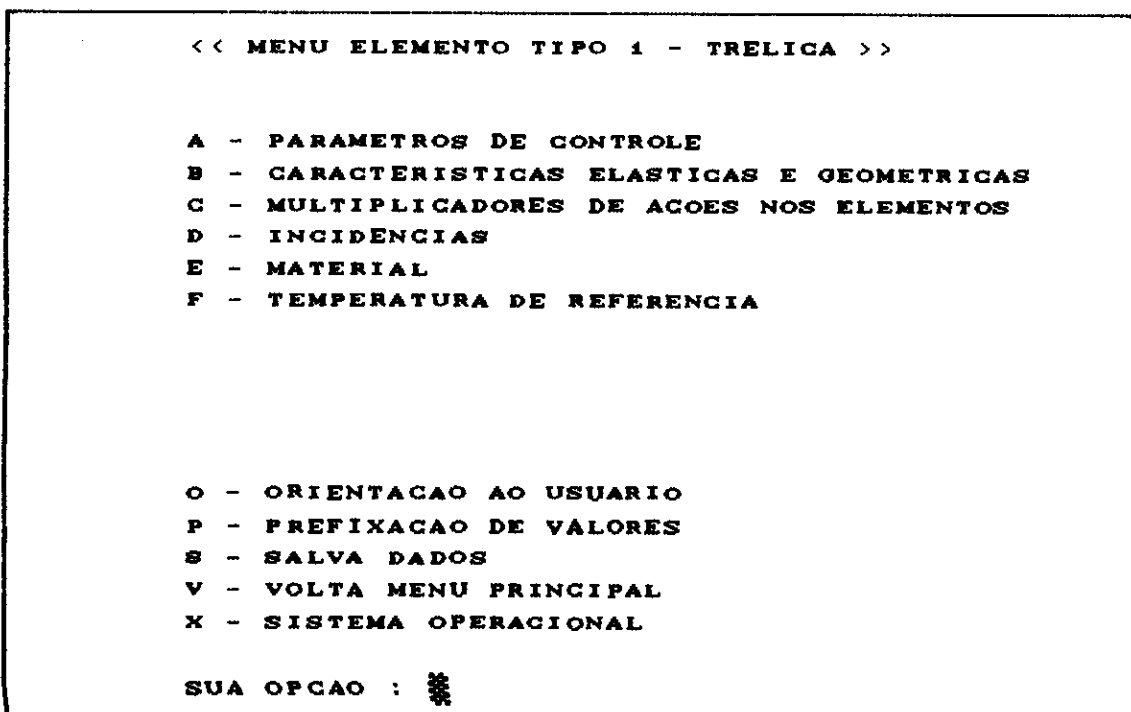


fig. 2.23 Menu do Elemento Tipo 1

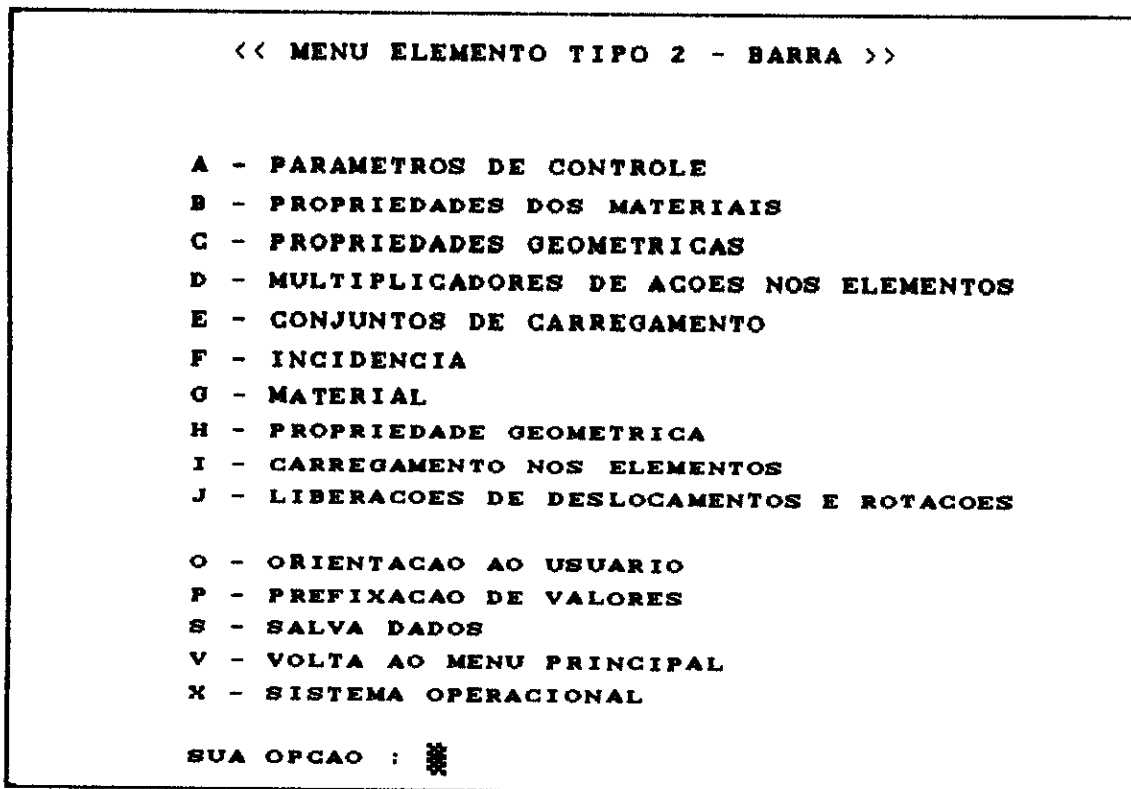


fig. 2.24 Menu do Elemento Tipo 2

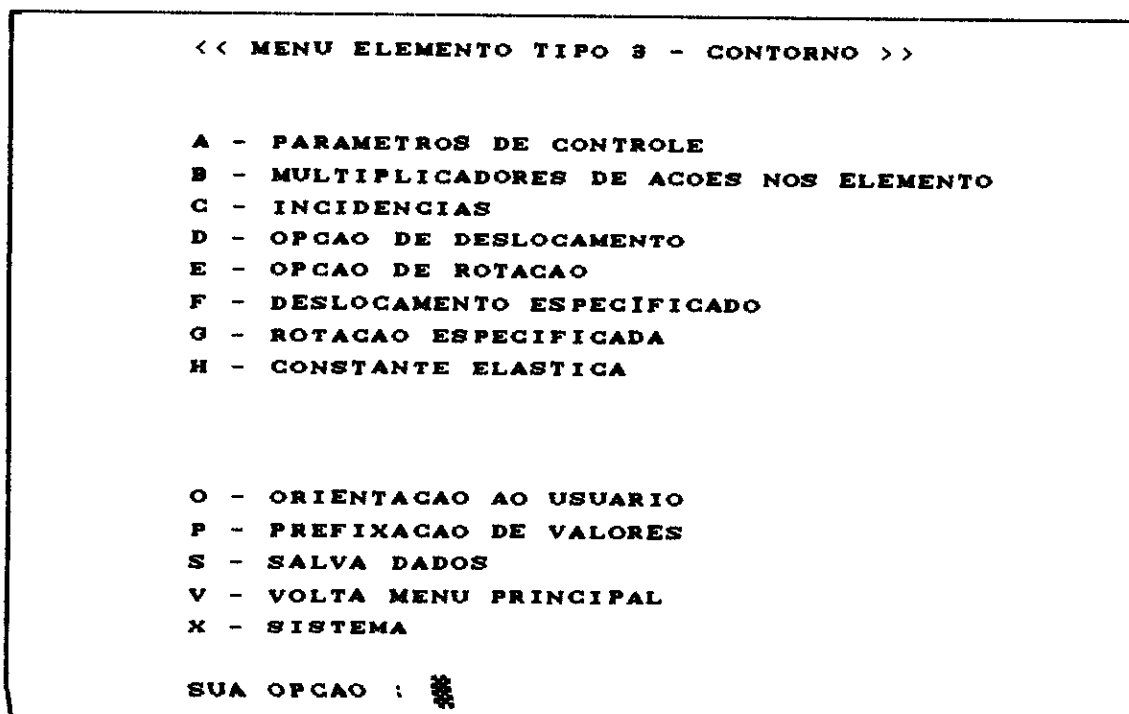


fig. 2.25 Menu do Elemento Tipo 3

```

<< MENU ELEMENTO TIPO 4 - MEMBRANA >>

A - PARAMETROS DE CONTROLE
B - PROPRIEDADES DOS MATERIAIS
C - MULTIPLICADORES DE ACOES NOS ELEMENTOS
D - INCIDENCIAS
E - MATERIAL
F - PRESSAO LATERAL
G - OPCAO DE APRESENTACAO DE RESULTADOS
H - ESPESSURA

O - ORIENTACAO AO USUARIO
P - PREFIXACAO DE VALORES
S - SALVA DADOS
V - VOLTA MENU PRINCIPAL
X - SISTEMA

SUA OPCAO : █

```

fig. 2.26 Menu do Elemento Tipo 4

```

<< MENU ELEMENTO TIPO 5 - PLANO >>

A - PARAMETROS DE CONTROLE
B - PROPRIEDADES DOS MATERIAIS
C - MULTIPLICADORES DE ACOES NOS ELEMENTOS
D - INCIDENCIAS
E - MATERIAL
F - TEMPERATURA DE REFERENCIA
G - PRESSAO LATERAL
H - OPCAO DE APRESENTACAO DE RESULTADOS
I - ESPESSURA

O - ORIENTACAO AO USUARIO
P - PREFIXACAO DE VALORES
S - SALVA DADOS
V - VOLTA MENU PRINCIPAL
X - SISTEMA

SUA OPCAO : █

```

fig. 2.27 Menu do Elemento Tipo 5

```

<< MENU ELEMENTO TIPO 6 - SOLIDO >>

A - PARAMETROS DE CONTROLE
B - PROPRIEDADES DOS MATERIAIS
C - CARGAS DISTRIBUIDAS DE SUPERFICIE
D - MULTIPLICADORES DE ACOES NOS ELEMENTOS
E - INCIDENCIAS
F - NUMERO DE PONTOS DE INTEGRACAO
G - MATERIAL
H - CARREGAMENTO NOS ELEMENTOS
I - OPCAO DE APRESENTACAO DE RESULTADOS
J - TEMPERATURA DE REFERENCIA

O - ORIENTACAO AO USUARIO
P - PREFIXACAO DE VALORES
S - SALVA DADOS
V - VOLTA MENU PRINCIPAL
X - SISTEMA

SUA OPCAO : █

```

fig. 2.28 Menu do Elemento Tipo 6

```

<< MENU ELEMENTO TIPO 7 - PLACA/CASCA >>

A - PARAMETROS DE CONTROLE
B - PROPRIEDADES DOS MATERIAIS
C - MULTIPLICADORES DE ACOES NOS ELEMENTOS
D - INCIDENCIAS
E - MATERIAL
F - ESPESSURA
G - PRESSAO LATERAL
H - TEMPERATURA DE REFERENCIA
I - GRADIENTE DE TEMPERATURA

O - ORIENTACAO AO USUARIO
P - PREFIXACAO DE VALORES
S - SALVA DADOS
V - VOLTA MENU PRINCIPAL
X - SISTEMA

SUA OPCAO : █

```

fig. 2.29 Menu do Elemento Tipo 7

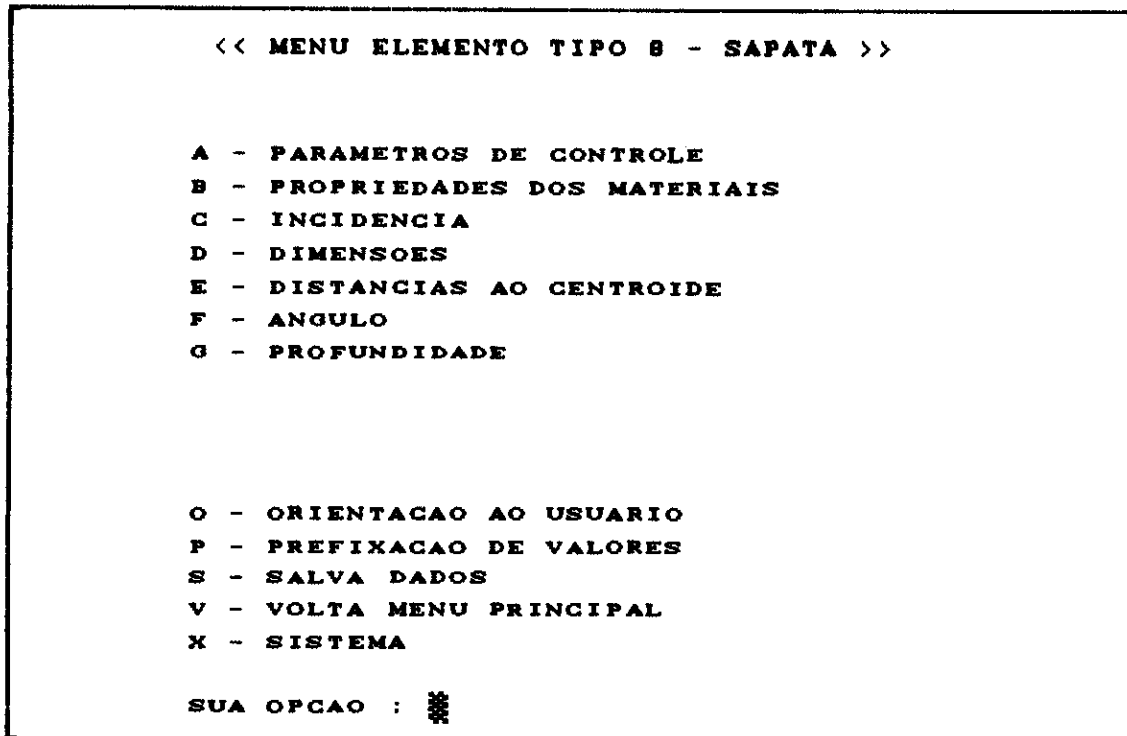


fig 2.30 Menu do Elemento Tipo 8

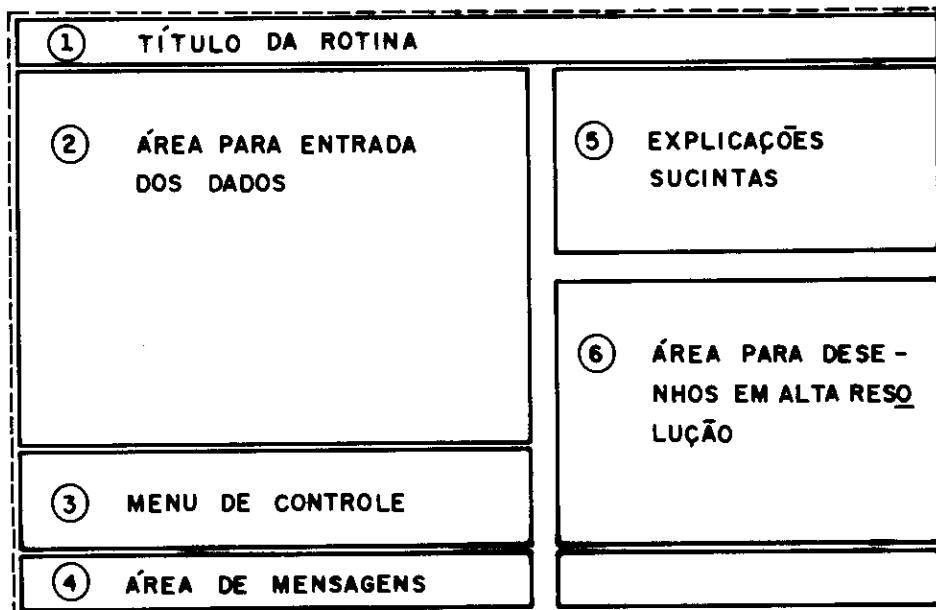


fig. 2.31 Divisão da tela em áreas distintas

formatação do vídeo será sempre a mesma, segundo o esquema de seis sub-regiões apresentadas pela figura 2.31. Todas as linhas limites são traçadas em alta resolução formando regiões com tamanhos e destinação pré-definidas, de modo que o usuário possa acostumar-se rapidamente ao "layout" utilizado. A seguir são apresentadas algumas explicações sobre o tamanho e a função de cada uma dessas áreas:

#### 2.10.7.2. Área 1

Região destinada a saídas em baixa resolução com tamanho de 1 linha por 80 colunas. Sua função é apresentar o título da rotina que está em uso. Os nomes que aqui aparecem são as opções contidas nos diversos menus apresentados nas figuras de 2.22 a 2.30. Por exemplo, COORDENADAS NODAIS - GERACAO TRIDIMENSIONAL, ELEMENTO 1 - INCIDENCIAS , ELEMENTO 3 - TEMPERATURA DE REFERENCIA, etc.

#### 2.10.7.3. Área 2

Local destinado à entrada dos dados propriamente ditos. Seu tamanho total é de 16 linhas por 36 colunas. Nessa região o programa escreve as variáveis que devem ser fornecidas e dá toda a ajuda ao usuário para que esses dados sejam corretamente digitados.

#### 2.10.7.4. Área 3

Área destinada à colocação do menu de controle. Seu tamanho total é de 3 linhas por 36 colunas. Esse menu também será sempre o mesmo em todas as rotinas, oferecendo as seguintes opções:

<ENTER> : Tecla que comanda a entrada do dado digitado. Até que ela seja pressionada o usuário poderá digitar, apagar, voltar a digitar qualquer caracter permitido dentro do campo fixado para a entrada.

<BASPC> : (Back Space) Comanda o retrocesso do cursor, realizando o apagamento dos caracteres já digitados e restaurando os traços horizontais inferiores que demarcam o campo de entrada. Quando pressionado com o cursor na primeira posição do campo dá acesso ao campo imediatamente anterior.

<TAB> : Comando usado para limpar totalmente os caracteres digitados sobre um determinado campo, restaurando os traços horizontais e colocando o cursor na primeira posição.

<ESC> : Produz o encerramento da execução da rotina, permitindo ao usuário acessar o menu imediatamente anterior. Ressalta-se que ao se pressionar esta tecla os dados até então fornecidos continuam em memória central, prontos para novas alterações.

<^S> : Comando usado para "salvar", ou seja guardar em arquivo, os dados já fornecidos. Esse comando faz com que o programa archive todos os dados já digitados, e não somente os fornecidos pela rotina em uso.

<^J> : Provoca a apresentação de uma página de informações sobre os dados a serem fornecidos na rotina em uso. Essas explicações vem praticamente eliminar a necessidade de consulta aos manuais tornando simples e rápido o período de adaptação do usuário ao programa. As informações aparecem e desaparecem instantaneamente pois estão contidas em memória central e são transferidas diretamente para o "buffer" do vídeo.

#### 2.10.7.5. Área 4

Local onde são escritas todas as mensagens do programa. Seu tamanho total é de 1 linha por 80 colunas. Existem dois níveis de consistência de dados : a consistência feita simultaneamente com a digitação dos dados e a consistência do dado fornecido frente às informações já recebidas, que pode resultar numa mensagem de atenção ou de erro. Em qualquer caso o programa emite um alarme sonoro e apresenta uma mensagem que indica o problema detectado.

Assim sendo, desde a digitação de um número real em um campo destinado a números inteiros como a tentativa de se fornecer cargas para um nó não definido serão recusadas, com o aviso correspondente apresentado nesta área. De modo semelhante, quando se redefine as coordenadas de um determinado nó, o programa aceita a redefinição, mas apresenta uma mensagem chamando a atenção do usuário para o fato.

#### 2.10.7.6. Área 5

Campo destinado a um resumo explicativo dos dados a serem adquiridos pela rotina em uso. Seu espaço total é de 7 ou 20 linhas por 38 colunas. Como já foi anteriormente mencionado, o usuário pode pedir a apresentação de uma página com explicações relativamente detalhadas sobre os itens a serem fornecidos para a rotina. Isso entretanto pode nem ser necessário já que se reservou este local para explicações sucintas, que na grande maioria dos casos é suficiente para elucidar as dúvidas.

Em algumas rotinas, a área reservada para esse resumo compreende apenas 7 linhas, pois o campo definido como área



6, onde são apresentados desenhos em alta resolução, está ativado. Nesse caso se enquadram todas as rotinas que definem coordenadas nodais e incidência de elementos, onde os nós ou elementos vão sendo desenhados simultaneamente com a edição. Nas outras rotinas a área 6 não precisa ser utilizada. Portanto, o espaço reservado às explicações aumenta, passando a ocupar 20 linhas e mantendo as 38 colunas.

#### 2.10.7.7. Área 6

É um local destinado a desenhos em alta resolução. Considerando-se que a placa de vídeo que equipa a maioria dos computadores compatíveis PC, é a CGA ("Color Graphics Adapter"), optou-se por utilizar a resolução de 640 pontos na horizontal por 200 pontos na vertical, para a apresentação de esboços da estrutura editada. Como foi reservada 1/4 da área total da tela para esse uso, conclui-se que a área 6 tem 320 x 100 pontos.

Nesse espaço é que estarão representados graficamente os nós e elementos que vão sendo definidos. Esse desenho possibilita ao usuário acompanhar, simultaneamente com a entrada dos dados, a forma que a estrutura vai tomando com respeito as coordenadas nodais e incidência dos elementos. A escala evidentemente depende do tamanho da estrutura, mas normalmente é possível uma boa visualização de sua forma geral, evitando-se erros grosseiros que possam ser cometidos.

Quando se fornecem dados de coordenadas nodais e incidência de elementos para uma estrutura cujos dados já se encontram arquivados, o programa realiza uma pesquisa nas coordenadas nodais definidas posicionando o desenho dentro do campo da melhor maneira possível. Assim, à medida que os nós e elementos vão sendo redefinidos, são apagados os antigos e mostrados os novos.

Entretanto, quando se define pela primeira vez uma estrutura, é necessário fornecer ao programa os eixos segundo os quais se pretende o esboço. Além disso, são necessárias as coordenadas nodais máximas e mínimas nesses eixos. Portanto, ao ser digitado o nome de um arquivo não encontrado no diretório, o usuário será solicitado a fornecer esses dados necessários ao bom posicionamento do desenho da estrutura.

#### 2.10.7.8. Exemplo

Para finalizar esta apresentação das rotinas utilizadas para aquisição dos dados, vai-se apresentar um exemplo. Trata-se de uma cópia da tela durante a utilização da rotina para entradas de coordenadas nodais, mostrada na figura 2.32.

Lá pode-se observar os valores digitados para o número do nó, e para as coordenadas  $X_1$ ,  $X_2$  e  $X_3$ , sendo que para a coordenada  $X_3$  ainda não foi pressionada a tecla <ENTER>. Por essa razão o campo ainda está demarcado. Mostra também o título da rotina, o menu de controle, as explicações sucintas e um conjunto de nós desenhados na área de alta resolução definida para esse fim.

Logicamente seria impossível a apresentação de todas as rotinas disponíveis no programa. Entretanto, todas elas têm a mesma filosofia de utilização, variando-se apenas os parâmetros a serem lidos.

#### 2.10.8. Gravação<sup>N</sup> do Arquivo de Dados

Quando o usuário dá por encerrada a sua sessão de trabalho ou simplesmente aciona de algum ponto do programa o comando de salvar os dados digitados, o módulo editor

Coordenadas Nodais - Entrada No por No

<p> <b>NN = 15</b>  <b>X = 8.90</b>  <b>Y = 6.50</b>  <b>Z = 1.30</b> </p>	<p> <b>NN</b> : numero do no  <b>X</b> : coordenada segundo eixo X  <b>Y</b> : coordenada segundo eixo Y  <b>Z</b> : coordenada segundo eixo Z                 </p>
<p> <b>&lt;ENTER&gt;</b> : Entra  <b>&lt;TAB&gt;</b> : Limpa  <b>&lt;^S&gt;</b> : Salva                 </p>	<p> <b>&lt;BASPC&gt;</b> : Retrocede  <b>&lt;ESC&gt;</b> : Menu  <b>&lt;^J&gt;</b> : Ajuda                 </p>

Mensagem :

fig. 2.32 Tela para entrada de coordenadas nodais

realiza algumas tarefas. Em primeiro lugar, verifica se existe um arquivo já gravado com nome coincidente com o que está sendo digitado. Caso isso seja verdadeiro, o programa realiza uma cópia do arquivo alterando o nome com a adição do sufixo BAK.

A partir daí, começa a compilar os dados que se encontram em memória central, transferindo-os para o arquivo comandado. Obviamente, essa transferência se dá em ordem pré-determinada, exatamente a ordem em que os parâmetros serão lidos pelo programa LS. A sequência de gravação é a seguinte : parâmetros globais da estrutura, características nodais (coordenadas, temperatura e restrições), dados relativos aos grupos de elementos, carregamentos nodais e multiplicadores para os casos de carregamento.

Após estar completo o trabalho de gravação, o programa fica aguardando o próximo comando do usuário. Se os dados continuarem a ser fornecidos, o processamento continua normalmente com todos os valores anteriormente fornecidos ainda na memória central. Caso o próximo comando seja a volta para o sistema operacional, o programa imediatamente executa seu procedimento de saída, retirando-se da memória central do computador.

## 2.11. CAPACIDADE, DESEMPENHO E CONCLUSÕES

### 2.11.1. Introdução

Neste item, serão apresentados alguns detalhes a respeito de três tópicos que ajudam a caracterizar um programa de análise estrutural. São eles : a capacidade, em termos de número de pontos nodais e graus de liberdade, a acuidade dos resultados obtidos e, finalmente, o tempo de processamento gasto para as análises. Através dessas características, procurar-se-á uma comparação com outros programas de análise estrutural desenvolvidos para microcomputadores compatíveis PC.

Essa comparação não é simples de ser realizada. Quando se considera a capacidade de análise, isso se verifica pois os programas existentes não deixam claro seus recursos a esse respeito, sendo necessário um grande esforço na tentativa de pelo menos se aproximar desses limites. Por outro lado, para a comparação da acuidade de resultados e do tempo de processamento, existe a dificuldade de se executar análises, com uma mesma estrutura e num mesmo computador, utilizando-se os programas entre os quais se deseja a comparação. Mesmo assim, tentar-se-á seriamente estabelecer essas comparações e a partir daí desenvolver algumas conclusões a respeito dos resultados obtidos na presente pesquisa.

Devido às dificuldades mencionadas no parágrafo anterior, a referida comparação será, sempre que possível, realizada entre o módulo LS e apenas mais três programas de análise : o SUPERSAP, da Algor Interactive Systems, Inc., e SAP80/SAP90, desenvolvidos pela Computers & Structures, Inc.. A execução dos processamentos foi realizada em computadores compatíveis com IBM AT , com coprocessador matemático 80287 e frequência de 12 MHertz.

Faz-se necessário agradecer às seguintes empresas pela gentileza da cessão de seus programas para os testes executados : França e Ungaretti Consultoria e Projetos, de São Paulo, e Estádio 3 Projetos Estruturais, de Porto Alegre.

### 2.11.2. Capacidade de Análise

A capacidade do programa LS para analisar grandes estruturas está ligada basicamente a duas expressões. Pode-se apresentá-las da seguinte maneira :

$$nn \leq 10.000 \quad \dots(2.10)$$

$$6 nn + 2 neqb nb \leq 100.000 \quad \dots(2.11)$$

onde,

nn : número de pontos nodais da estrutura.

neqb : número de equações por bloco do sistema global.

nb : número de blocos.

A expressão 2.10 é correspondente à entrada dos pontos nodais da estrutura. Nesse momento estarão em memória central, na já mencionada área de trabalho de 400 Kbytes, a matriz de restrições IR, de dimensão "nn x 6", e os três vetores de coordenadas mais um vetor de temperatura nodais, todos de dimensão "nn". Portanto o máximo número de nós da estrutura a ser analisada precisa ser 10.000.

Já a expressão 2.11 diz respeito à entrada das cargas nodais. Nesse momento, também na referida área de trabalho, estarão armazenados a mesma matriz IR e, na hipótese mais otimista, um vetor F de "neqb x nb" elementos reais de 8 bytes. Diz-se "na hipótese mais otimista" pois supõe-se que se esteja considerando apenas um caso de carregamento. Se o

número de casos de carregamento, "nc", for maior que 1, na verdade o vetor F deveria ter "neqb x nb x nc" elementos. Entretanto, essa não é uma limitação que impeça a análise pois o programa verifica o número de casos de carregamento que podem ser considerados através da expressão :

$$nc \leq \frac{100.000 - 6 nn}{2 neqb nb} \quad \dots(2.12)$$

Caso o número de carregamentos fornecido seja maior que o valor calculado na expressão 2.12, o programa é executado apenas com os "nc" carregamentos que a memória comporta, sendo emitida uma mensagem correspondente.

Quanto às fases de montagem e solução do sistema de equações global, que à primeira vista deveriam causar as maiores limitações, praticamente não influem no limite da capacidade do programa. Isso porque pode-se chegar a ter apenas duas equações em cada bloco do sistema. Para se ter uma idéia do que isso significa basta fazer o número de equações por bloco igual a dois nas expressões 2.1, 2.2 e 2.3, no item 2.7. Verifica-se dessa maneira que, muito antes de serem atingidos aqueles limites, as expressões 2.10 e 2.11 já indicarão a impossibilidade da análise.

Convém ressaltar que esses limites poderiam ser modificados de modo a permitir a análise de estruturas ainda maiores. Por exemplo, na leitura dos pontos nodais, não haveria uma necessidade absoluta da matriz e dos vetores mencionados estarem simultaneamente na área de trabalho, acontecendo o mesmo na entrada das cargas nodais. Entretanto, caso isso não ocorresse, sem dúvida o acesso aos arquivos em disco, e por consequência o tempo de processamento, iriam crescer de modo significativo. Deve-se notar que esse acréscimo seria verificado para qualquer estrutura, grande ou pequena. Lembrando-se que os limites apresentados nas expressões 2.10 e 2.11 permitem, com folga, a análise da imensa maioria das estruturas que se

possa imaginar, não seria lógico tomar esse tipo de providência.

Para maior clareza a respeito das limitações verificadas, pode-se mencionar que as seguintes estruturas poderiam ser analisadas pelo programa LS : treliças planas e chapas com 10.000 nós; treliças espaciais, pórticos planos, placas e sólidos com 8.300 nós; cascas e pórticos espaciais com 5.550 nós.

Quanto à capacidade de análise dos outros programas estruturais mencionados, apenas o SAP80 traz uma indicação de que seu módulo básico, que permite cálculo de estruturas de barras, pode analisar até 10.000 graus de liberdade. Isso corresponderia, no caso de estruturas com seis graus de liberdade por nó, a um limite de 1.667 nós. Já o SAP90 não traz indicação de suas limitações e infelizmente não foi possível utilizar o programa de modo a estabelecer esses valores. Quanto ao SUPERSAP, que também não fornece explicitamente suas limitações, foi possível através de informações a respeito do andamento do processamento verificar que seu limite deve ser da ordem de 4.400 nós, o que leva a um máximo teórico de 26.400 graus de liberdade.

Num resumo, esses valores estão organizados na tabela 2.9. Lá verifica-se que o programa LS, apesar de permitir 10.000 nós, não possibilita os 60.000 graus de liberdade teoricamente possíveis.

Limites	SAP80	SAP90	SUPERSAP	LS
Nós	* 1.667	-	4.400	10.000
G. Lib.	10.000	-	** 26.400	33.330

tabela 2.9

obs:

\* : calculado a partir de "G. Lib.". Pode ser maior.

\*\* : calculado a partir de "Nós". Pode ser menor.



### 2.11.3. Acuidade Numérica

Este tópico diz respeito à precisão numérica envolvida na solução do sistema de equações globais. Não se analisa aqui a aproximação conseguida com este ou aquele tipo de elemento em particular. Essa aproximação será estudada caso a caso no próximo capítulo.

O programa LS parte de um princípio diferente da maioria dos programas de análise estrutural. Essa maioria utiliza reais de 8 bytes em todas as etapas da análise. No caso do programa LS isso não acontece. Apenas as variáveis para as quais operações de multiplicação ou divisão mal condicionadas possam levar à ocorrência de erros são definidas como reais de 8 bytes. Todas as outras, onde essa circunstância não está presente, são definidas como reais de 4 bytes. Essa providência leva a duas vantagens significativas : aumento na capacidade do programa e redução do tempo de processamento verificado numa análise.

Entretanto, é necessário um bom controle a respeito desse item, pois a ocorrência de erros numéricos é um fato muito grave. Ela leva a resultados totalmente mascarados e inaceitáveis, em especial por tratar-se de um programa cuja utilização indevida pode ocasionar graves acidentes.

O procedimento clássico para realizar o controle mencionado é verificar o comportamento do programa para a solução de estruturas muito flexíveis. Nesses casos a matriz de rigidez global começa a apresentar valores muito discrepantes em sua diagonal principal, aumentando muito a possibilidade de ocorrência de erros numéricos. Um caso muito adequado a essa verificação é tomar uma viga, engastada em uma extremidade e livre na outra, e discretizá-la com muitos elementos barra. Nesse caso, apenas um elemento barra forneceria o resultado correto, desde que o carregamento aplicado fosse apenas nodal. Entretanto, dividindo-se o vão em muitos elementos, a

situação será muito propícia à verificação da acuidade numérica na solução do sistema.

Para programas em que os erros numéricos são passíveis de ocorrer, a divisão do vão em 50 elementos é suficiente para a obtenção de resultados completamente absurdos. Na verdade, a partir de 20 elementos a tendência à divergência já é bastante clara. Para se testar o comportamento do programa LS, definiu-se a utilização de uma viga engastada/livre com todas as características unitárias. Assim sendo, vão, módulo de rigidez à flexão e a uma carga na extremidade livre foram feitos igual a um. Nessas condições foram montadas cinco discretizações : 1, 100, 250, 500 e 1.000 elementos. Os resultados obtidos são apresentados pela tabela 2.10.

Elem	Max/Min	$\delta_{el}$	$\phi_{el}$	$M_{ef}$	$V_{ef}$
1	.12E+02	0.33333333	-0.50000000	-1.00000000	-1.00000000
100	.24E+08	0.3334746	-0.5002191	-1.00000000	-1.00000000
250	.38E+09	0.3267792	-0.4899136	-0.983635	-1.00000000
500	.30E+10	0.3278086	-0.4934319	-0.977845	-1.00000000
1000	.24E+11	0.3149348	-0.4718692	-0.946398	-1.00000000

tabela 2.10

onde,

- Elem : número de elementos da discretização.
- Max/Min : relação entre o máx. e o mín. elemento da diag.
- $\phi_{el}$  : rotação na extremidade livre.
- $\delta_{el}$  : deslocamento na extremidade livre.
- $M_{ef}$  : momento fletor na extremidade fixada.
- $V_{ef}$  : cortante na extremidade fixada.

A observação dos resultados apresentados permite considerar muito bom o desempenho do programa em relação à acuidade numérica na solução do sistema de equações global.

É claro que apenas esse exemplo não garante o bom funcionamento do programa. Entretanto ele é representativo das situações encontradas em todos os outros casos pesquisados. Lajes cogumelo e cascas de grande flexibilidade, pórticos tridimensionais de altura relativamente elevada e algumas outras situações desfavoráveis foram testadas e o comportamento em todos os casos revelou-se excelente.

Apenas como informação adicional, todos os outros programas testados apresentaram resultados satisfatórios quanto à ocorrência de erros numéricos. Merece destaque o programa SUPERSAP, onde a precisão numérica conseguida revelou-se excepcional, constituindo-se em um dos seus pontos fortes.

#### **2.11.4. Tempo de Processamento**

Quanto ao tempo de processamento gasto para a realização de análises, procurar-se-á discutir o assunto em dois tópicos distintos. O primeiro diz respeito aos tempos de processamento total e por etapas observados exclusivamente com a utilização do programa LS. Para tanto serão definidas uma série de estruturas representativas e os tempos verificados nessas análises serão mostrados e discutidos. Numa segunda etapa, procurar-se-á uma comparação entre os tempos totais obtidos com a utilização do programa LS e os demais programas já mencionados anteriormente, através da solução de uma mesma estrutura em um mesmo computador.

Ressalta-se que a comparação com os programas SAP80 e SAP90 torna-se mais complicada do que com o programa SUPERSAP. São dois os principais motivos para essa dificuldade. Em primeiro lugar os programas SAP não possuem os recursos de geração dos programas LS e SUPERSAP,

dificultando a modelagem das estruturas. Depois, durante a análise, os programas SAP emitem relatórios extensos na tela do computador, aumentando muito o tempo de processamento. Para um usuário experiente em computação é possível atuar a nível de sistema operacional, e através de um redirecionamento das saídas, conseguir que as mensagens não sejam apresentadas. Isso torna a comparação desejada mais viável em termos teóricos. Entretanto, talvez não seja a correta ao se pensar em um usuário padrão.

Para a apresentação dos tempos de processamento total e por etapas do programa LS foram as seguintes as estruturas escolhidas :

Estrutura 1 : pórtico tridimensional, 422 nós, 640 elementos barra.

Estrutura 2 : viga parede, 216 nós, 196 elementos membrana.

Estrutura 3 : bloco tridimensional, 499 nós, 320 elementos sólido.

Estrutura 4 : pavimento de edifício, 515 nós, 370 elementos placa e 231 elementos barra.

Estrutura 5 : pavimento de edifício, 1250 nós, 1130 elementos placa, 358 elementos barra e 5 elementos treliça.

Estrutura 6 : parabolóide hiperbólico, 295 nós, 256 elementos casca e 12 elementos contorno.

Desse modo espera-se a obtenção de valores que sirvam de referência para a estimativa de tempo de processamento de estruturas de tipos variados. Os resultados obtidos encontram-se organizados na tabela 2.11, onde os tempos estão anotados em horas, minutos e segundos, para cada estrutura mencionada e cada etapa de processamento.

Etp\Est	Estru1	Estru2	Estru3	Estru4	Estru5	Estru6
Etp 1	0:00:12	0:00:07	0:00:14	0:00:16	0:00:35	0:00:07
Etp 2	0:02:25	0:01:43	0:09:20	0:10:08	0:29:38	0:07:17
Etp 3	-	0:00:14	0:01:16	0:00:45	0:04:10	-
Etp 4	0:01:44	0:00:40	0:03:04	0:01:59	0:10:50	0:01:34
Etp 5	0:06:49	0:02:12	0:29:27	0:07:26	1:28:07	0:23:19
Etp 6	0:01:22	0:00:28	0:03:16	0:00:42	0:01:31	0:00:15
Total	0:12:45	0:05:34	0:47:04	0:20:08	2:15:41	0:32:45

tabela 2.11

Inicialmente, pela observação dos tempos totais da tabela 2.11, percebe-se claramente que é muito viável a utilização de microcomputadores para a análise de estruturas. Considerando-se que os exemplos escolhidos são do porte dos que compõem o dia a dia de um bom escritório de projetos, percebe-se que a utilização de um modelo matemático relativamente sofisticado não implica na obtenção de tempos computacionais elevados.

Quanto aos tempos devidos às etapas de processamento, percebe-se que a maior parcela do esforço computacional concentra-se na solução do sistema de equações, etapa 5. Em média metade do tempo total, e em alguns casos até mais, é destinada à essa solução. Esse é o motivo de ser tão importante um cuidado extremo na definição do algoritmo de solução. Logo após, em ordem decrescente de importância, situa-se a montagem da matriz de rigidez dos elementos. Corresponde a algo em torno de 15 a 20% do tempo total. Em seguida a montagem do sistema global, de 5 a 10%, e depois a minimização da largura da banda e cálculo e impressão das tensões e ou esforços nos elementos, menos de 5% do total. A etapa que corresponde à entrada das características

nodais praticamente nada significa perante o total gasto.

Já para a comparação de tempos de processamento do programa LS em relação aos tempos de processamento obtidos com outros programas de análise estrutural, irá-se adicionar mais duas pequenas estruturas ao conjunto das seis médias anteriormente definidas :

Estrutura 7 : pórtico plano, 8 nós e 9 elementos barra.

Estrutura 8 : laje, 9 nós e 4 elementos placa.

Desse modo, os tempos de processamento de cada estrutura, para os programas analisados, estarão apresentados na tabela 2.12. Ressalta-se que para os programas SAP80 e SAP90, esses tempos foram medidos eliminando-se as saídas de tela através de redirecionamento.

Estrut\Prog	SAP80	SAP90	SUPERSAP	LS
Estrut 1	0:26:46	0:16:02	0:31:25	0:12:45
Estrut 2	0:11:50	0:09:34	0:12:14	0:05:34
Estrut 3	-	-	1:23:49	0:47:04
Estrut 4	-	-	0:40:09	0:20:08
Estrut 5	-	-	3:05:03	2:15:41
Estrut 6	-	-	0:48:15	0:32:45
Estrut 7	0:00:46	0:01:26	0:00:58	0:00:08
Estrut 8	0:01:01	0:01:28	0:01:07	0:00:15

tabela 2.12

Pela comparação dos tempos de processamento entre os programas, pode-se notar que para pequenas estruturas o programa LS é muito mais rápido que os demais. Essa enorme diferença ocorre provavelmente porque o carregamento e

inicializações do LS são mais eficientes. Basta notar que todo o programa LS tem de código apenas 275 Kbytes. Já módulo elástico linear do SUPERSAP tem 584 Kbytes e ainda necessita para o processamento de mais um módulo de 135 Kbytes. No caso do SAP80, os módulos executados para a análise elástica linear somam 575 Kbytes. Para o SAP90 a situação é ainda pior, pois os módulos da parte elástica ocupam arquivos de mais de 1 Mbyte. Além disso, o SAP80 e SAP90 são compostos por módulos independentes, chamadas por uma macro-instrução, tendo portanto a necessidade de um volume maior de acessos a disco rígido para gravação e leitura de arquivos temporários.

Mesmo para as estruturas médias o programa LS leva vantagem em velocidade de análise. Aproxima-se dele o programa SAP90, ficando os demais com tempos muito superiores, em média de 2 a 3 vezes maior.

#### 2.11.5. Conclusões Gerais

Todos os objetivos colocados no item 1.4 do trabalho foram alcançados. O programa apresenta uma boa capacidade de análise, uma boa acuidade de resultados, tem 8 tipos de elementos a serem utilizados nas modelagens e ainda apresenta tempos de processamentos relativamente pequenos.

Obviamente que um programa tão complexo precisa ser constantemente atualizado. Desse modo, os bons parâmetros obtidos podem estar completamente obsoletos a médio prazo se um grande esforço de novos desenvolvimentos não for implementado. Desse modo, como fecho para este capítulo, podem ser destacados alguns itens que devem receber maior prioridade para esses desenvolvimentos pretendidos.

O primeiro tópico a ser repensado é a montagem e solução do sistema de equações globais. Pensa-se poder reduzir os tempos de processamento atualmente obtidos com

mudanças nesse algoritmo. Outro ponto que deve merecer atenção é a biblioteca de elementos. Pesquisas já desenvolvidas estão ampliando a possibilidade dos elementos. Pode-se destacar a consideração de diafragmas rígidos e efeito de não-linearidade geométrica para os elementos treliça e barra. O desenvolvimento de um novo tipo de elemento placa, inclusive com critérios de plastificação do material. E ainda trabalha-se na otimização do processamento de estruturas com elementos de geometria repetida.

Desse modo, com o concurso de alguns pesquisadores ligados aos métodos numéricos, espera-se corrigir eventuais deficiências existentes no desenvolvimento até aqui realizado.



## 2.12 REFERÊNCIAS BIBLIOGRÁFICAS

[2.1] "FORTRAN 4.01 - User's Guide"; Microsoft Corporation; 1987.

[2.2] "FORTRAN 4.01 - Reference Manual"; Microsoft Corporation; 1987.

[2.3] Alway, G.G.; Martin, D.W.; "An Algorithm for Reducing the Bandwidth of a Matrix of Simetric Configuration"; Computer Journal; Vol 8; 1965.

[2.4] Akyus, F.A.; Utku, S.; "An Automatic Relabeling Scheme for Bandwidth Minimization of Stiffness Matrices"; Journal of the American Institute of Aeronautics and Astronautics; Vol 6; 1968.

[2.5] Rosen, R.; "Matrix Bandwidth Minimization"; Proceedings of the 23rd National Conference Association for Computing Machinery; Brandon Systems Press; Princeton, NJ; 1968.

[2.6] Grooms, H.R.; "Algorithm for Matrix Bandwidth Reduction"; ASCE, J. Structural Division; Vol 98; 1972.

[2.7] Cuthill, E.; McKee, J.; "Reducing the Bandwidth of Sparse Symmetric Matrices"; Proc. 24th National Conference of the Association for Computing Machinery, ; New York, 1969.

[2.8] King, I.P.; "An Automatic Reordering Scheme for Simultaneous Equations Derived from Network Systems"; International Journal for Numerical Methods in Engineering, Vol 2; 1970.

- [2.9] George, A.; "Computer Implementation of the Finite Element Method"; STAN-CS-71\_208, Computer Science Dept.; Stanford University; Stanford; 1971.
- [2.10] Collins, R.J.; "Bandwidth Reduction by Automatic Renumbering"; International Journal for Numerical Methods in Engineering; Vol 6; 1973
- [2.11] Roberts, E.; "Relabeling of Finite Element Meshes Using a Random Process"; TM X-2660, NASA ; Lewis Research Center; Cleveland; 1972
- [2.12] Gibbs, N.; Poole, W.G.; Stockmeyer, P.K.; "An Algorithm for Reducing the Bandwidth and Profile of a Sparse Matrix"; SIAM, J. Numer. Anal.; Vol 13; No. 2; 1976.
- [2.13] Irons, B.M.; "A Frontal Solution Program for Finite Element Analysis"; International Journal for Numerical Methods in Engineering; Vol 2; 1970.
- [2.14] Soriano, H.L.; "Sistemas de Equações Algébricas Lineares em Problemas Estruturais"; Seminário 280, LNEC; Lisboa, 1981.
- [2.15] Ida, N.; Lord, W.; "Solution of Linear Equations for Small Computer Systems"; International Journal for Numerical Methods in Engineering; Vol 20; 1984.
- [2.16] Zisserman, A.; "Program F.Matinv for the Solution of Sparse Linear System  $Ax=b$ "; Advanced Engineering Software; Vol 6; Nº 1; 1984.
- [2.17] Hazony, Y.; "A Linear Solver for Sparse Banded Matrices"; Advanced Engineering Software; Vol 6; Nº 1; 1984.

[2.18] Cheu, T.C.; Johnson, C.P.; Craig, R.R.; "A Solution Method of Equilibrium Equations for Large Structural Systems"; Computers & Structures; Vol 20; N° 1-3; 1985.

[2.19] Ramalho, M.A.; "Apresentação de um Editor de Dados para Estruturas com Recursos Gráficos Acoplados"; Anais do Colloquia 1987, Univ. Federal do Rio Grande do Sul; P. Alegre; 1987.

[2.20] "TURBO PASCAL - Reference Manual"; Borland Internacional Inc.; Scotts Valley, CA; 1984.

[2.21] Wood, S.; "Using TURBO PASCAL"; Ed. Osborne McGraw-Hill; Berkeley, CA; 1986.

## CAPÍTULO 3

### Biblioteca de Elementos

#### 3.1. Introdução

Neste capítulo serão apresentados os tipos de elementos disponíveis para utilização no programa LS. Aliás, serão apresentados os elementos tipos 1 a 7. O elemento tipo 8, sapata rígida, cujo desenvolvimento apresenta alguns aspectos um pouco diferentes da teoria tradicional, está detalhado em capítulo à parte. Em resumo, são os seguintes os tipos de elementos aqui discutidos :

Elemento 1 : Treliça Espacial.

Elemento 2 : Barra Espacial.

Elemento 3 : Contorno.

Elemento 4 : Membrana.

Elemento 5 : Plano.

Elemento 6 : Tridimensional.

Elemento 7 : Placa e Casca

Os três primeiros elementos da lista apresentada, elementos de 1 a 3, são baseados numa formulação típica de análise matricial de estruturas. Sendo utilizado o processo dos deslocamentos, as respectivas matrizes de rigidez são montadas diretamente por equilíbrio. Um procedimento tão conhecido, dispensa maiores comentários, se bem que as referências [3.1], [3.2], [3.3] e [3.4] apresentem muito bem os aspectos gerais sobre esse assunto.

Na verdade, esses três primeiros elementos formam um conjunto muito potente para a solução de estruturas lineares. Elementos de treliça e barra para as peças propriamente ditas e o elemento contorno para a definição de condições de apoio especiais. Aliados à grande

capacidade de processamento do programa, podem ser muito úteis na análise de estruturas como pórticos e treliças tridimensionais com muitos graus de liberdade.

A partir do elemento número 4, a utilização se volta para os meios contínuos, baseando-se esses elementos no Método dos Elementos Finitos. Sobre essa técnica, extremamente disseminada, também não será necessário tecer muitos comentários. Entretanto, no intuito de apenas apresentar as notações utilizadas, será elaborado um pequeno resumo sobre a utilização do método. Desse modo, ficará mais cômodo para cada elemento ser discutido, com os desenvolvimentos sendo feitos de forma mais direta, sem preocupação com detalhes a respeito da origem das equações utilizadas.

É interessante mencionar que, ao contrário dos elementos baseados em conceitos de pura análise matricial de estruturas, para os elementos finitos têm-se muitas formulações para resolver um mesmo problema. Como o método baseia-se na discretização do domínio total através da aplicação de funções aproximadoras por subdomínios, fica claro que para cada função haverá um comportamento diferente, gerando por sua vez resultados que podem estar mais ou menos próximos dos valores exatos.

Portanto, se as funções aproximadoras de um determinado elemento são pobres é necessário uma discretização em maior número de elementos. Em caso contrário, a rede pode ser menos densa, mas sempre haverá necessidade de um maior esforço computacional para tratar cada elemento em particular. Na verdade, o que se ganha por um lado se perde por outro.

Neste programa, por uma questão de filosofia de trabalho, sempre se adotará elementos mais simples, e que portanto necessitam de uma discretização mais fina. Acredita-se ser essa a solução mais vantajosa no cômputo geral. As razões dessa crença estão baseadas na maneira de se trabalhar com as estruturas que realmente ocorrem em

casos práticos.

A verdade é que dificilmente são analisadas estruturas cujas condições de geometria e carregamentos permite o uso disseminado de grandes elementos. Pelo contrário, em casos usuais o que se verifica é a necessidade de discretização de detalhes e condições especiais, que praticamente exigem a utilização de pequenos elementos. Desse modo, pode haver um grande desperdício de esforço computacional, sem qualquer benefício a nível de resultados a serem obtidos.

Por outro lado, se os elementos são pobres, a grande desvantagem apontada é a dificuldade na entrada dos dados e análise dos resultados, já que devem ser utilizados muitos elementos. Novamente a prática mostra que com geradores eficientes e programas pós-processadores que tratem essas saídas graficamente, tais dificuldades são completamente eliminadas.

Dentro dessa filosofia mencionada é que se deu a escolha dos elementos a serem desenvolvidos. Sempre que possível são elementos isoparamétricos de grande simplicidade de formulação, se bem que relativamente eficientes. A única exceção é o elemento adotado para as placas. Nesse caso, não é possível a adoção de um elemento nessas condições e a escolha necessariamente teve que se basear em outros critérios.

Para encerrar-se este item introdutório, menciona-se que o capítulo está organizado de uma forma bastante simples. Inicialmente, após esta introdução, é apresentado o breve resumo a respeito do Método dos Elementos Finitos, cuja existência já se mencionou. Logo após, cada um em seu próprio item, estão os elementos desenvolvidos, com sua formulação, considerações sobre o desempenho e exemplos de interesse. Quanto ao desempenho, todos os tempos mencionados referem-se a análises feitas em microcomputador compatível IBM/AT de 12 MHertz, com coprocessador matemático 80287.

## 3.2. CONCEITOS BÁSICOS SOBRE O MÉTODO DOS ELEMENTOS FINITOS

### 3.2.1. Energia Potencial Total

Para um corpo contínuo, elástico-linear, pode-se escrever a energia potencial total, funcional do problema, como sendo :

$$\pi = \frac{1}{2} \int_V \tilde{\epsilon}^T \tilde{\sigma} dV - \int_V \tilde{U}^T \tilde{f}_V dV - \int_S \tilde{U}_S^T \tilde{f}_S dS - \sum_i \tilde{U}_i^T F_i \dots(3.1)$$

onde,

$\tilde{\epsilon}$  : estado de deformações.

$\tilde{\sigma}$  : estado de tensões.

$\tilde{U}$  : deslocamentos para a situação inicial.

$\tilde{f}_V$  : forças volumétricas.

$\tilde{f}_S$  : forças distribuídas na superfície.

$F_i$  : força concentrada num ponto  $i$ .

A primeira parcela diz respeito à energia potencial interna, armazenada no corpo. As demais são trabalhos realizados pelos carregamentos aplicados, volumétricos, distribuídos ou concentrados.

### 3.2.2. Princípio dos Deslocamentos Virtuais

Um desenvolvimento equivalente a expressar o equilíbrio do corpo é usar o princípio dos deslocamentos virtuais. Ele estabelece que o equilíbrio requer que, para qualquer campo de pequenos deslocamentos virtuais compatíveis imposto sobre o corpo, o trabalho total das

forças virtuais internas deve ser igual ao trabalho total das forças externas, ou seja,  $\delta\pi = 0$ . Isso pode se expresso pela equação :

$$\int_V \bar{\epsilon}^T \sigma dV = \int_V \bar{U}^T f_v dV + \int_S \bar{U}_S^T f_s dS + \sum_i \bar{U}_i^T F_i \quad \dots(3.2)$$

Onde,

$\bar{\epsilon} = \delta\epsilon =$  deformações virtuais.

$\bar{U} = \delta U =$  deslocamentos virtuais.

Convém ressaltar que o campo de deslocamentos, e portanto o de deformações dele decorrente, são compatíveis e devem satisfazer as condições de contorno em termos cinemáticos.

### 3.2.3. Subdivisão do Domínio

A equação 3.2 é extremamente importante pois representa uma relação entre as grandezas a serem consideradas na solução da estrutura mencionada. Entretanto, não é pela integração fechada de todo o domínio que se pretende resolver o problema. No caso, vai-se adotar a divisão desse domínio em "n" subdomínios. Esses subdomínios, chamados elementos finitos, estão interligados através de "N" pontos nodais, onde os deslocamentos que ocorrem na estrutura são comuns. Desse modo a equação 3.2. pode ser escrita como :

$$\sum_n \left[ \int_{V^{(n)}} \bar{\epsilon}^{(n)T} \sigma^{(n)} dV^{(n)} \right] =$$



$$\begin{aligned}
& + \sum_n \left[ \int_{V^{(n)}} \bar{U}^{(n)T} f_{\tilde{V}}^{(n)} dV^{(n)} + \int_{S^{(n)}} \bar{U}_s^{(n)T} f_{\tilde{S}}^{(n)} dS^{(n)} \right] + \\
& + \sum_i \bar{U}_i^T F_i \quad \dots(3.3)
\end{aligned}$$

### 3.2.4. Montagem das Equações de Equilíbrio

Pode-se perceber que na equação 3.3 as integrações a serem realizadas restringem-se aos subdomínios definidos. Entretanto não é apenas essa a simplificação pretendida. Os próprios deslocamentos que ocorrem dentro de cada elemento são aproximados por funções polinomiais dependentes de valores nodais, sendo que para o elemento "n" pode-se escrever :

$$\tilde{u}^{(n)} = \tilde{H}^{(n)} \tilde{U} \quad \dots(3.4)$$

onde,

$\tilde{u}^{(n)}$  : deslocamentos em coordenadas locais do elemento (por enquanto vai-se supor que o sistema local está apenas transladado em relação ao sistema global).

$\tilde{H}^{(n)}$  : matriz de interpolação dos deslocamentos (depende da função aproximadora escolhida).

$\tilde{U}$  : deslocamentos nos pontos nodais (vetor de  $3 \cdot N$  posições)

Dentro dessa visão, as deformações também podem ser escritas em função dos deslocamentos nodais através da relação :

$$\tilde{\xi}^{(n)} = \tilde{B}^{(n)} \tilde{U} \quad \dots(3.5)$$

onde,

$\varepsilon^{(n)}$  : deformações em coordenadas locais do elemento (mesma observação anteriormente feita para os deslocamentos).

$B^{(n)}$  : matriz que relaciona os deslocamentos globais com as deformações locais.

$U$  : deslocamentos nos pontos nodais.

Já as tensões podem ser relacionadas às deformações através da equação :

$$\sigma^{(n)} = C^{(n)} \varepsilon^{(n)} + \sigma_I^{(n)} \quad \dots(3.6)$$

onde,

$\sigma^{(n)}$  : tensões em coordenadas locais.

$C^{(n)}$  : matriz que relaciona as tensões com as deformações (parâmetros elásticos do elemento).

$\varepsilon^{(n)}$  : deformações em coordenadas locais.

$\sigma_I^{(n)}$  : tensões iniciais atuantes no elemento (por ex. efeitos térmicos).

Substituindo-se as relações 3.4 a 3.6 na equação 3.3, obtém-se :

$$\begin{aligned} \bar{U}^T \left[ \sum_n \int_{V^{(n)}} B^{(n)T} C^{(n)} B^{(n)} dV^{(n)} \right] U = \\ \bar{U}^T \left[ \left\{ \sum_n \int_{V^{(n)}} H^{(n)T} f_V^{(n)} dV^{(n)} \right\} + \right. \\ \left. + \left\{ \sum_n \int_{S^{(n)}} H_S^{(n)T} f_S^{(n)} dS^{(n)} \right\} + \right. \\ \left. - \left\{ \sum_n \int_{V^{(n)}} B^{(n)T} \sigma_I^{(n)} dV^{(n)} \right\} + F_C \right] \dots(3.7) \end{aligned}$$

Fazendo-se os deslocamentos virtuais unitários, cada um a seu próprio tempo, tem-se que  $\bar{U}^T$  torna-se uma matriz identidade. Desse modo a equação 3.7 pode ser escrita como sendo :

$$R U = F \quad \dots(3.8)$$

onde,

$$R = \sum_n \int_{V^{(n)}} B^{(n)T} C^{(n)} B^{(n)} dV^{(n)} = \sum_n R^{(n)}$$

$$F = F_V + F_S + F_I + F_C$$

$$F_V = \sum_n \int_{V^{(n)}} H^{(n)T} f_V^{(n)} dV^{(n)} = \sum_n F_V^{(n)}$$

$$F_S = \sum_n \int_{S^{(n)}} H_S^{(n)T} f_S^{(n)} dS^{(n)} = \sum_n F_S^{(n)}$$

$$F_I = - \sum_n \int_{V^{(n)}} B^{(n)T} \phi_I^{(n)} dV^{(n)} = - \sum_n F_I^{(n)}$$

Esse procedimento de solução das estruturas é muito interessante pois permite que a montagem do sistema de equações seja feito elemento por elemento, simplificando sobremaneira o processo e tornando-o passível de automatização.

Como último detalhe, menciona-se que o o desenvolvimento das equações da maneira como foi mostrado costuma ser chamado "método da rigidez direta". Nesse caso, a matriz de rigidez R e o vetor de cargas F são encontrados pela simples adição das respectivas matrizes e vetores dos elementos. Isso é possível pois todas essas variáveis

encontram-se referenciadas a um vetor que contém a totalidade dos deslocamentos.

Nos elementos desenvolvidos neste trabalho, essa situação será um pouco alterada, sendo as matrizes e vetores montados em relação apenas aos deslocamentos dos nós ligados ao elemento em questão. A correta colocação dos valores no sistema global é realizada pela utilização dos já mencionados vetores de contribuição (ver item 2.5). Entretanto, a base dos procedimentos aqui mostrados não será alterada por essa mudança.

### 3.3. ELEMENTO TIPO 1 - TRELIÇA

#### 3.3.1. Caracterização do Elemento

O elemento treliça é definido por dois nós de extremidade e pode ser colocado em qualquer posição do espaço definido pelo sistema de referência global da estrutura, conforme mostra-se na figura 3.1.

Genericamente falando tem três graus de liberdade por nó. Esses graus de liberdade são as três translações de um ponto no espaço, pois o elemento apenas transmite força normal entre seus nós de extremidade.

O material a ser definido para os elementos necessita das seguintes características :

$E$  : módulo de elasticidade longitudinal.

$\alpha$  : coeficiente de dilatação térmica (opcional).

$\gamma$  : peso específico (opcional).

Quanto às propriedades das seções, apenas deve ser fornecido o valor de  $A$ , área da seção transversal.

Em termos de carregamento nos elementos, é permitida a definição de peso próprio, segundo os três eixos globais, e uma variação constante de temperatura,  $\Delta T$ , ao longo do comprimento.

Resta destacar que sua utilização principal dá-se exatamente nas estruturas chamadas de treliças. Além disso, apenas algumas modelagens especiais, como por exemplo blocos de concreto analisados sob o aspecto de estruturas compostas de bielas de concreto comprimidas e tirantes de aço tracionados. Podem também ser utilizados para modelar tirantes em geral, quando essas peças aparecem em outras estruturas.

### 3.3.3. Coordenadas Locais e Globais

O elemento treliça, em termos de sistema de referência local, possui apenas um eixo  $x_1$  orientado de seu nó inicial para o nó final. Já as coordenadas, onde se desenvolvem os parâmetros de forças e deslocamentos, são duas, uma para cada extremidade. O sistema e as coordenadas locais encontram-se mostradas na figura 3.2.

Nessas coordenadas podem ser definidos os vetores de forças e deslocamentos locais  $f$  e  $u$  :

$$\underset{\sim}{f} = \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} \quad \text{e} \quad \underset{\sim}{u} = \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad \dots(3.9)$$

Quanto às coordenadas globais, ou seja forças e deslocamentos segundo o sistema de referência global da estrutura, a situação é a apresentada pela figura 3.3. Observando-se a referida figura, percebe-se que os vetores de forças e deslocamentos terão agora seis elementos e podem ser escritos como sendo :

$$\underset{\sim}{F} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{Bmatrix} \quad \text{e} \quad \underset{\sim}{U} = \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \end{Bmatrix} \quad \dots(3.10)$$

A relação entre esforços e deslocamentos locais e globais pode ser definida através de uma matriz de transformação  $\lambda$  segundo as relações que se seguem :

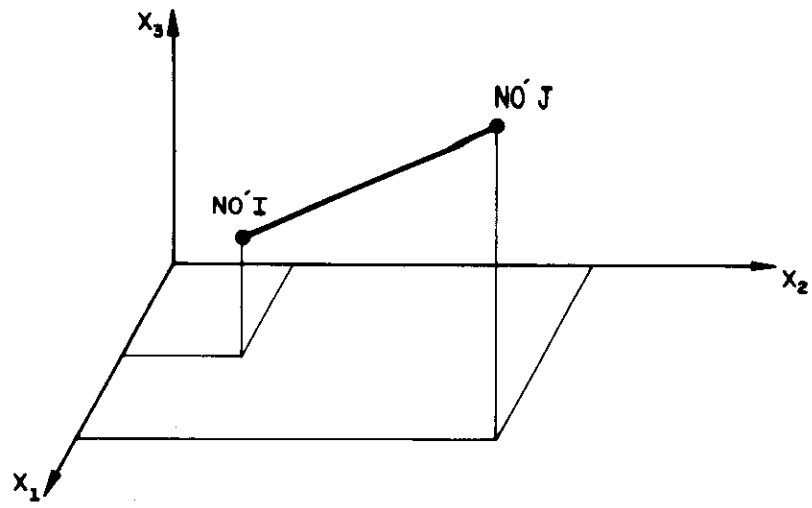


fig. 3.1 Elemento treliça

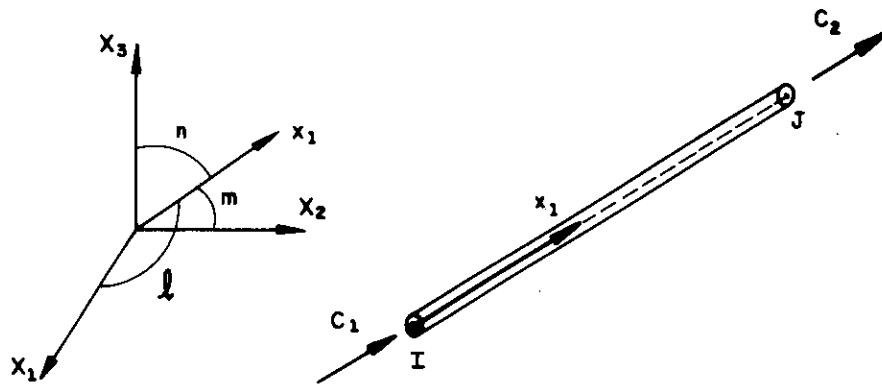


fig. 3.2 Coordenadas locais e seus cossenos diretores

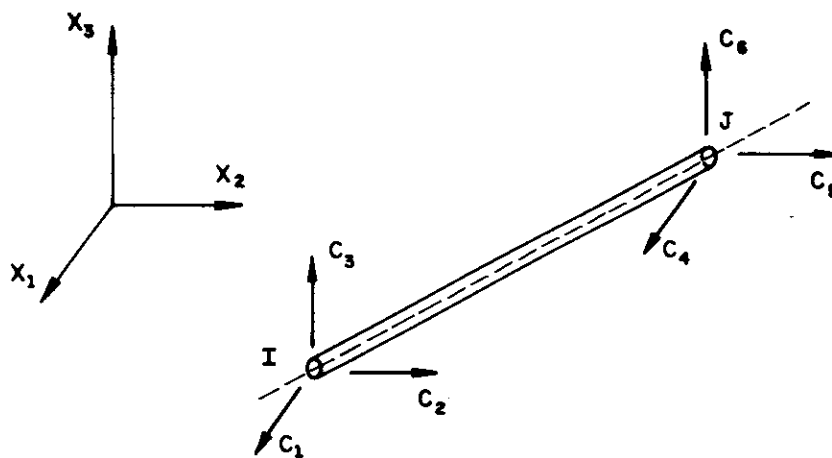


fig. 3.3 Coordenadas globais

$$\begin{matrix} f & = & \lambda & F \\ \sim & & \sim & \sim \end{matrix} \quad \dots(3.11)$$

$$\begin{matrix} u & = & \lambda & U \\ \sim & & \sim & \sim \end{matrix}$$

A matriz  $\lambda$ , que relaciona esforços ou deslocamentos globais com esforços ou deslocamentos locais, é calculada pelos cossenos diretores do eixo local  $x_1$  em relação aos eixos globais  $X_1$ ,  $X_2$  e  $X_3$ . A já referida figura 3.2 mostra esses cossenos, chamados de  $l$ ,  $m$  e  $n$ , respectivamente. Dessa maneira, a matriz  $\lambda$  pode ser escrita como se segue :

$$\begin{matrix} \lambda & = & \begin{bmatrix} l & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & n \end{bmatrix} \\ \sim & & \end{matrix} \quad \dots(3.12)$$

### 3.3.3. Matriz de Rigidez Global

Antes da determinação da matriz de rigidez do elemento em relação aos eixos globais de referência, é interessante que inicialmente sejam desenvolvidas considerações sobre a matriz de rigidez para as coordenadas locais. Essa é a matriz que relaciona as forças e deslocamentos nessas referências, segundo a equação :

$$\begin{matrix} f & = & r & u \\ \sim & & \sim & \sim \end{matrix} \quad \dots(3.13)$$

Considerando-se os parâmetros elásticos e geométricos do elemento, sua expressão pode ser obtida da referência [3.2] como sendo :

$$\begin{matrix} r & = & \frac{A E}{L} & \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \\ \sim & & \end{matrix} \quad \dots(3.14)$$



onde,

L : comprimento do elemento.

Entretanto como o objetivo final é a determinação da matriz de rigidez em relação às coordenadas globais, utilizando-se as expressões 3.11 e 3.13 pode-se escrever :

$$\underset{\sim}{\lambda} \underset{\sim}{F} = \underset{\sim}{r} \underset{\sim}{\lambda} \underset{\sim}{U} \quad \dots(3.15)$$

Verificando-se que, no caso particular aqui estudado, a matriz  $\lambda$  é de rotação, sendo portanto sua inversa igual à transposta, tem-se :

$$\underset{\sim}{F} = \underset{\sim}{\lambda}^T \underset{\sim}{r} \underset{\sim}{\lambda} \underset{\sim}{U} \quad \dots(3.16)$$

Ou seja, a matriz de rigidez do elemento em relação aos eixos de referência global resulta :

$$\underset{\sim}{RE} = \underset{\sim}{\lambda}^T \underset{\sim}{r} \underset{\sim}{\lambda} \quad \dots(3.17)$$

Calculando-se a expressão 3.17, tendo em vista as equações 3.12 e 3.14, obtém-se :

$$\underset{\sim}{RE} = \frac{A E}{L} \begin{bmatrix} \underset{\sim}{r}^0 & - \underset{\sim}{r}^0 \\ - \underset{\sim}{r}^0 & \underset{\sim}{r}^0 \end{bmatrix} \quad \dots(3.18)$$

onde,

$$\underset{\sim}{r}^0 = \begin{bmatrix} l^2 & lm & ln \\ lm & m^2 & mn \\ ln & mn & n^2 \end{bmatrix}$$

### 3.3.4. Vetor de Cargas do Elemento

O vetor de cargas do elemento é calculado de modo trivial. Na verdade não é apenas um vetor, mas sim um vetor para cada caso de carregamento da estrutura.

Foi mencionado que o elemento treliça permite a definição de peso próprio, com frações fornecidas pelo usuário segundo os eixos globais, e ainda uma variação de temperatura constante em todo o elemento. Desse modo, o vetor de cargas PE é calculado através da expressão :

$$\tilde{PE} = \frac{\gamma A L}{2} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} + E A \alpha \Delta T \begin{pmatrix} 1 \\ m \\ n \\ -1 \\ -m \\ -n \end{pmatrix} \quad \dots(3.18)$$

onde,

$a_1$ ,  $a_2$  e  $a_3$  : frações do peso próprio a serem consideradas segundo os eixos  $X_1$ ,  $X_2$  e  $X_3$ .

### 3.3.5. Matriz da Relação Tensão / Deslocamento

No presente caso, os esforços e tensões a serem determinados são  $N$  e  $\sigma$ , respectivamente, esforço solicitante normal e tensão normal no elemento.

O vetor que contém esses valores pode ser encontrado com base nas forças locais, que aparecem na equação 3.13, referenciadas aos deslocamentos também locais. Entretanto, é necessário que os deslocamentos a serem considerados sejam os globais. Para tanto basta substituir a conveniente expressão 3.11 na já referida equação 3.13, obtendo-se :

$$\underset{\sim}{f} = \underset{\sim}{r} \underset{\sim}{\lambda} \underset{\sim}{U} \quad \dots(3.20)$$

onde,

$$\underset{\sim}{r} \underset{\sim}{\lambda} = \frac{A E}{L} \begin{bmatrix} 1 & m & n & -1 & -m & -n \\ -1 & -m & -n & 1 & m & n \end{bmatrix}$$

Com as adaptações convenientes, pode-se escrever a matriz TD, relação entre os deslocamentos globais e o esforço e a tensão a serem calculados, como sendo :

$$\underset{\sim}{TD} = \frac{E}{L} \begin{bmatrix} A1 & Am & An & -A1 & -Am & -An \\ -1 & -m & -n & 1 & m & n \end{bmatrix} \quad \dots(3.21)$$

### 3.3.6. Vetor de Esforços ou Tensões Iniciais

Também aqui, de modo semelhante ao vetor PE, para cada carregamento considerado deve ser montado um vetor de esforços e tensões iniciais. Esse vetor contém  $N_I$  e  $\sigma_I$ , respectivamente, esforço normal e tensão normal iniciais.

No elemento treliça, o único carregamento realmente interno é o efeito da variação da temperatura. O peso próprio é apenas calculado com as características do elemento, mas atua no modelo como força nodal. Desse modo o vetor TI resulta :

$$\underset{\sim}{TI} = - E \alpha \Delta T \begin{Bmatrix} A \\ 1 \end{Bmatrix} \quad \dots(3.22)$$

onde  $(E A \alpha \Delta T)$  é a força produzida pela variação da temperatura e  $(E \alpha \Delta T)$  a correspondente tensão normal.

Com este último vetor, o programa LS tem condições de calcular o esforço e a tensão finais do elemento, para cada

caso de carregamento, através da relação já mostrada no item 2.7 :  $T = TD U + TI$  .

### 3.3.7. Desempenho

Os resultados obtidos com o elemento treliça não dependem da discretização realizada, pois a expressão dos deslocamentos ao longo do elemento é exata, isto é, cada barra da treliça real pode ser discretizada com apenas um elemento treliça.

Resta então, quanto ao desempenho, apresentar uma noção do tempo de processamento necessário para a montagem e gravação das matrizes e vetores do elemento, e depois para o cálculo do esforço e da tensão resultantes. Esse tempo de processamento é muito pequeno em relação a outros tipos existentes. Pode-se adotar o valor de 0.034 seg para cada elemento estar com suas matrizes e vetores totalmente calculados e gravados em disco. Quanto ao cálculo das tensões, o tempo por elemento situa-se em torno de 0.020 seg.

Devido à grande simplicidade do elemento aqui considerado e à necessidade de se resumir essa apresentação, deixa-se de apresentar um exemplo de aplicação. Pouca coisa de interessante poderia ser comentada sobre essa análise.

### 3.4. ELEMENTO TIPO 2 - BARRA

#### 3.4.1. Caracterização do Elemento

O elemento barra, de modo semelhante ao treliça, também é definido por dois nós de extremidade e pode ser colocado em qualquer posição do espaço definido pelo sistema de referência global da estrutura. De modo geral, tem seis graus de liberdade por nó, três translações e três rotações, perfazendo um total de 12 parâmetros.

Um ponto de grande importância é a definição de planos de referência para o elemento. Ocorre que, de modo diverso do elemento treliça, a barra trabalha não apenas segundo seu eixo longitudinal. Trabalha também segundo planos ortogonais, dos quais o eixo longitudinal é o traço. Assim sendo, o usuário precisa definir a posição desses planos.

A definição desses planos está ligada à definição do sistema local de referência. Como pode-se observar pela figura 3.4, o eixo local  $x_1$  é definido pelo eixo longitudinal da barra, com sentido orientado do nó "I" para o nó "J". Os outros dois eixos, coincidentes com os eixos principais de inércia da seção transversal, é que precisam ser definidos pelo usuário. Existem duas maneiras de se fazer essa definição. Na primeira basta fornecer o eixo global da estrutura ao qual o eixo local  $x_2$  é paralelo. Entretanto, se esse paralelismo não existir o usuário pode fornecer um terceiro nó, na figura 3.4 o ponto "K", que define, juntamente com os nós "I" e "J" o plano que contém o eixo  $x_2$ . A partir daí o sistema local dextrorso estará construído, sendo o eixo  $x_3$  adotado normal ao plano formado pelos dois anteriormente definidos.

Os materiais a serem fornecidos para os elementos necessitam dos seguintes parâmetros :

$E$  : módulo de elasticidade longitudinal.  
 $G$  : módulo de elasticidade transversal.  
 $\gamma$  : peso específico (opcional)

Quanto às propriedades das seções, são as seguintes as definições necessárias :

$A$  : área da seção transversal.  
 $Ac_2$  : área resistente à cortante segundo  $x_2$  (opcional).  
 $Ac_3$  : área resistente à cortante segundo  $x_3$  (opcional).  
 $I_t$  : momento de inércia torçor.  
 $I_2$  : momento de inércia em torno do eixo  $x_2$ .  
 $I_3$  : momento de inércia em torno do eixo  $x_3$ .  
 $W_2$  : módulo de resist. à flexão em torno de  $x_2$  (opcional).  
 $W_3$  : módulo de resist. à flexão em torno de  $x_3$  (opcional).

É interessante mencionar que as áreas que efetivamente atuam na rigidez à força cortante podem ou não apresentar valor diferente de zero, dependendo da intenção do usuário de considerar ou não essa rigidez nos seus cálculos. De modo semelhante, o usuário também decide se deseja ou não o cálculo das tensões devidas à flexão nos elementos. Se os valores  $W_2$  e  $W_3$  forem fornecidos diferentes de zero, essa operação será executada. Caso contrário, as saídas estarão restritas aos valores dos esforços solicitantes.

Em termos de carregamento nos elementos, é permitida a definição de peso próprio, segundo os três eixos globais e ainda conjuntos de ações que atuam nas extremidades. Esses conjuntos de ações são compostos por 12 valores de esforços que podem representar qualquer carregamento atuante no elemento, inclusive cargas distribuídas de vários tipos e efeitos térmicos. São esforços de engastamento perfeito que, para facilidade do usuário, podem ser montados automaticamente pelo programa gerador do arquivo de dados apresentado no item 2.10.

Para modelagem de detalhes especiais, o elemento barra

possui dois recursos interessantes. O primeiro é a liberação de deslocamentos e rotações junto a seus nós de extremidade, de acordo com opção do usuário. Desse modo, o elemento adquire uma flexibilidade muito grande, como se na verdade fosse uma série de elementos diferentes. Por exemplo, pela liberação das rotações nas duas extremidades, pode-se modelar treliças, como se o elemento utilizado fosse do tipo 1. O segundo recurso é o constrangimento de um grau de liberdade a outro, podendo-se garantir o mesmo deslocamento ou rotação para os dois graus de liberdade assim ligados. Esse procedimento é muito útil na modelagem de edifícios sob carga horizontal, onde os pavimentos devem ser considerados diafragmas rígidos em seu próprio plano.

Resta destacar que a utilização do elemento aqui discutido é muito ampla, abrangendo uma série enorme de estruturas. Pórticos de contraventamento de edifícios submetidos a ações horizontais, estruturas metálicas com peças submetidas à flexão e modelagens de vigas em pavimentos de edifícios são apenas alguns exemplos dessas aplicações.

#### **3.4.2. Coordenadas Locais e Globais**

Foi mencionado no item anterior o sistema de referência local do elemento barra. Em função desse sistema local é que se dá a orientação das 12 coordenadas onde se desenvolvem os deslocamentos ou esforços locais presentes no elemento. A figura 3.5 apresenta essas coordenadas. Em relação a elas pode-se definir os vetores de deslocamentos e esforços como se segue :

$$\underset{\sim}{f} = \begin{bmatrix} f_1 \\ f_2 \\ \cdot \\ \cdot \\ f_{11} \\ f_{12} \end{bmatrix} \quad \text{e} \quad \underset{\sim}{u} = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ u_{11} \\ u_{12} \end{bmatrix} \quad \dots(3.23)$$

Já as coordenadas relativas aos eixos globais de referência, também em número de 12, são mostradas na figura 3.6. Em relação a elas também são montados vetores de deslocamentos e esforços, U e F respectivamente, de modo totalmente análogo aos apresentados na expressão 3.23.

A relação entre esforços e deslocamentos locais e globais é definida através de uma matriz de transformação  $\lambda$ , quadrada de ordem 12, segundo as relações 3.11 já apresentadas. Essa matriz  $\lambda$ , encontrada na referência [3.2], é montada com os cossenos diretores dos eixos locais  $x_1, x_2$  e  $x_3$  em relação aos eixos globais  $X_1, X_2$  e  $X_3$ . A figura 3.7 mostra esses cossenos e esclarece a nomenclatura adotada. Desse modo a matriz  $\lambda$  pode ser expressa da maneira que se segue :

$$\underset{\sim}{\lambda} = \begin{bmatrix} \lambda_0 & 0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & \lambda_0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & \lambda_0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & 0 & \lambda_0 \\ \sim & \sim & \sim & \sim \end{bmatrix} \quad \dots(3.24)$$

onde,

$$\underset{\sim}{\lambda_0} = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix}$$



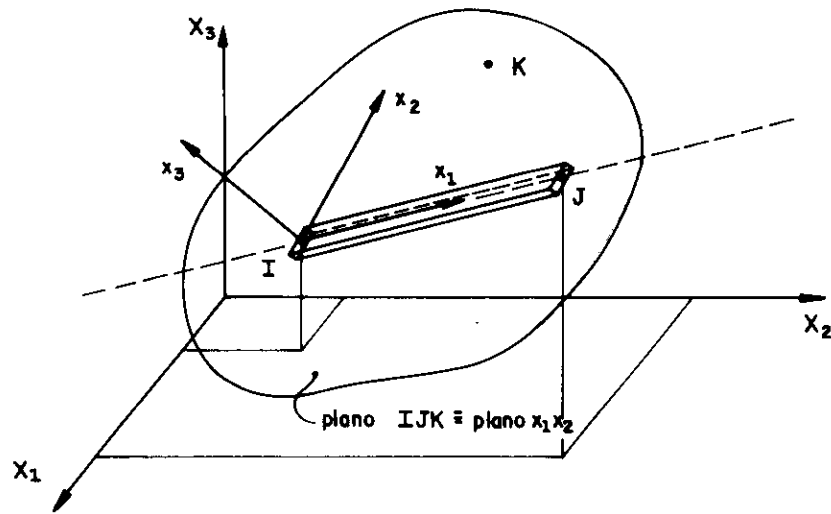


fig. 3.4 Sistema local do elemento barra

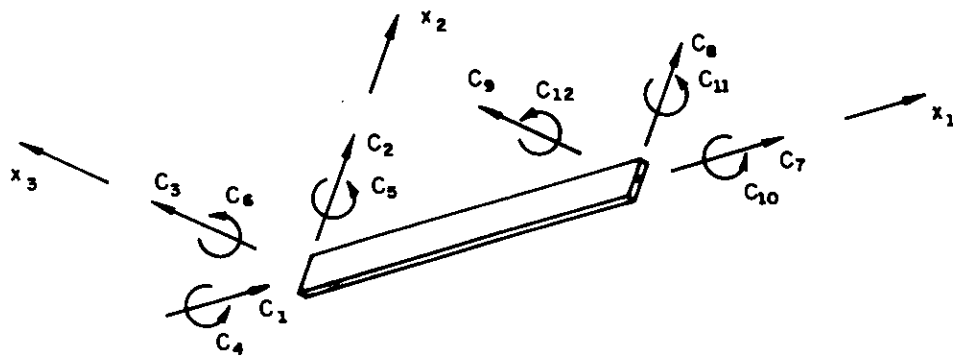


fig. 3.5 Coordenadas locais

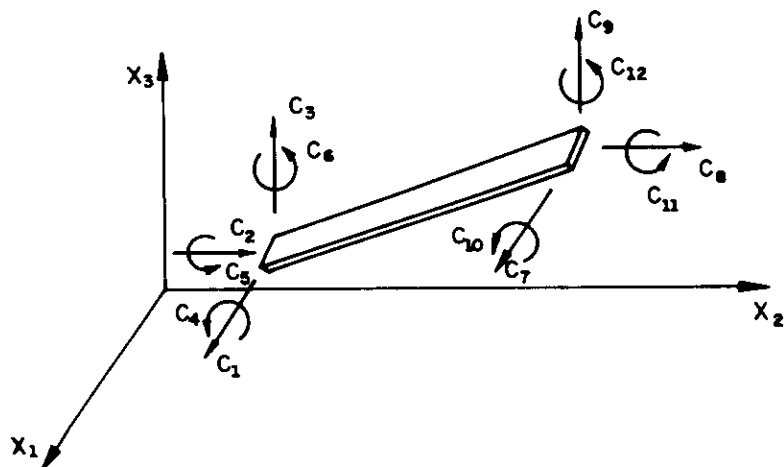


fig. 3.6 Coordenadas globais

### 3.4.3. Matriz de Rigidez em Coordenadas Locais

A matriz de rigidez do elemento barra, em relação às coordenadas locais, também está deduzida com detalhes na referência [3.2]. Em termos de submatrizes, pode ser expressa da seguinte maneira :

$$\tilde{r} = \begin{bmatrix} \tilde{r}_1 & \tilde{r}_2 & -\tilde{r}_1 & \tilde{r}_2 \\ & \tilde{r}_3 & \tilde{r}_4 & \tilde{r}_5 \\ & & \tilde{r}_1 & \tilde{r}_2 \\ \text{sim.} & & & \tilde{r}_3 \end{bmatrix} \quad \dots(3.25)$$

As submatrizes, de ordem 3, que compõem a matriz de rigidez local tem as seguintes expressões :

$$\tilde{r}_1 = \begin{bmatrix} \frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI_3}{L^3(1+\phi_2)} & 0 \\ 0 & 0 & \frac{12EI_2}{L^3(1+\phi_3)} \end{bmatrix} \quad \dots(3.26)$$

$$\tilde{r}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{6EI_3}{L^2(1+\phi_2)} \\ 0 & \frac{-6EI_2}{L^2(1+\phi_3)} & 0 \end{bmatrix} \quad \dots(3.27)$$

$$\underset{\sim}{r_2} = \begin{bmatrix} \frac{GI_t}{L} & 0 & 0 \\ 0 & \frac{(4+\phi_3)EI_2}{L(1+\phi_3)} & 0 \\ 0 & 0 & \frac{(4+\phi_2)EI_3}{L(1+\phi_2)} \end{bmatrix} \quad \dots(3.28)$$

$$\underset{\sim}{r_4} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{6EI_2}{L^2(1+\phi_3)} \\ 0 & \frac{-6EI_3}{L^2(1+\phi_2)} & 0 \end{bmatrix} \quad \dots(3.29)$$

$$\underset{\sim}{r_3} = \begin{bmatrix} \frac{-GI_t}{L} & 0 & 0 \\ 0 & \frac{(2-\phi_3)EI_2}{L(1+\phi_3)} & 0 \\ 0 & 0 & \frac{(2-\phi_2)EI_3}{L(1+\phi_2)} \end{bmatrix} \quad \dots(3.30)$$

Os parâmetros  $\phi_2$  e  $\phi_3$  representam a influência da força cortante. Se forem admitidos iguais a zero, a matriz toma a forma mais usual, levando em conta apenas a deformação por momento fletor nas direções  $x_2$  e  $x_3$ . A expressão dos referidos parâmetros é a que se segue :

$$\phi_2 = \frac{12 E I_3}{G A c_2 L^2} \quad \dots(3.31)$$

$$\phi_3 = \frac{12 E I_2}{G A c_3 L^2}$$

#### 3.4.4. Liberação de Deslocamentos e Rotações

Para a liberação de deslocamentos e rotações junto aos nós de extremidade, portanto alterando-se os graus de liberdade do elemento, é necessário realizar operações na matriz de rigidez local e nas cargas prescritas de extremidade.

Essas operações consistem em tornar nulos os elementos da linha e da coluna correspondentes à coordenada local que se deseja eliminar. Considerando-se "n", sempre de 1 a 12, o número da coordenada a ser suprimida, o roteiro dessas alterações pode ser resumido no seguinte :

i de 1 até 12

$$RAUX(i) = R(n,i)$$

i de 1 até 12

$$AUX(i) = R(i,n) / RAUX(n)$$

j de 1 até 12

$$R(i,j) = R(i,j) - AUX(i)*RAUX(j)$$

i de 1 até nc

$$p_{aux} = P(n,i)$$

j de 1 até 12

$$P(j,i) = P(j,i) - AUX(j)*p_{aux}$$

onde,

R : matriz de rigidez local (12 x 12).

P : cargas prescritas nas extremidades (12 x nc), em coordenadas locais.

nc : número de carregamentos.

#### 3.4.5. Matriz de Rigidez em Coordenadas Globais

A matriz de rigidez do elemento em relação aos eixos de referência globais é calculada em função da matriz local

e da matriz de transformação  $\lambda$ , conforme a equação 3.17. Em termos de submatrizes, pode-se escrever :

$$\begin{matrix} RE \\ \sim \end{matrix} = \begin{bmatrix} \lambda_0^T r_1 \lambda_0 & \lambda_0^T r_2 \lambda_0 & -\lambda_0^T r_1 \lambda_0 & \lambda_0^T r_2 \lambda_0 \\ \sim & \sim & \sim & \sim \\ & \lambda_0^T r_3 \lambda_0 & \lambda_0^T r_4 \lambda_0 & \lambda_0^T r_5 \lambda_0 \\ & \sim & \sim & \sim \\ & \text{sim.} & \lambda_0^T r_1 \lambda_0 & \lambda_0^T r_2 \lambda_0 \\ & & \sim & \sim \\ & & & \lambda_0^T r_3 \lambda_0 \\ & & & \sim \end{bmatrix} \dots(3.32)$$

A expressão 3.32 representa uma forma otimizada de se calcular a matriz de rigidez global dos elementos. Isso por que as operações de multiplicação, que teoricamente teriam que ser realizadas para matrizes de ordem 12, na verdade restringem-se a operações com submatrizes de ordem 3.

### 3.4.6. Vetor de Cargas do Elemento

Os carregamentos aplicados no elemento barra são de dois tipos. Inicialmente, o peso próprio, com frações fornecidas pelo usuário segundo os eixos globais. Depois, as forças e momentos de extremidade,  $p_1$  a  $p_{12}$ , definidos pelo usuário em relação às coordenadas locais e que podem simular a existência de qualquer tipo de carregamento sobre o elemento.

Essas forças e momentos nas extremidades precisam ser expressas em relação às coordenadas globais do elemento. Para tanto, basta fazer :

$$\begin{matrix} P \\ \sim \end{matrix} = \begin{matrix} \lambda \\ \sim \end{matrix} p \dots(3.33)$$

Desse modo, o vetor de cargas PE, para cada caso de

carregamento, é calculado através da expressão :

$$\tilde{P}E = \frac{\gamma A L}{2} \cdot \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ 0 \\ 0 \\ 0 \\ a_1 \\ a_2 \\ a_3 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \\ P_{10} \\ P_{11} \\ P_{12} \end{pmatrix} \quad \dots (3.34)$$

onde,

$a_1, a_2$  e  $a_3$  : frações do peso próprio a serem consideradas segundo os eixos  $X_1, X_2$  e  $X_3$ .

$P_1, \dots, P_{12}$  : ações de engastamento perfeito nos extremos, em coordenadas globais.

### 3.4.7. Matriz da Relação Tensão / Deslocamento

Para o elemento barra, a matriz que relaciona os esforços e tensões com os deslocamento pode ter duas dimensões diferentes. Caso não tenham sido fornecidos valores de módulos de resistência à flexão, ela será quadrada de ordem doze. Isso porque apenas os esforços solicitantes nas coordenadas locais serão considerados. Nesse caso, as saídas, para cada uma das extremidades, conterão os seguintes valores :

$N$  : esforço normal.

$V_2$  : força cortante segundo o eixo local  $x_2$ .

$V_3$  : força cortante segundo o eixo local  $x_3$ .

T : esforço torçor.

$M_2$  : momento fletor em torno do eixo local  $x_2$ .

$M_3$  : momento fletor em torno do eixo local  $x_3$ .

A matriz TD, quando os módulos de flexão não são definidos, é simplesmente calculada pela equação 3.20. Assim, basta pós-multiplicar a matriz de rigidez local do elemento pela matriz de transformação  $\lambda$ . O resultado, em termos de submatrizes, será o que se segue :

$$TD = \begin{bmatrix} \tilde{r}_1 \lambda_0 & \tilde{r}_2 \lambda_0 & -\tilde{r}_1 \lambda_0 & \tilde{r}_2 \lambda_0 \\ & \tilde{r}_3 \lambda_0 & \tilde{r}_4 \lambda_0 & \tilde{r}_5 \lambda_0 \\ & \text{sim.} & \tilde{r}_1 \lambda_0 & \tilde{r}_2 \lambda_0 \\ & & & \tilde{r}_3 \lambda_0 \end{bmatrix} \quad \dots(3.35)$$

Entretanto, se os módulos de flexão forem fornecidos com valores diferentes de zero, não serão apenas os esforços solicitantes nas coordenadas locais que devem estar presentes na saída dos resultados. O programa também providencia o cálculo das tensões em quatro pontos da seção transversal. Esses pontos são as extremidades da seção do elemento sobre os eixos locais  $x_2$  e  $x_3$ . São as seguintes as expressões dessas tensões :

$$\begin{aligned} \sigma_1 &= \frac{N}{A} + \frac{M_2}{W_2} \\ \sigma_2 &= \frac{N}{A} - \frac{M_2}{W_2} \\ \sigma_3 &= \frac{N}{A} + \frac{M_3}{W_3} \\ \sigma_4 &= \frac{N}{A} - \frac{M_3}{W_3} \end{aligned} \quad \dots(3.36)$$

Nesse caso, considerando-se que as mencionadas tensões são calculadas nas duas extremidades do elemento, a matriz TD será aumentada em 8 linhas. Essas novas linhas serão obtidas por combinação linear das doze originais. O roteiro de tal alteração é mostrado a seguir :

$$\begin{aligned}
 (13^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (1^{\text{a}} \text{ linha}) + \frac{1}{W_2} \cdot (5^{\text{a}} \text{ linha}) \\
 (14^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (1^{\text{a}} \text{ linha}) - \frac{1}{W_2} \cdot (5^{\text{a}} \text{ linha}) \\
 (15^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (1^{\text{a}} \text{ linha}) + \frac{1}{W_2} \cdot (6^{\text{a}} \text{ linha}) \\
 (16^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (1^{\text{a}} \text{ linha}) - \frac{1}{W_2} \cdot (6^{\text{a}} \text{ linha}) \\
 (17^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (7^{\text{a}} \text{ linha}) + \frac{1}{W_2} \cdot (11^{\text{a}} \text{ linha}) \\
 (18^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (7^{\text{a}} \text{ linha}) - \frac{1}{W_2} \cdot (11^{\text{a}} \text{ linha}) \\
 (19^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (7^{\text{a}} \text{ linha}) + \frac{1}{W_2} \cdot (12^{\text{a}} \text{ linha}) \\
 (20^{\text{a}} \text{ linha}) &= \frac{1}{A} \cdot (7^{\text{a}} \text{ linha}) - \frac{1}{W_2} \cdot (12^{\text{a}} \text{ linha})
 \end{aligned}$$

...(3.37)

#### 3.4.8. Vetor de Esforços ou Tensões Iniciais

Também aqui, de modo semelhante ao vetor PE, para cada carregamento considerado deve ser montado um vetor de esforços e tensões iniciais. Esse vetor pode ter doze elementos, caso os módulos de resistência à flexão não tenham sido fornecidos, ou vinte elementos, se os módulos foram fornecidos.

Entretanto sua montagem é trivial. Caso os módulos não tenham sido fornecidos, seus doze elementos serão exatamente os valores de cargas prescritas, em coordenadas globais, multiplicadas por - 1. Assim, tem-se :



$$\tilde{T}I^T = - \left\{ P_1 \ P_2 \ \dots \ P_{11} \ P_{12} \right\} \quad \dots(3.38)$$

Se os módulos de flexão tiverem valor diferente de zero, simplesmente o vetor será completado com elementos nulos da 13<sup>a</sup> à 20<sup>a</sup> posição.

### 3.4.9. Constrangimento de Graus de Liberdade

Quando o elemento possui nó de extremidade com deslocamento ou rotação constrangido a outro nó, são necessárias algumas transformações na matriz de rigidez RE e na matriz de relação tensão e ou esforço por deslocamento, TD. Essas transformações são realizadas após a completa determinação dessas variáveis, como último procedimento antes da gravação em arquivo.

As mencionadas transformações são de natureza bastante diferentes, dependendo do fato de ser um deslocamento ou uma rotação o grau de liberdade a ser "escravizado" a outro. Mais especificamente, quando se tratar de uma rotação, basta colocar a rigidez que lhe seria atribuída para a equação correspondente ao grau de liberdade do nó "senhor". Isso é feito simplesmente pela troca do número da equação na posição correspondente do vetor de contribuições nodais do elemento. Nesse caso, nenhuma alteração na matriz TD será necessária.

Entretanto, se o grau de liberdade a ser escravizado é um deslocamento, o procedimento torna-se um pouco mais complexo. Em primeiro lugar as rigidezes que lhe seriam atribuídas são agora levadas para a equação correspondente ao novo grau de liberdade, de maneira análoga ao caso das rotações. Depois, é necessário que as rotações do nó "senhor", que podem estar a ele relacionadas, tornem-se graus de liberdade do elemento, implicando num aumento das

dimensões da matriz de rigidez. Por exemplo, se um determinado nó "n" tem o seu deslocamento segundo  $X_1$  escravizado a um outro nó "m", as rotações de "m" em torno de  $X_2$  e  $X_3$  tornam-se graus de liberdade do elemento que contém o nó "n". Nessas novas linhas da matriz de rigidez são colocadas as rigidezes do grau de liberdade escravo multiplicadas pelas distâncias convenientes.

Além disso, é necessário fazer-se uma alteração na matriz TD. Essa alteração consiste numa ampliação do número de colunas de modo a compatibilizar a matriz com o aumento do número de graus de liberdade do elemento. Nessas novas colunas, estará colocada a coluna correspondente ao grau de liberdade escravizado, multiplicada pelas distâncias respectivas.

Com essas alterações, a montagem da matriz de rigidez global, a obtenção dos deslocamentos nodais e ainda o cálculo dos esforços ou tensões nos elementos, serão normalmente processados, como se os constrangimentos não estivessem sendo definidos.

#### 3.4.10. Desempenho

Quanto à precisão dos resultados obtidos, o elemento barra tem comportamento semelhante ao elemento treliça. A função de deslocamentos é exata para suas condições de utilização.

Já com respeito ao tempo de processamento, ele pode variar dependendo do fato de estarem ou não sendo utilizados alguns recursos especiais como liberação ou constrangimento de graus de liberdade. Entretanto, como valor médio, pode-se adotar algo próximo de 0.15 seg para a montagem e gravação das matrizes e vetores de cada elemento. Para o cálculo dos esforços e tensões o tempo verificado é de aproximadamente 0.065 seg.

### 3.4.11. Exemplos

#### 3.4.11.1. Viga Engastada

Através deste exemplo muito simples, procurar-se-á discutir a influência da consideração do esforço cortante nos deslocamentos obtidos. Para tanto vai-se adotar uma viga em balanço de 3 m de vão e submetida a uma carga concentrada, na extremidade livre, de 10 KN. O módulo de elasticidade adotado é 2000 KN/cm<sup>2</sup>.

A referência [3.5] traz considerações a respeito do cálculo da área que efetivamente resiste à força cortante. Esse valor é encontrado em função da área da seção transversal multiplicada por um coeficiente de ajuste, que depende da forma da seção. Ou seja :

$$A_c = c A \quad \dots(3.39)$$

Neste exemplo, vai-se adotar seções retangulares e o valor de "c" resultará aproximadamente 1.2.

Para um melhor acompanhamento da tendência da solução do problema, serão adotadas três diferentes seções: 12x100, 12x50, 12x30, em centímetros. Portanto, as relações altura da seção, H, e vão, L, serão 1/3, 1/6 e 1/10, respectivamente. Os resultados obtidos, também em centímetros, encontram-se organizados na tabela 3.1, onde  $\delta_M$  representa os deslocamentos devidos exclusivamente ao momento fletor e  $\delta_{M+V}$  os devidos à consideração do momento fletor e da força cortante.

Pela observação da tabela 3.1, pode-se concluir que a variação dos resultados não é muito significativa. Mesmo para relação H/L igual a 1/3, onde seria esperada a maior diferença pois a parcela devida à força cortante é relativamente maior, a diferença é da ordem de 5%.

Seção	H/L	$\delta_M$	$\delta_{M+V}$	Dif %
12x100	1/3	0.0450	0.0474	5.3
12x50	1/6	0.3600	0.3648	1.3
12x30	1/10	1.6667	1.6746	0.5

tabela 3.1

#### 3.4.11.2. Edifício Sob Carga Horizontal

Neste exemplo, vão ser apresentados alguns resultados obtidos da análise de um edifício de 1 pavimento térreo e mais 19 pavimentos tipos, com um total de 55 m de altura, submetido à ação do vento.

A estrutura, discretizada com 422 nós e 640 elementos barra, é apresentada em perspectiva na figura 3.8. O módulo de elasticidade do material componente das peças é de  $2 \times 10^7$  KN/m<sup>2</sup> e foram definidas 15 propriedades de seções. Quanto ao carregamento, ele atua segundo o eixo  $X_2$  e foi calculado com base na NBR-6123.

É interessante mencionar que todos os nós de um pavimento foram forçados a apresentar um mesmo deslocamento segundo o eixo  $X_2$ . Isso foi feito mediante a escravização dos graus de liberdade segundo  $X_2$  de todos os nós do pavimento em relação ao primeiro, onde atua a carga total do nível. Desse modo, a laje de concreto estará sendo considerada como um diafragma rígido, sendo esse um bom modelo para o cálculo de um edifício nessas condições.

Os resultados da análise realizada encontram-se organizados na tabela 3.2. São apresentados deslocamentos e rotações para o primeiro nó de cada pavimento, ou seja, o nó onde todos os graus de liberdade segundo  $X_2$  estão escravizados. Os deslocamentos estão em metros e as

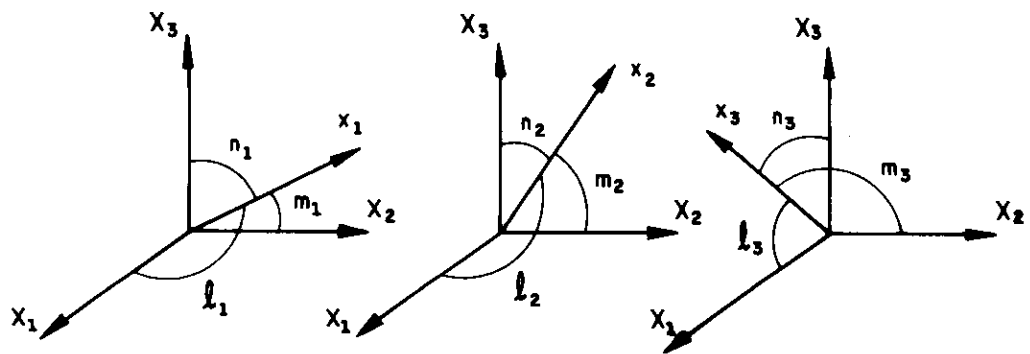


fig. 3.7 Cossenos diretores dos eixos locais

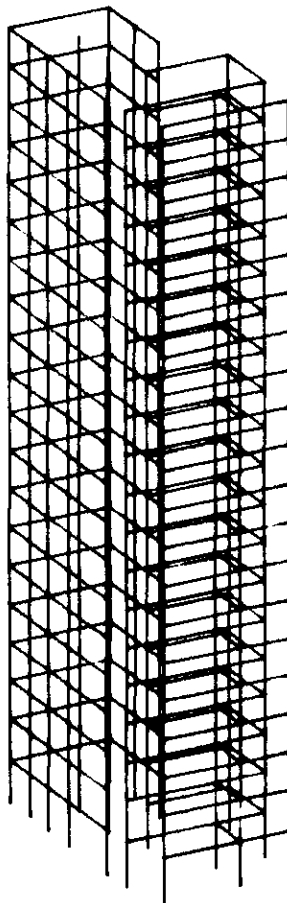


fig. 3.8 Discretização do pórtico tridimensional

rotações em radianos.

Devido à escravização dos graus de liberdade, qualquer nó do pavimento tem o mesmo deslocamento segundo  $X_2$ . Entretanto, os demais valores de deslocamentos e rotações são diferentes. Observando-se os resultados, percebe-se que o edifício apresenta, pela atuação do carregamento devido ao vento, uma flecha na extremidade superior da ordem de 12 cm. Também pode ser verificada a variação desse deslocamento ao longo da altura da estrutura, resultando aproximadamente linear para este caso específico.

Pav.	Nó	Trans. $X_2$	Trans. $X_3$	Rotação $X_1$	Rotação $X_2$
Térreo	21	.0015825	-.0000919	-.0009765	-.0000080
1º Pav	41	.0055564	-.0001782	-.0016611	-.0000148
2º Pav	61	.0107579	-.0002541	-.0021146	-.0000203
3º Pav	81	.0170115	-.0003255	-.0024352	-.0000261
4º Pav	101	.0239922	-.0003917	-.0026500	-.0000312
5º Pav	121	.0314400	-.0004524	-.0027798	-.0000359
6º Pav	141	.0391463	-.0005073	-.0028416	-.0000401
7º Pav	161	.0469440	-.0005566	-.0028488	-.0000437
8º Pav	181	.0546991	-.0006003	-.0028123	-.0000469
9º Pav	201	.0623053	-.0006385	-.0027413	-.0000497
10º Pav	221	.0696782	-.0006715	-.0026434	-.0000520
11º Pav	241	.0767536	-.0006997	-.0025251	-.0000539
12º Pav	261	.0834841	-.0007234	-.0023925	-.0000555
13º Pav	281	.0898384	-.0001609	-.0022645	-.0000490
14º Pav	301	.0958004	-.0007587	-.0021070	-.0000577
15º Pav	321	.1013702	-.0007712	-.0019657	-.0000584
16º Pav	341	.1065651	-.0007806	-.0018344	-.0000588
17º Pav	361	.1114218	-.0007874	-.0017196	-.0000604
18º Pav	381	.1159998	-.0007919	-.0016343	-.0000532
19º Pav	401	.1203868	-.0007943	-.0015755	-.0001000

tabela 3.2

### 3.5. ELEMENTO TIPO 3 - CONTORNO

#### 3.5.1. Caracterização do Elemento

O elemento contorno é utilizado para a implementação de condições de vinculação especiais e cálculo de reações em nós da estrutura. Basicamente, sua aplicação estende-se a todos os casos que possam ser simulados pela atuação de um mola sobre um ponto nodal.

Assim sendo, vinculações rígidas ou elásticas e ainda deslocamentos e rotações prescritas, segundo os eixos globais ou uma direção qualquer do espaço, podem ser facilmente simulados através da utilização do elemento. Nesse caso, as reações nodais advindas desse impedimento ou prescrição, são calculadas e apresentadas como resultado.

O elemento pode ser praticamente dividido em dois casos básicos. Em primeiro lugar, a situação onde a atuação dos impedimentos ou prescrições faz-se segundo os eixos de referência global da estrutura. Nesse caso são as seguintes as características a serem fornecidas como dados :

- N : ponto nodal ao qual liga-se o elemento.
- $K_1$  a  $K_6$  : constantes de mola que atuam segundo os seis graus de liberdade do nó (se o valor fornecido for 1, o grau de liberdade correspondente é considerado rigidamente impedido).
- $D_1$  a  $D_6$  : deslocamentos e rotações prescritas segundo os seis graus de liberdade do elemento.

Caso o elemento esteja atuando segundo uma direção qualquer, os parâmetros a serem fornecidos são diferentes :

- N : ponto nodal ao qual liga-se o elemento.
- NR : ponto nodal que define a direção de atuação.

$K_d$  : constante de mola para o deslocamento na direção do eixo do elemento (se igual a 1, rigidamente impedido).

$K_r$  : constante de mola para a rotação em torno do eixo do elemento (se igual a 1, rigidamente impedida).

D : deslocamento prescrito.

R : rotação prescrita.

### 3.5.2. Elemento Atuando Segundo os Eixos Globais

Para o caso de atuação segundo os eixos globais da estrutura, a montagem das matrizes e vetores do elemento é absolutamente trivial. Isso se dá pois as coordenadas locais são perfeitamente coincidentes com as globais, conforme mostra-se na figura 3.9.

Pode-se começar pela matriz de rigidez global RE, que resulta simplesmente diagonal :

$$\tilde{R}_E = \begin{bmatrix} K_1 & & & & & \\ & K_2 & & & & \\ & & K_3 & & & \\ & & & K_4 & & \\ & & & & K_5 & \\ & & & & & K_6 \end{bmatrix} \quad \dots(3.40)$$

Já para a determinação do vetor de cargas, um para cada carregamento da estrutura, pode-se escrever :

$$\tilde{P}_E = \begin{Bmatrix} K_1 D_1 \\ K_2 D_2 \\ K_3 D_3 \\ K_4 D_4 \\ K_5 D_5 \\ K_6 D_6 \end{Bmatrix} \quad \dots(3.41)$$

Quanto à matriz da relação dos esforços com os



deslocamentos, para o caso aqui considerado, ela resulta exatamente igual à matriz de rigidez, ou seja :

$$\underset{\sim}{TD} = \underset{\sim}{RE} \quad \dots(3.42)$$

Por fim deve-se mencionar que os vetores de esforços iniciais,  $\underset{\sim}{TI}$ , serão simplesmente carregados com zeros.

### 3.5.3. Elemento Atuando Segundo Direção Qualquer

No caso do elemento atuar segundo uma direção qualquer do espaço, o equacionamento é um pouco diferente. Inicialmente menciona-se que está definido um eixo local  $x_1$ , orientado do nó de atuação para o nó de referência, na direção do elemento. As coordenadas locais, apresentadas na figura 3.10 juntamente com a referência local, compõem-se de apenas dois valores. Um deslocamento e uma rotação, respectivamente na direção e em torno do eixo do elemento.

Como as coordenadas globais continuam sendo as apresentadas na figura 3.9, é necessário a definição de uma matriz de transformação  $\lambda$ , de acordo com as equações 3.11. Essa matriz é obtida pelo arranjo conveniente dos cossenos diretores do eixo local  $x_1$  em relação aos eixos globais da estrutura. Esses cossenos diretores são mostrados na figura 3.11 e a matriz  $\lambda$  resulta :

$$\underset{\sim}{\lambda} = \begin{bmatrix} 1 & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & m & n \end{bmatrix} \quad \dots(3.43)$$

Já a matriz de rigidez em relação às coordenadas locais pode ser escrita como se segue :

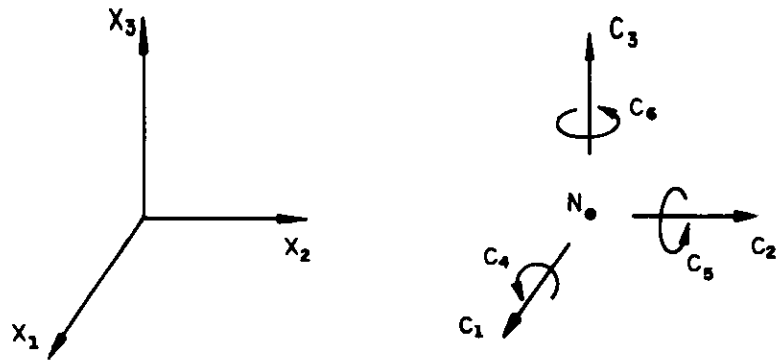


fig. 3.9 Sistema de referência e coordenadas globais

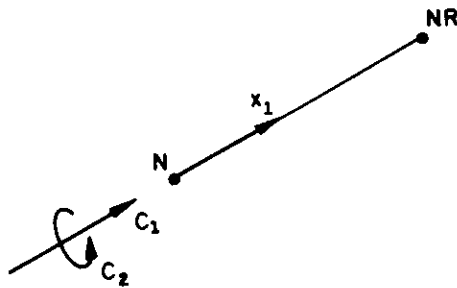


fig. 3.10 Sistema de referência e coordenadas locais

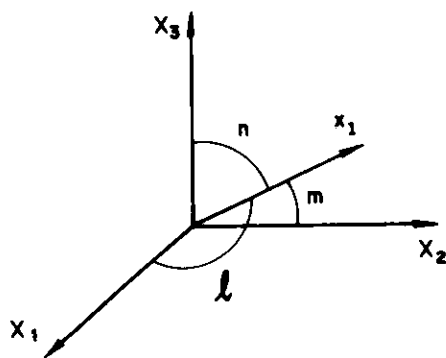


fig. 3.11 Cossenos diretores de  $x_1$

$$\tilde{r} = \begin{bmatrix} K_d & 0 \\ 0 & K_r \end{bmatrix} \quad \dots(3.44)$$

Através da equação 3.16, pode-se encontrar a matriz de rigidez em função das coordenadas globais. A matriz RE resulta :

$$\tilde{RE} = \begin{bmatrix} K_d^{1^2} & K_d^{1m} & K_d^{1n} & & & & \\ K_d^{1m} & K_d^{m^2} & K_d^{mn} & & & & \\ K_d^{1n} & K_d^{mn} & K_d^{n^2} & & & & \\ & & & \tilde{0} & & & \\ & & & & K_r^{1^2} & K_r^{1m} & K_r^{1n} \\ & & & & K_r^{1m} & K_r^{m^2} & K_r^{mn} \\ & & & & K_r^{1n} & K_r^{mn} & K_r^{n^2} \end{bmatrix} \quad \dots(3.45)$$

O vetor de cargas também pode ser montado com facilidade, de acordo com a expressão que se segue :

$$\tilde{PE} = \begin{bmatrix} K_d^{D1} \\ K_d^{Dm} \\ K_d^{Dn} \\ K_r^{R1} \\ K_r^{Rm} \\ K_r^{Rn} \end{bmatrix} \quad \dots(3.46)$$

Já a matriz da relação dos esforços com os deslocamentos, matriz TD, pode ser obtida com a matriz de rigidez local pós-multiplicada pela matriz de transformação  $\lambda$ . Desse modo, tem-se :

$$\tilde{TD} = \begin{bmatrix} K_d^1 & K_d^m & K_d^n & 0 & 0 & 0 \\ 0 & 0 & 0 & K_r^1 & K_r^m & K_r^n \end{bmatrix} \quad \dots(3.47)$$

Por fim, ressalta-se que de modo semelhante ao caso do

elemento atuando segundo os eixos globais, os vetores  $T_i$  serão apenas completados com zeros.

#### 3.5.4. Desempenho

A única análise de desempenho que se pode realizar para o presente elemento diz respeito ao tempo de processamento necessário para a montagem e gravação das matrizes e vetores que lhe correspondem e, posteriormente, o cálculo e apresentação dos valores dos esforços obtidos.

Esses tempos são extremamente reduzidos devido à simplicidade da formulação do elemento. Para a etapa de montagem e gravação de matrizes e vetores pode-se adotar 0.04 seg para cada elemento. Já quanto ao cálculo das tensões e apresentação dos resultados, gasta-se em torno de 0.02 seg por elemento.

### 3.6. ELEMENTO TIPO 4 - MEMBRANA

#### 3.6.1. Caracterização do Elemento

O elemento membrana é utilizado para a análise de estados planos de tensão. É definido por três ou quatro pontos nodais e pode ser colocado em qualquer posição do espaço definido pelo sistema de referência global da estrutura, conforme mostra-se na figura 3.12.

Genericamente falando tem três graus de liberdade por nó. Eles são as três translações de um ponto no espaço, já que o elemento apenas possui rigidez no seu próprio plano. O sistema de referência local é composto por dois eixos normais,  $x_1$  e  $x_2$ . Esses eixos locais são definidos pela numeração dos nós do elemento, seja ele triangular ou quadrangular. O eixo local  $x_1$  está colocado sobre o lado "IJ", de "I" para "J". Já o eixo  $x_2$  é normal a  $x_1$ , conforme mostra-se na figura 3.13

O elemento permite apenas a consideração de material isotrópico e os parâmetros a serem fornecidos são :

$\gamma$  = peso específico do material (opcional)

$E$  = módulo de elasticidade longitudinal.

$\nu$  = coeficiente de Poisson.

$G$  = módulo de elasticidade transversal (opcional).

$\alpha$  = coeficiente de dilatação térmica (opcional).

Convém ressaltar que não sendo fornecido valor para  $G$ , módulo de elasticidade transversal, o programa realizará o cálculo automático de acordo com a expressão da teoria da elasticidade :

$$G = \frac{E}{2(1 + \nu)} \quad \dots(3.48)$$

Entretanto, caso seu valor seja diferente de zero, ele será considerado como fornecido, não sendo mais admitida a relação 3.48.

Quanto a detalhes especiais de modelagem, é interessante mencionar que o elemento permite a inclusão de quatro modos incompatíveis de deslocamentos. Esses modos incompatíveis são verdadeiros graus de liberdade adicionais, que melhoram sensivelmente a convergência dos resultados para a modelagem de estruturas submetidas a momentos fletores. Essa inclusão é feita normalmente, sendo considerada um procedimento "default". Caso o usuário não a deseje, é necessário informar ao programa através de uma variável específica na linha de comando do elemento.

Por apresentar rigidez apenas no seu próprio plano, o elemento também só resiste a carregamentos contidos nesse plano. E as cargas atuantes podem ser de três tipos. Em primeiro lugar é possível a definição de carregamento distribuído constante, na direção normal e na própria direção do lado "IJ" de cada elemento. Efeitos térmicos também são permitidos, sendo a variação de temperatura calculada como a diferença entre um valor de referência, dado para cada elemento, e a média entre as temperaturas nodais envolvidas. Por fim, é admitido o cálculo automático de peso próprio, através de frações atuantes segundo os eixos globais da estrutura.

A utilização do elemento aqui analisado é também bastante ampla, incluindo-se nessa relação todos os casos de estruturas bidimensionais carregadas exclusivamente no seu próprio plano. Vigas paredes, chapas diversas e demais estruturas que se enquadrem na especificação mencionada anteriormente podem ser citadas como exemplo.

Como últimas informações deste item introdutório, é necessário ressaltar que todo o desenvolvimento das matrizes e vetores relativos ao elemento estudado será feito para o elemento quadrangular. Em item específico, mais a frente, serão discutidas as implicações de se ter um

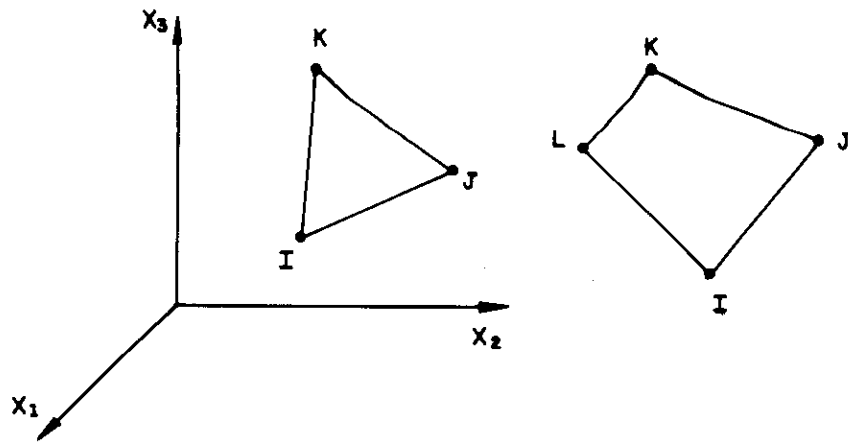


fig. 3.12 Elementos membrana de três e quatro nós

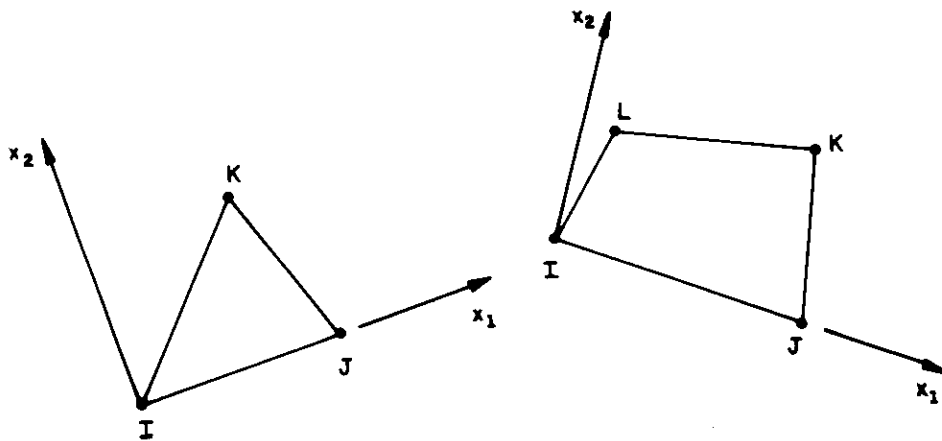


fig. 3.13 Sistema locais de referência

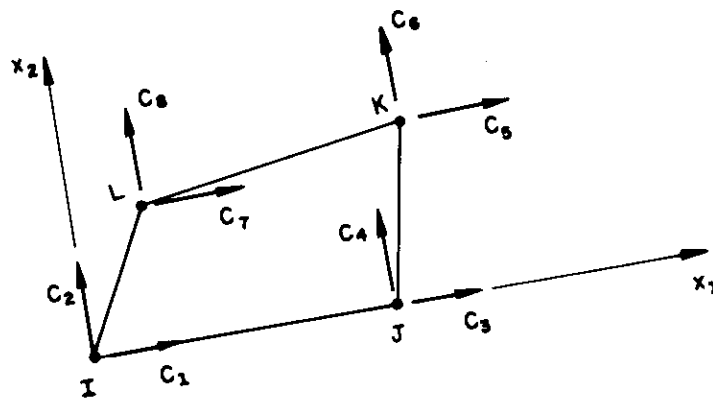


fig. 3.14 Coordenadas locais

elemento triangular. Por fim, menciona-se que esses desenvolvimentos foram feitos com base em diversas publicações, apresentadas como referências [3.6], [3.7], [3.8], [3.9], [3.10].

### 3.6.2. Coordenadas Locais e Globais

No item anterior foi definido um sistema local de referência. As coordenadas do elemento segundo esse sistema local são as apresentadas na figura 3.14. De modo semelhante ao que foi feito para os elementos anteriores, pode-se definir para os elementos de quatro nós os seguintes parâmetros :

$$\underset{\sim}{f} = \begin{pmatrix} f_1 \\ f_2 \\ \cdot \\ \cdot \\ f_7 \\ f_8 \end{pmatrix} \quad \text{e} \quad \underset{\sim}{u} = \begin{pmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ u_7 \\ u_8 \end{pmatrix} \quad \dots(3.49)$$

Quanto ao sistema de referência global, as coordenadas do elemento são apresentadas na figura 3.15. No caso, são três parâmetros para cada nó, resultando portanto nos seguintes vetores :

$$\underset{\sim}{F} = \begin{pmatrix} F_1 \\ F_2 \\ \cdot \\ \cdot \\ F_{11} \\ F_{12} \end{pmatrix} \quad \text{e} \quad \underset{\sim}{U} = \begin{pmatrix} U_1 \\ U_2 \\ \cdot \\ \cdot \\ U_{11} \\ U_{12} \end{pmatrix} \quad \dots(3.50)$$



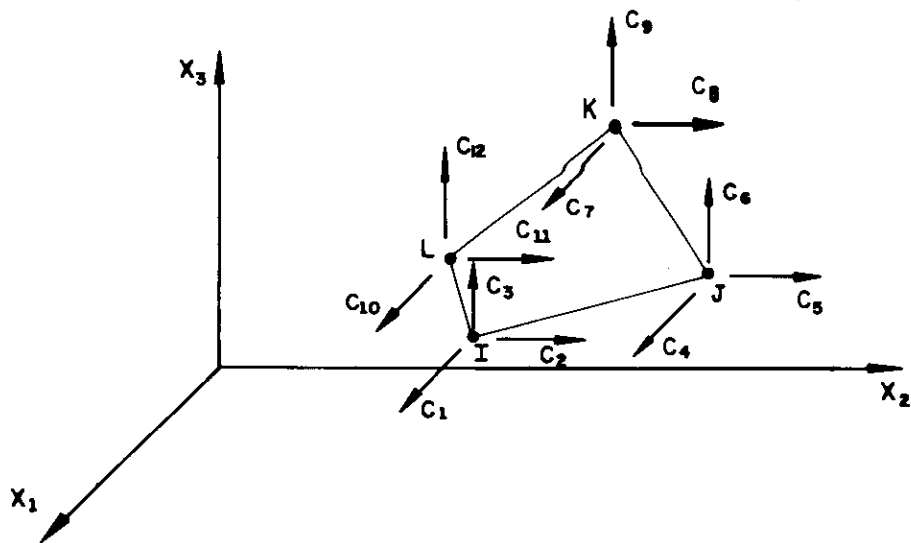


fig. 3.15 Coordenadas globais

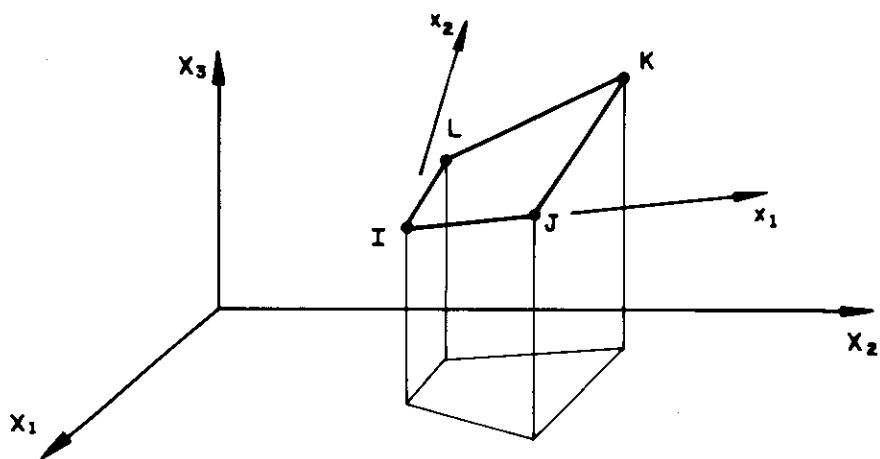


fig. 3.16 Sistemas local e global

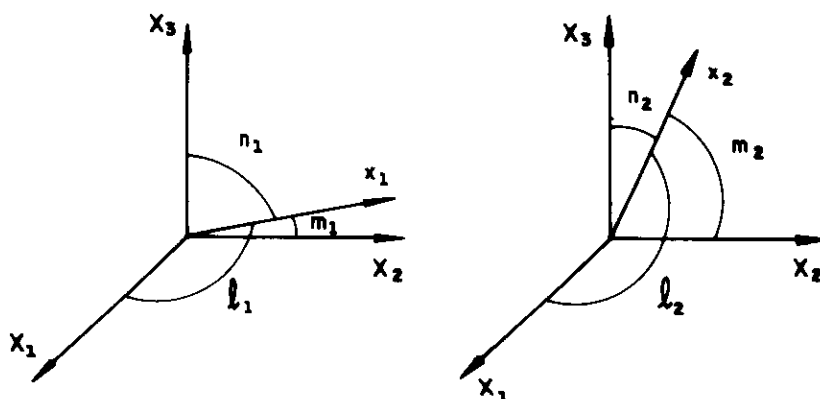


fig. 3.17 Cossenos diretores dos eixos locais

Também aqui será importante a definição de uma matriz que relacione os esforços e deslocamentos globais aos esforços e deslocamentos locais. Essa relação é realizada através de uma matriz de transformação  $\lambda$ , de oito linhas por doze colunas, segundo as equações 3.11. A matriz  $\lambda$  é montada com os cossenos diretores dos eixos locais  $x_1$  e  $x_2$  em relação aos eixos globais  $X_1$ ,  $X_2$  e  $X_3$ . A figura 3.16 mostra o comportamento geral e a 3.17 detalha esses cossenos e esclarece a nomenclatura adotada. Desse modo, a matriz  $\lambda$  pode ser expressa da maneira que se segue :

$$\lambda \sim = \begin{bmatrix} \lambda_0 & 0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & \lambda_0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & \lambda_0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & 0 & \lambda_0 \\ \sim & \sim & \sim & \sim \end{bmatrix} \quad \dots(3.51)$$

onde,

$$\lambda_0 \sim = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \end{bmatrix} \quad e \quad \theta \sim = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

### 3.6.3. Funções de Interpolação

O elemento adotado é isoparamétrico, ou seja, utiliza as mesmas funções para aproximar a geometria e os deslocamentos. Para a definição dessas funções foi adotado um sistema de dois eixos naturais, ou homogêneos, como referência auxiliar. Esses eixos aparecem na figura 3.18, juntamente com o sistema local de referência e a indicação de deslocamentos "u" e "v". A interpolação é feita em função dos valores nodais dos parâmetros. Assim sendo, pode-se escrever :

$$\begin{aligned}
 x(r,s) &= h_1 x^{(1)} + h_2 x^{(2)} + h_3 x^{(3)} + h_4 x^{(4)} \\
 u(r,s) &= h_1 u^{(1)} + h_2 u^{(2)} + h_3 u^{(3)} + h_4 u^{(4)}
 \end{aligned}
 \quad \dots(3.52)$$

onde,

$x^{(n)}$  : valor da coordenada  $x$  do nó " $n$ " (pode ser  $x_1$  ou  $x_2$ ),  
 $u^{(n)}$  : valor da deslocamento  $u$  do nó " $n$ " (pode ser  $u$  ou  $v$ ).

$$\begin{aligned}
 h_1 &= \frac{1}{4} (1-r)(1-s) \\
 h_2 &= \frac{1}{4} (1+r)(1-s) \\
 h_3 &= \frac{1}{4} (1+r)(1+s) \\
 h_4 &= \frac{1}{4} (1-r)(1+s)
 \end{aligned}$$

A relação entre as derivadas do sistema local e as derivadas do sistema homogêneo pode ser escrita como sendo:

$$\begin{bmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial s} \end{bmatrix} = \begin{bmatrix} \frac{\partial x_1}{\partial r} & \frac{\partial x_2}{\partial r} \\ \frac{\partial x_1}{\partial s} & \frac{\partial x_2}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \end{bmatrix} \quad \text{ou} \quad \frac{\partial}{\partial r} = J \frac{\partial}{\partial x} \quad \dots(3.53)$$

onde a matriz  $J$  é o jacobiano da transformação.

Em cálculos posteriores, devido à utilização de integração numérica, interessará o valor do jacobiano para um ponto específico de coordenadas  $r_i$  e  $s_j$ . Assim sendo, obter-se-á  $J_{ij}$ , cujos componentes serão :

$$\begin{aligned}
 \frac{\partial x_1}{\partial r} &= -\frac{1}{4}(1-s_j)x_1^{(1)} + \frac{1}{4}(1-s_j)x_1^{(2)} + \frac{1}{4}(1+s_j)x_1^{(3)} - \frac{1}{4}(1+s_j)x_1^{(4)} \\
 \frac{\partial x_1}{\partial s} &= -\frac{1}{4}(1-r_i)x_1^{(1)} - \frac{1}{4}(1+r_i)x_1^{(2)} + \frac{1}{4}(1+r_i)x_1^{(3)} + \frac{1}{4}(1-r_i)x_1^{(4)} \\
 \frac{\partial x_2}{\partial r} &= -\frac{1}{4}(1-s_j)x_z^{(1)} + \frac{1}{4}(1-s_j)x_z^{(2)} + \frac{1}{4}(1+s_j)x_z^{(3)} - \frac{1}{4}(1+s_j)x_z^{(4)} \\
 \frac{\partial x_2}{\partial s} &= -\frac{1}{4}(1-r_i)x_z^{(1)} - \frac{1}{4}(1+r_i)x_z^{(2)} + \frac{1}{4}(1+r_i)x_z^{(3)} + \frac{1}{4}(1-r_i)x_z^{(4)}
 \end{aligned}
 \quad \dots(3.54)$$

### 3.6.4. Matriz da Relação Deformações / Deslocamentos

Muito importante para o desenvolvimento da matriz de rigidez do elemento é a matriz que relaciona deformações com deslocamentos, ou seja :

$$\tilde{\epsilon} = B \tilde{u} \quad \dots(3.55)$$

Os componentes do vetor  $\tilde{\epsilon}$ , deformações do elementos em função do sistema de referência local, podem ser escritas como :

$$\begin{aligned} \epsilon_1 &= \frac{\partial u}{\partial x_1} \\ \epsilon_2 &= \frac{\partial v}{\partial x_2} \\ \gamma &= \frac{\partial u}{\partial x_2} + \frac{\partial v}{\partial x_1} \end{aligned} \quad \dots(3.56)$$

Assim sendo, considerando-se as funções de interpolação adotadas, tem-se para a matriz B :

$$B = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & 0 & \frac{\partial h_2}{\partial x_1} & 0 & \frac{\partial h_a}{\partial x_1} & 0 & \frac{\partial h_4}{\partial x_1} & 0 \\ 0 & \frac{\partial h_1}{\partial x_2} & 0 & \frac{\partial h_2}{\partial x_2} & 0 & \frac{\partial h_a}{\partial x_2} & 0 & \frac{\partial h_4}{\partial x_2} \\ \frac{\partial h_1}{\partial x_2} & \frac{\partial h_1}{\partial x_1} & \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \frac{\partial h_a}{\partial x_1} & \frac{\partial h_a}{\partial x_2} & \frac{\partial h_4}{\partial x_1} & \frac{\partial h_4}{\partial x_2} \end{bmatrix} \quad \dots(3.57)$$

As derivadas que aparecem na expressão 3.57 podem ser calculadas aos pares, para os pontos  $r_i$  e  $s_j$ . Suas expressões serão :

$$\begin{Bmatrix} \frac{\partial h_1}{\partial x_1} \\ \frac{\partial h_1}{\partial x_2} \end{Bmatrix} = \frac{1}{4} J_{ij}^{-1} \begin{Bmatrix} -(1-s_j) \\ -(1-r_i) \end{Bmatrix}$$

$$\begin{Bmatrix} \frac{\partial h_2}{\partial x_1} \\ \frac{\partial h_2}{\partial x_2} \end{Bmatrix} = \frac{1}{4} J_{ij}^{-1} \begin{Bmatrix} (1-s_j) \\ -(1+r_i) \end{Bmatrix}$$

...(3.58)

$$\begin{Bmatrix} \frac{\partial h_3}{\partial x_1} \\ \frac{\partial h_3}{\partial x_2} \end{Bmatrix} = \frac{1}{4} J_{ij}^{-1} \begin{Bmatrix} (1+s_j) \\ (1+r_i) \end{Bmatrix}$$

$$\begin{Bmatrix} \frac{\partial h_4}{\partial x_1} \\ \frac{\partial h_4}{\partial x_2} \end{Bmatrix} = \frac{1}{4} J_{ij}^{-1} \begin{Bmatrix} -(1+s_j) \\ (1-r_i) \end{Bmatrix}$$

### 3.6.5. Matriz de Constantes Elásticas

Considere-se a relação entre as tensões e as deformações :

$$\tilde{\sigma} = C \tilde{\epsilon} \quad \dots(3.59)$$

A matriz C é formada por constantes elásticas que dizem respeito ao material utilizado e também ao tipo de estado de tensão que se verifica. No presente caso, material isotrópico e estado plano de tensão, a matriz C resulta da seguinte forma :

$$C = \begin{bmatrix} E/(1-\nu^2) & E\nu/(1-\nu^2) & 0 \\ E\nu/(1-\nu^2) & E/(1-\nu^2) & 0 \\ 0 & 0 & G \end{bmatrix} \quad \dots(3.60)$$

### 2.6.6. Matriz de Rigidez em Coordenadas Globais

A matriz de rigidez do elemento, ainda em relação aos eixos locais, pode ser calculada pela expressão (ver item 3.2) :

$$\tilde{r} = \int_{\tilde{V}} \tilde{B}^T \tilde{C} \tilde{B} dV \quad \dots(3.61)$$

onde,

$\tilde{B}$  : matriz da relação deformações / deslocamentos.

$\tilde{C}$  : matriz de constantes elásticas do material.

Mas, considerando-se a espessura constante "t", pode-se escrever :

$$dV = |\det J| t dr ds \quad \dots(3.62)$$

Portanto, a equação 3.61 resulta :

$$\tilde{r} = \int_A \tilde{B}^T \tilde{C} \tilde{B} |\det J| t dr ds \quad \dots(3.63)$$

Considerando-se a integração numérica pelo processo de Gauss, tem-se :

$$\tilde{r} = \sum_{i,j} t \omega_{ij} R_{ij} \quad \dots(3.64)$$

onde,

t : espessura do elemento.

$\omega_{ij}$  : fator de peso dado pelo processo de Gauss.

Já a matriz  $R_{ij}$  tem como expressão :

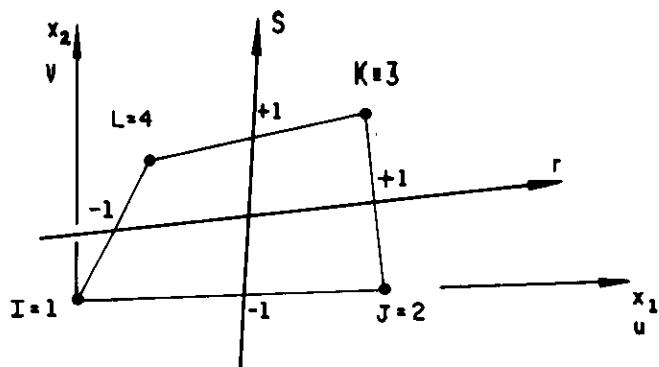


fig. 3.18 Sistema auxiliar de coordenadas homogêneas

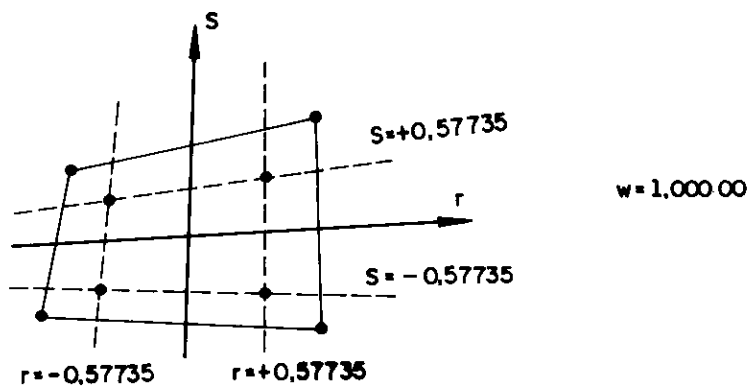


fig. 3.19 Pontos de Gauss para integração numérica

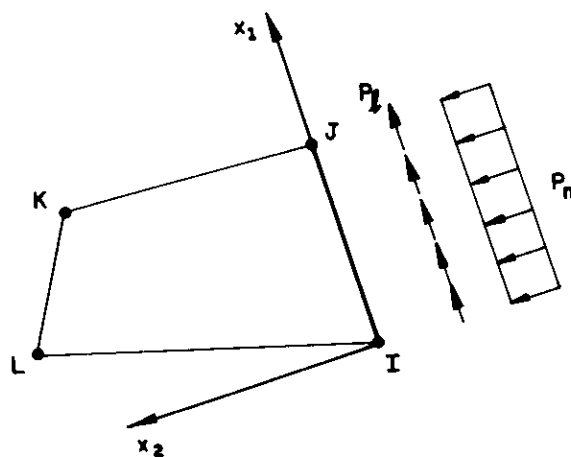


fig. 3.20 Forças de superfície sobre a face IJ

$$\tilde{R}_{ij} = \tilde{B}_{ij} C \tilde{B}_{ij} |\det \tilde{J}_{ij}| \quad \dots(3.65)$$

onde,

$\tilde{B}_{ij}$  : matriz da relação deslocamentos / deformações calculada no ponto de Gauss de coordenada  $r_i$  e  $s_j$ .

$\tilde{J}_{ij}$  : jacobiano calculado para o ponto  $r_i$  e  $s_j$ .

A integração numérica é feita utilizando-se quatro pontos, suficientes para produzir o resultado exato se o elemento tiver a forma de um paralelogramo. Esses pontos estão mostrados na figura 3.19, inclusive com o valor de  $\omega$  a ser considerado.

A matriz de rigidez do elemento, em relação aos eixos de referência globais, é calculada em função da matriz local e da matriz de transformação  $\lambda$ , conforme a equação 3.17. No caso tem-se :

$$\tilde{R}E = \lambda^T r \lambda \quad \dots(3.66)$$

### 3.6.7. Vetor de Cargas

São permitidos três tipos de carregamento sobre o elemento : força de superfície atuante na face "IJ", variação de temperatura e ainda peso próprio. Os vetores de carga, um para cada carregamento, são portanto formados pela soma de três parcelas.

Quanto ao carregamento sobre a face, somente podem ser definidos valores constantes, na direção normal e na sua própria direção, cujos sentidos positivos são mostrados na figura 3.20. Em função das coordenadas locais, o vetor de cargas devido às tensões sobre a face "IJ" pode ser encontrado pela equação (ver item 3.2) :



$$\tilde{p}_1 = \int_S \tilde{H}_s^T \tilde{f}_s dS \quad \dots(3.67)$$

Na equação 3.67, a matriz  $H_s$  relaciona os deslocamentos nos nós com os de um determinado ponto da face "IJ". Ou seja :

$$\tilde{u}_s = \tilde{H}_s \tilde{u} \quad \dots(3.68)$$

A matriz  $H_s$  é montada com as funções de interpolação. Considerando-se que na face "IJ" tem-se sempre  $s = -1$ , pode-se escrever :

$$\tilde{H}_s = \begin{bmatrix} (1-r) & 0 & (1+r) & 0 & 0 & 0 & 0 & 0 \\ 0 & (1-r) & 0 & (1+r) & 0 & 0 & 0 & 0 \end{bmatrix} \dots(3.69)$$

Considerando-se ainda a equação 3.67, deve-se lembrar que :

$$dS = t dl = t |\det J_s| dr \quad \dots(3.70)$$

onde,

t : espessura do elemento.

$$\det J_s = \sqrt{\left(\frac{\partial x_1}{\partial r}\right)^2 + \left(\frac{\partial x_2}{\partial r}\right)^2} = \frac{1}{2}$$

Assim, sendo "l" o comprimento do lado "IJ", pode-se escrever :

$$dS = \frac{t l}{2} \quad \dots(3.71)$$

Já o vetor  $\tilde{f}_s$  pode ser escrito como :

$$\underset{\sim}{f}_s = \begin{Bmatrix} p_n \\ p_1 \end{Bmatrix} \quad \dots(3.72)$$

Portanto a equação 3.67 pode ser reescrita na forma :

$$\underset{\sim}{p}_1 = \int_{-1}^{+1} \underset{\sim}{H}^T \underset{\sim}{f}_s \frac{t \ 1}{2} dr \quad \dots(3.73)$$

Logo, o vetor  $p_1$  resulta :

$$\underset{\sim}{p}_1 = \frac{t \ 1}{2} \begin{Bmatrix} p_1 \\ p_n \\ p_1 \\ p_n \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \dots(3.74)$$

Para calcular o vetor em relação às coordenadas globais, basta fazer :

$$\underset{\sim}{P}_1 = \underset{\sim}{\lambda}^T \underset{\sim}{p}_1 \quad \dots(3.75)$$

Para a consideração do efeito de uma variação de temperatura sobre o elemento, tem-se (ver item 3.2) :

$$\underset{\sim}{p}_2 = - \int_V \underset{\sim}{B}^T \underset{\sim}{\sigma}_I dV = - \int_V \underset{\sim}{B}^T \underset{\sim}{C} \underset{\sim}{\epsilon}_I dV \quad \dots(3.76)$$

Já o vetor de deformações iniciais devidas a um variação de temperatura  $\Delta T$  será o seguinte :

$$\tilde{\epsilon}_I = \alpha \Delta T \begin{Bmatrix} 1 \\ 1 \\ 0 \end{Bmatrix} \quad \dots(3.77)$$

Quanto à equação 3.76, deve-se lembrar que :

$$dV = |\det \tilde{J}| t dr ds \quad \dots(3.78)$$

Ou seja :

$$p_2 = - \int_A \tilde{B}^T \tilde{C} \tilde{\epsilon}_I |\det \tilde{J}| t dr ds \quad \dots(3.79)$$

Ou ainda, considerando-se integração numérica :

$$p_2 = - \sum_{i,j} t \omega_{ij} T_{ij} \quad \dots(3.80)$$

onde,

t : espessura do elemento.

$\omega_{ij}$  : fator de peso dado pelas expressões de Gauss.

A matriz  $T_{ij}$  tem como expressão :

$$T_{ij} = \tilde{B}_{ij}^T \tilde{C} \tilde{\epsilon}_I |\det \tilde{J}_{ij}| \quad \dots(3.81)$$

Para a obtenção do vetor de cargas, em relação às coordenadas globais, basta fazer :

$$P_2 = \lambda^T p_2 \quad \dots(3.82)$$

Como última contribuição existe o peso próprio. De acordo com o que foi feito em outros elementos, o peso próprio não será calculado como carga consistente, mas

simplesmente colocado segundo as direções globais de acordo com frações fornecidas pelo usuário. Desse modo, é necessário o cálculo do volume (área x espessura) associado a cada um dos quatro nós. Esse volume, de  $V_1$  a  $V_4$ , pode ser determinado pela expressão :

$$V_k = \sum_{i,j} t w_{ij} |\det J_{ij}| h_k^{(i,j)} \quad \dots(3.83)$$

onde,

$h_k^{(i,j)}$  : representa a função interpoladora  $h_k$  calculada para o ponto  $r_i$  e  $s_j$ .

Desse modo o vetor de contribuição do peso próprio, já em coordenadas globais resulta :

$$P_{\sim 3} = \gamma \begin{pmatrix} V_1 a_1 \\ V_1 a_2 \\ V_1 a_3 \\ V_2 a_1 \\ V_2 a_2 \\ V_2 a_3 \\ V_3 a_1 \\ V_3 a_2 \\ V_3 a_3 \\ V_4 a_1 \\ V_4 a_2 \\ V_4 a_3 \end{pmatrix} \quad \dots(3.84)$$

onde,

$a_1, a_2$  e  $a_3$  : são as frações do peso próprio a serem consideradas segundo os eixos  $X_1, X_2$  e  $X_3$ .

Portanto, para o vetor de cargas do elemento obtém-se:

$$PE_{\sim} = P_{\sim 1} + P_{\sim 2} + P_{\sim 3} \quad \dots(3.85)$$

### 3.6.8. Matriz da Relação Tensão / Deslocamento

A matriz que relaciona as tensões com os deslocamento é montada em função dos pontos do elemento nos quais os resultados serão apresentados. Considerando-se as equações 3.11, 3.55 e 3.59 é muito simples concluir que :

$$\underset{\sim}{\sigma} = \underset{\sim}{C} \underset{\sim}{B} \underset{\sim}{\lambda} \underset{\sim}{U} \quad \dots(3.86)$$

Imaginando-se que as tensões sejam calculadas num ponto de coordenadas  $r_i$  e  $s_j$ , tem-se :

$$\underset{\sim}{\sigma}_{ij} = \underset{\sim}{C} \underset{\sim}{B}_{ij} \underset{\sim}{\lambda} \underset{\sim}{U} \quad \dots(3.87)$$

No caso do programa LS, o usuário pode optar por até 5 pontos onde as tensões serão apresentadas. Assim sendo, cada vez que a equação 3.87 for aplicada para um determinado ponto escolhido, estarão sendo montadas três linhas da matriz. Portanto, a matriz TD tem dimensão variável, podendo apresentar de três a quinze linhas, sempre com doze colunas.

Como exemplo, se os resultados das tensões forem apresentados em apenas um ponto, de coordenadas  $r_i$  e  $s_j$ , pode-se escrever a matriz TD da seguinte forma :

$$\underset{\sim}{TD} = \underset{\sim}{C} \underset{\sim}{B}_{ij} \underset{\sim}{\lambda} \quad \dots(3.88)$$

### 3.6.9. Vetor de Tensões Iniciais

O vetor de tensões iniciais contém apenas o efeito da variação da temperatura. Tanto o carregamento na face como o peso próprio não contribuem para sua formação. Desse

modo, pode-se escrever :

$$\tilde{\sigma}_I = - \tilde{C} \tilde{\epsilon}_I \quad \dots(3.89)$$

onde,

$\tilde{\epsilon}_I$  : deformações iniciais, fornecidas pela expressão 3.77.

Como pode-se perceber, essas tensões iniciais são constantes para qualquer ponto do elemento. Apenas a ressaltar que para cada caso de carregamento da estrutura deve existir um vetor de tensões iniciais  $\tilde{T}_I$ .

De modo semelhante à matriz  $\tilde{T}_D$ , para cada ponto onde os resultados sejam apresentados é necessário montar três posições no vetor  $\tilde{T}_I$ . Assim sendo, para um elemento com os resultados sendo calculados num único ponto, o vetor  $\tilde{T}_I$  seria :

$$\tilde{T}_I = - \tilde{C} \tilde{\epsilon}_I \quad \dots(3.90)$$

### 3.6.10. Modos Incompatíveis

Foi mencionado no item 3.6.1 que o elemento aqui apresentado permite a incorporação de modos incompatíveis. Entretanto, em todo o desenvolvimento das matrizes e vetores calculados foi propositalmente esquecido esse recurso. Agora, se fará uma explanação geral a respeito desses modos, indicando os pontos a serem modificados pela sua utilização.

O conceito da implementação é bastante simples. Inicialmente toma-se a segunda equação 3.52, aproximação dos deslocamentos, e adicionam-se a ela mais quatro coeficientes multiplicados por termos de ordem mais elevada da função interpoladora. Desse modo, a aproximação dos

deslocamentos será feita pelas seguintes equações :

$$u = h_1 u^{(1)} + h_2 u^{(2)} + h_3 u^{(3)} + h_4 u^{(4)} + \alpha_1 (1-r^2) + \alpha_2 (1-s^2)$$

$$v = h_1 v^{(1)} + h_2 v^{(2)} + h_3 v^{(3)} + h_4 v^{(4)} + \beta_1 (1-r^2) + \beta_2 (1-s^2)$$

...(3.91)

Com a utilização das funções aproximadoras 3.91, obtém-se a matriz B com três linhas e doze colunas, ao invés das oito originais. De acordo com as equações deduzidas, as matrizes e vetores que dependem da matriz B para serem calculados terão suas dimensões aumentadas. São os especificados a seguir : matriz de rigidez, "r"; matriz da relação tensões / deslocamentos, "TD" ; e a parcela do vetor de cargas referente à atuação de uma variação de temperatura, "p<sub>2</sub>".

Entretanto, somente existem oito graus de liberdade verdadeiros. Os quatro restantes precisam ser então eliminados através de um procedimento de condensação estática [3.11], voltando as matrizes e vetores a apresentarem a dimensão original. Esse processo de condensação nada mais é do que uma triangularização da variável considerada operando sobre as linhas correspondentes aos graus de liberdade a serem condensados. Desse modo, nas posições restantes ficam os graus de liberdade verdadeiros, já com a adição dos efeitos desejados.

Com o procedimento descrito, elementos retangulares podem representar com mais precisão casos onde exista uma solicitação típica de momento fletor, melhorando-se sobremaneira a convergência dos resultados. Entretanto, a adição desses graus de liberdade, não relacionados às coordenadas nodais, conduz à obtenção de deslocamentos incompatíveis entre os elementos. Desse modo, não existe garantia de convergência monotônica, ou mesmo garantia de

convergência.

Apesar disso, toda a experiência de utilização desses modos incompatíveis tem mostrado que as vantagens desse procedimento, quando corretamente utilizado, são indubitáveis. Para uma discussão a níveis mais práticos vai-se desenvolver, no item correspondente, um exemplo com e sem a utilização desses recursos, permitindo-se algumas conclusões interessantes.

### 3.6.11. Elementos de Forma Triangular

O fato de se ter um elemento de formulação isoparamétrica permite que, teóricamente, ele seja degenerado, formando triângulos pela simples providência de se atribuir a dois extremos do elemento o mesmo ponto nodal da estrutura. Assim sendo, pode-se fornecer, por exemplo, as mesmas coordenadas para os nós "I" e "J" de um determinado elemento, obtendo-se então a forma mostrada na figura 3.21a.

Entretanto, no programa LS, essa degeneração somente pode ocorrer pelo fornecimento de um mesmo ponto nodal da estrutura para os extremos "L" e "K", ou seja, para os dois últimos nós do elemento, conforme figura 3.21b. Isso se verifica por procedimentos internos do programa, não sendo uma restrição relativa à concepção do elemento. É simples perceber que seria impossível, por exemplo, degenerar o elemento pela junção dos nós "I" e "J". Nesse caso, o próprio sistema local adotado ficaria sem definição.

É interessante ressaltar que, no caso de um elemento definido por apenas três pontos nodais, a formulação obtida é equivalente àquela encontrada por um desenvolvimento direto considerando-se um campo linear de deslocamentos. Esse elemento, conhecido na literatura como CST ("Constant Strain Triangle"), constitui-se no mais simples elemento



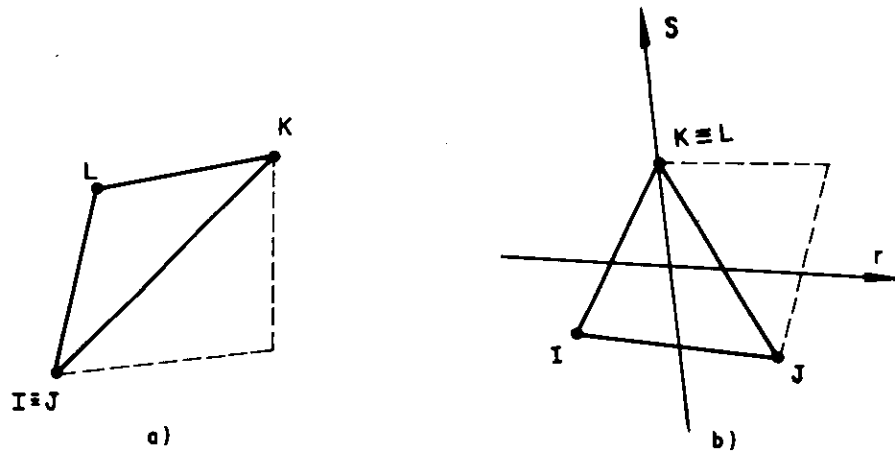


fig. 3.21 Elementos triangulares

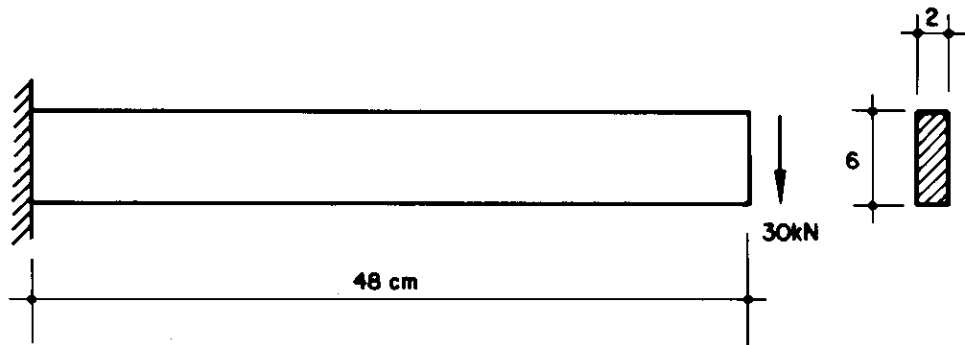


fig. 3.22 Viga em balanço

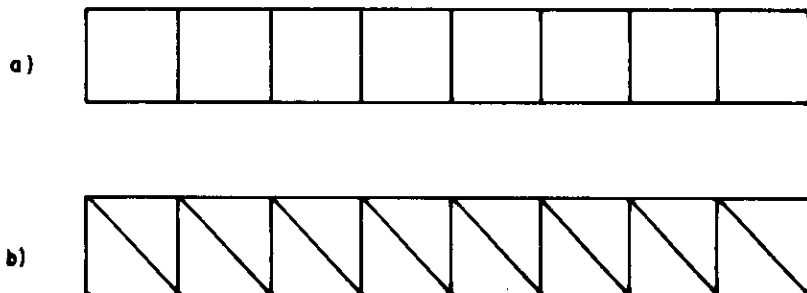


fig. 3.23 Discretizações 1 e 4

isoparamétrico que pode ser desenvolvido para modelar estados planos de tensão.

### 3.6.12. Desempenho

#### 3.6.12.1. Convergência de Resultados

Para examinar a convergência do elemento membrana aqui desenvolvido, será considerado o problema de uma viga em balanço mostrada na figura 3.22. Como dados adicionais pode-se citar :

$$E = 20\ 000\ \text{KN/cm}^2$$

$$\nu = 0.25$$

$$G = 8\ 000\ \text{KN/cm}^2$$

O acompanhamento dos resultados obtidos será realizado para seis diferentes discretizações, cujas redes são definidas da maneira que se segue :

Discretização 1 : 8 elementos retangulares ( fig 3.23a ).

Discretização 2 : 32 elementos retangulares ( fig 3.24a ).

Discretização 3 : 128 elementos retangulares ( fig 3.25a ).

Discretização 4 : 16 elementos triangulares ( fig 3.23b ).

Discretização 5 : 64 elementos triangulares ( fig 3.24b ).

Discretização 6 : 256 elementos triangulares ( fig 3.25b ).

Além disso, tanto para os elementos retangulares como para os triangulares, a análise foi realizada com e sem a inclusão dos modos incompatíveis. Desse modo, espera-se um diagnóstico relativamente completo sobre as potencialidades do elemento mostrado.

Tratando-se o problema em questão de uma viga, com

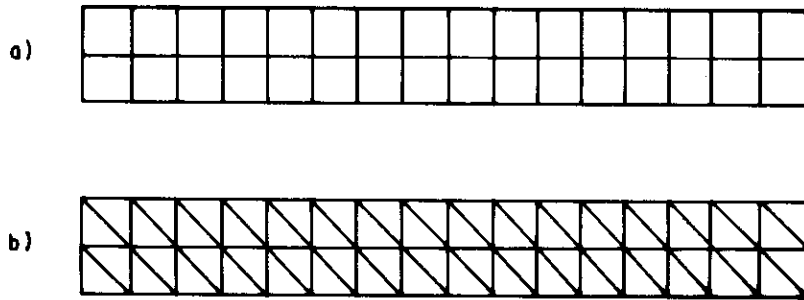


fig. 3.24 Discretizações 2 e 5

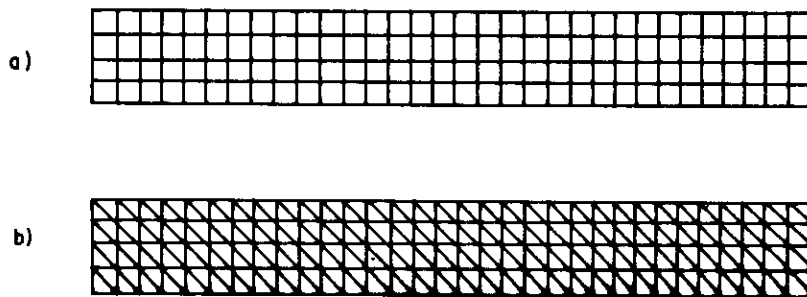


fig. 3.25 Discretizações 3 e 6

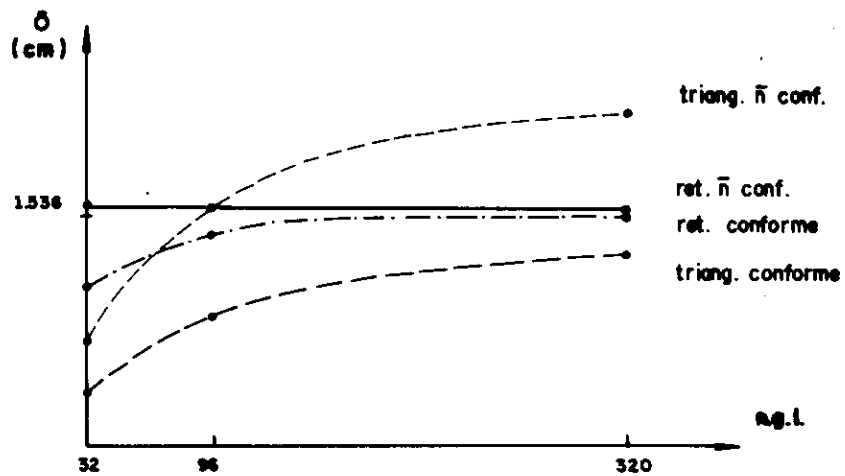


fig. 3.26 Convergência dos resultados

relação altura sobre vão de  $1/8$ , serão controlados os valores do deslocamento na extremidade livre, da tensão normal na extremidade superior de uma seção próxima ao engaste e ainda da tensão tangencial no eixo da peça. Quanto ao deslocamento da extremidade livre, os resultados obtidos encontram-se anotados em cm na tabela 3.3. Já os valores das tensões normais, em  $\text{KN/cm}^2$ , são apresentados pela tabela 3.4. Ressalta-se que nesse caso o ponto de comparação dos valores varia em relação às discretizações consideradas. Isso se deve ao fato de ser necessário fugir-se das perturbações causadas pelo engaste. Entretanto, para cada caso considerado, é apresentada a respectiva tensão normal obtida pela teoria de viga. Finalmente a tensão tangencial, constante ao longo do eixo longitudinal, é apresentada em  $\text{KN/cm}^2$  na tabela 3.5.

Pela observação dos resultados pode-se concluir algumas questões interessantes. Em primeiro lugar, os elementos com inclusão de modos incompatíveis, "ñcomp" nas referidas tabelas, apresentam comportamento completamente diverso. Os retangulares apresentam uma convergência espetacular e mesmo com a rede mais pobre já encontram-se muito próximos da solução exata. Já os triangulares divergem, conforme pode-se perceber pela figura 3.26 que apresenta o gráfico dos deslocamentos na extremidade livre para todas as análises. Quando trata-se dos elementos sem a inclusão dos modos incompatíveis, "comp" nas tabelas, os dois tipos acabam convergindo, se bem que os retangulares o façam de uma maneira mais acentuada.

Conclui-se, portanto, que como regra geral deve-se preferencialmente utilizar os elementos quadrangulares. Se isso não for possível, os elementos triangulares podem ser utilizados, entretanto sem a inclusão dos modos incompatíveis.

Deslocamento na extremidade livre						
Disc.	1	2	3	4	5	6
$\tilde{n}$ comp	1.54484	1.54518	1.54520	0.69213	1.54533	2.21345
comp	1.05826	1.38605	1.50798	0.36370	0.84233	1.27668
viga	1.53600					

tabela 3.3

Tensão Normal na Extremidade Superior da Seção						
Disc.	1	2	3	4	5	6
$\tilde{n}$ comp	97.49	101.20	106.80	43.58	98.35	145.60
comp	70.91	93.71	105.80	23.64	56.39	89.13
viga	97.50	101.25	106.88	97.50	101.25	106.88

tabela 3.4

Tensão Tangencial no Eixo da Viga						
Disc.	1	2	3	4	5	6
$\tilde{n}$ comp	2.500	2.500	3.430	-	7.839	4.693
comp	2.500	2.564	3.467	-	8.280	5.411
viga	3.750					

tabela 3.5

Um último comentário pode ser feito em relação ao valor da tensão tangencial. Pode-se perceber que apenas com uma discretização um pouco melhor segundo a altura da peça é que se pode obter bons resultados. Quando são utilizados

apenas um ou dois elementos segundo essa direção, os resultados afastam-se bastante do valor obtido com a teoria de vigas.

### 3.6.12.2. Tempo de Processamento

Quanto ao desempenho em relação aos tempos de processamento, o elemento tipo 4 comporta-se muito bem. Podem ocorrer algumas pequenas variações dependendo da inclusão ou não de modos incompatíveis e do número de tensões a serem calculadas como resultado. Entretanto, como valor médio admite-se 0.40 seg para a montagem e gravação das matrizes e vetores e 0.11 seg para o cálculo das tensões resultantes.

### 3.6.13. Exemplo

O exemplo aqui discutido é o apresentado na figura 3.27. Trata-se de duas paredes de alvenaria interceptadas em ângulo reto e submetidas a carregamentos diferentes, situação muito comum em edifícios de alvenaria auto-portante.

Nesse caso, costumam aparecer tensões de tração, nas imediações da ligação, que provocam fissuras nas paredes. Neste exemplo, vai-se mostrar como essas fissuras poderiam ser perfeitamente previstas com a utilização de uma análise através de elementos finitos.

Os dados adicionais para o material a ser considerado são :

$$E = 3\ 000\ \text{MPa}$$

$$\nu = 0.20$$

$$G = 1\ 250\ \text{MPa}$$

Quanto ao carregamento, foram adotados para  $p_1$  e  $p_2$ , respectivamente cargas sobre as paredes 1 e 2, os seguintes valores :

$$p_1 = 30 \text{ KN/m}$$

$$p_2 = 10 \text{ KN/m}$$

A discretização da estrutura foi realizada com elementos retangulares do tipo 4 de  $0.5 \times 0.5$  m considerando-se a inclusão dos modos incompatíveis. No total foram necessários 217 nós e 180 elementos, sendo a rede obtida mostrada em perspectiva na figura 3.28. A aplicação do carregamento foi feita nos pontos nodais superiores por meio de cargas concentradas.

Os resultados obtidos são apresentados através de curvas de isotensão, para cada uma das paredes. No caso, serão apresentados apenas dois tipos de tensão, em  $\text{KN/cm}^2$ , para cada uma das paredes. É a seguinte a relação das figuras e o que apresentam :

Figura 3.29 : máximas tensões de tração na parede 1.

Figura 3.30 : tensões de cisalhamento na parede 1.

Figura 3.31 : máximas tensões de tração na parede 2.

Figura 3.32 : tensões de cisalhamento na parede 2.

Pela observação dos resultados apresentados, pode-se perceber que existe uma grande interação na ligação das paredes. Além disso, as fissuras por tração devem ocorrer não exatamente na ligação, como seria o esperado, mas a aproximadamente a 1 m de distância, na parede 2. Pois essa é exatamente a situação verificada na prática, revelando a correção da análise realizada.

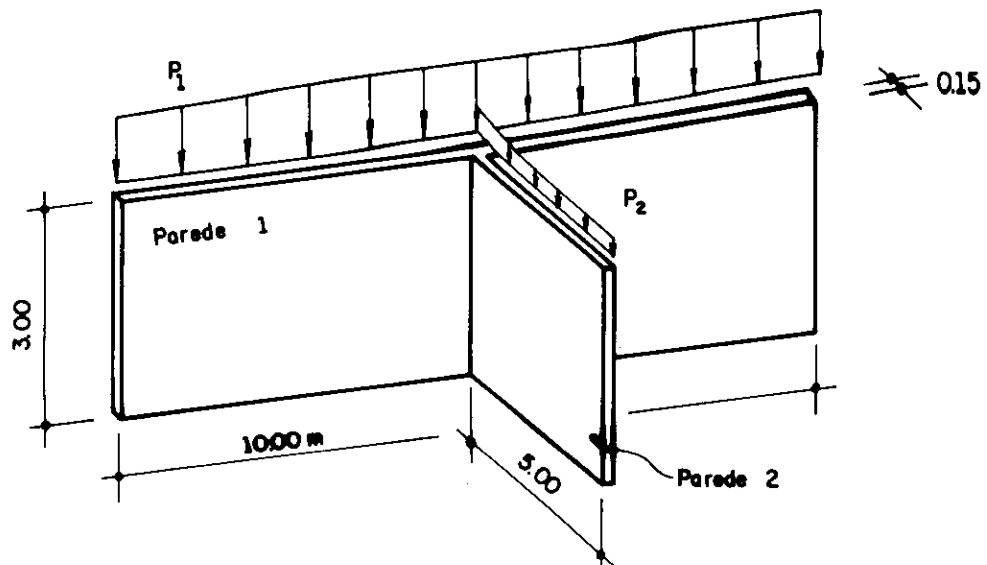


fig. 3.27 Paredes Interceptadas

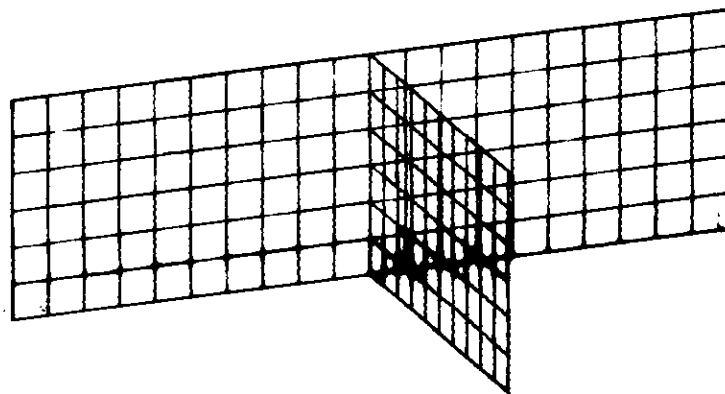


fig. 3.28 Discretização das paredes

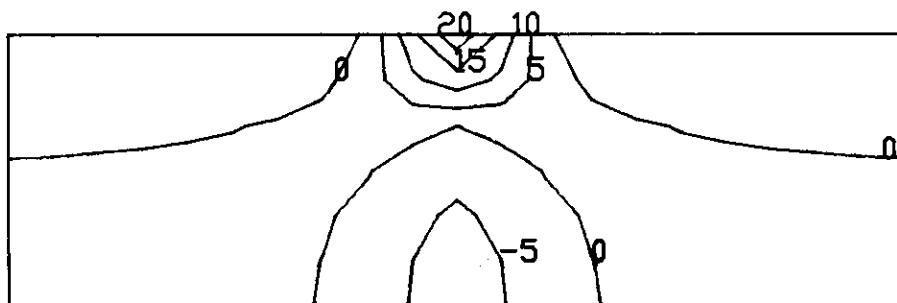


fig. 3.29 Tensão normal máxima na parede 1



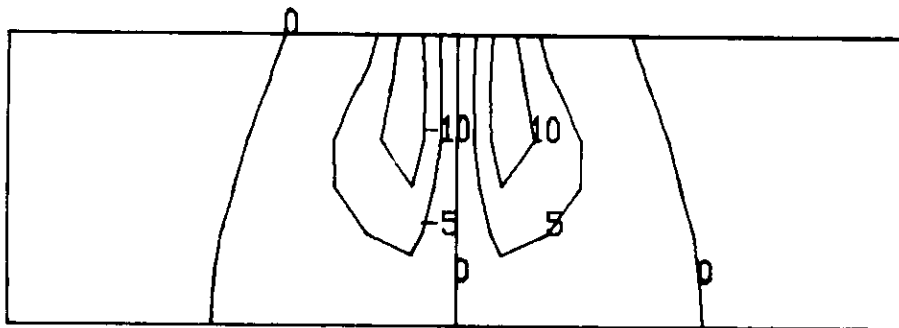


fig. 3.30 Tensão de cisalhamento na parede 1

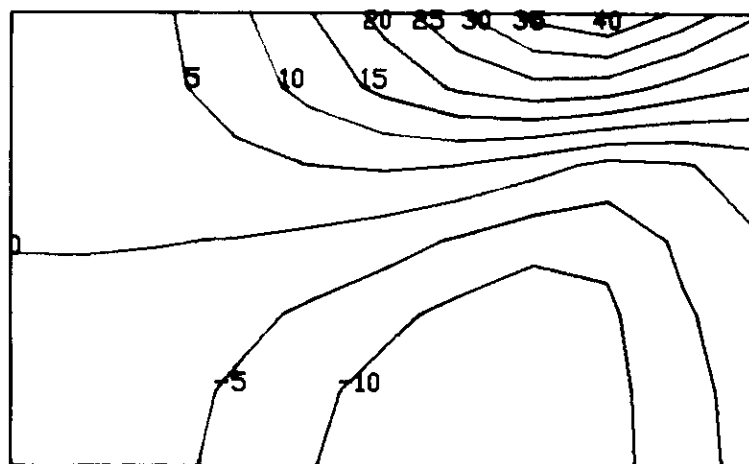


fig. 3.31 Tensão normal máxima na parede 2

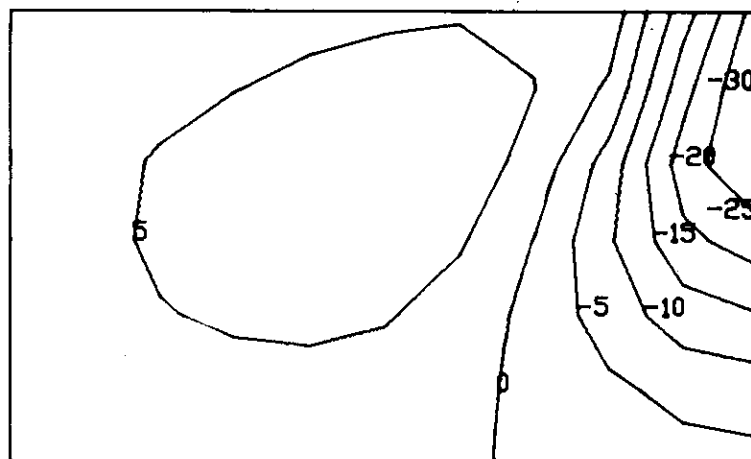


fig. 3.32 Tensão de cisalhamento na parede 2

### 3.7. ELEMENTO TIPO 5 - PLANO

#### 3.7.1. Caracterização do Elemento

O elemento plano é usado para a análise de estados planos de tensão, deformação e axissimétrico. Pode ser um quadrilátero ou um triângulo, sendo entretanto obrigatoriamente definido no plano principal  $X_2X_3$ , conforme mostra-se na figura 3.33.

O elemento tem apenas dois graus de liberdade, que são as translações de seus pontos nodais no referido plano  $X_2X_3$ . Seu sistema local de referência é composto por três eixos,  $x_1$ ,  $x_2$  e  $x_3$ , apresentados na figura 3.34. Ressalta-se que esses eixos locais são paralelos aos eixos globais da estrutura.

O elemento permite a consideração de material ortotrópico. Nesse caso, é necessário que os eixos de simetria do material coincidam com os eixos locais dos elementos. Isso significa também serem paralelos os eixos globais  $X_1$ ,  $X_2$  e  $X_3$ . Os parâmetros a serem fornecidos para os materiais dos elementos são :

- $\gamma$  = peso específico do material (opcional)
- $E_1$  = módulo de elasticidade longitudinal segundo  $x_1$ .
- $E_2$  = módulo de elasticidade longitudinal segundo  $x_2$ .
- $E_3$  = módulo de elasticidade longitudinal segundo  $x_3$ .
- $\nu_{12}$  = coeficiente de Poisson no plano  $x_1x_2$ .
- $\nu_{13}$  = coeficiente de Poisson no plano  $x_1x_3$ .
- $\nu_{23}$  = coeficiente de Poisson no plano  $x_2x_3$ .
- $G_{12}$  = módulo de elasticidade transversal no plano  $x_1x_2$ .
- $\alpha_1$  = coeficiente de dilatação térmica segundo  $x_1$  (opc).
- $\alpha_2$  = coeficiente de dilatação térmica segundo  $x_2$  (opc)
- $\alpha_3$  = coeficiente de dilatação térmica segundo  $x_3$  (opc)

É importante mencionar que o elemento aqui analisado e aquele apresentado no item anterior, com a denominação de membrana, são muito semelhantes. Aliás, o bloco de rotinas que realiza as duas montagens é o mesmo. As diferenças mais significativas dizem respeito exatamente aos tópicos já mencionados, ou seja, capacidade de análise de estados planos de deformação e axissimétrico, obrigatoriedade de estar contido no plano principal  $X_2X_3$  e possibilidade de uso de material ortotrópico.

De resto, as outras características são comuns para os dois tipos. Pode-se mencionar a possibilidade de inclusão de quatro modos incompatíveis de deslocamentos, forças de superfície constantes segundo os eixos locais na face IJ, variação de temperatura e ainda peso próprio, agora apenas segundo os eixos globais  $X_2$  e  $X_3$ .

Alguns outros pequenos detalhes de modelagem estão relacionados aos estados plano de deformação e axissimétrico. No caso do estado plano de deformação, é importante mencionar que o programa sempre vai assumir a espessura do elemento como unitária, já que qualquer valor para esse parâmetro não teria maior significado. No caso do estado axissimétrico, o programa assume que os elementos definidos representam uma parcela de um radiano da estrutura total, sendo  $X_3$  o eixo de simetria da mesma. Desse modo, a espessura do elemento é variável e calculada em relação à distância do ponto considerado ao referido eixo  $X_3$ , conforme mostra-se na figura 3.35.

Com respeito às utilizações principais deste elemento, pode-se mencionar, além dos casos já citados para o elemento anterior e desde que totalmente contidos no plano  $X_2X_3$ , as estruturas de barragens, vasos de pressão, túneis, etc.

Conforme foi mencionado, é muito grande a semelhança deste elemento com o apresentado no item anterior. Para evitar-se a repetição de conceitos, o elemento plano será discutido com maiores detalhes apenas nos pontos onde

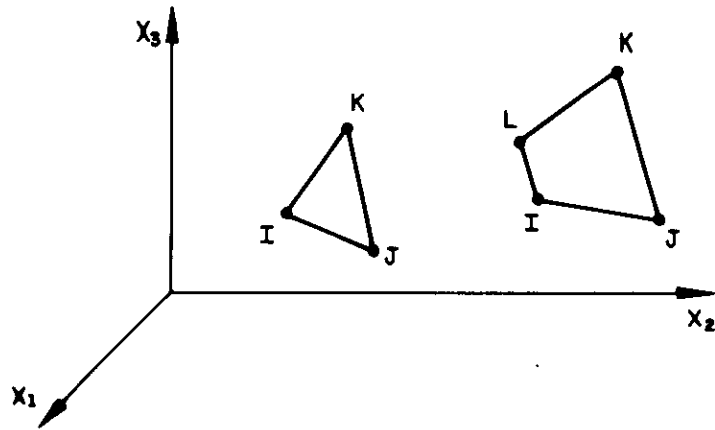


fig. 3.33 Elementos planos de 3 e 4 nós

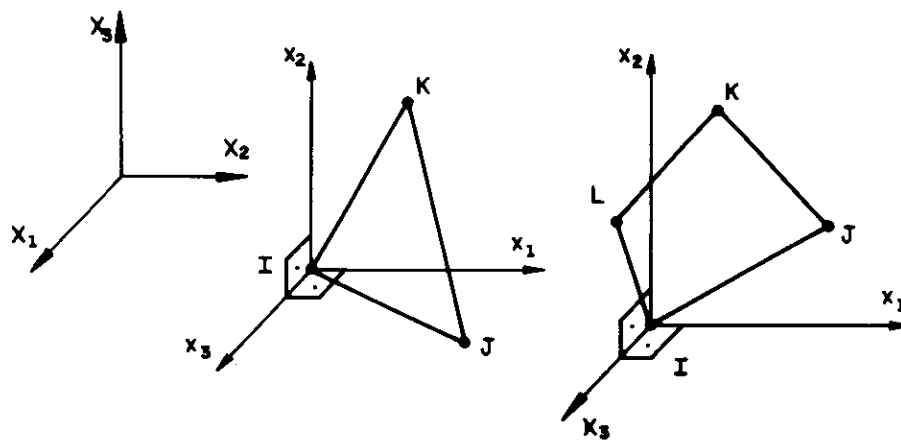


fig. 3.34 Sistemas de referência local

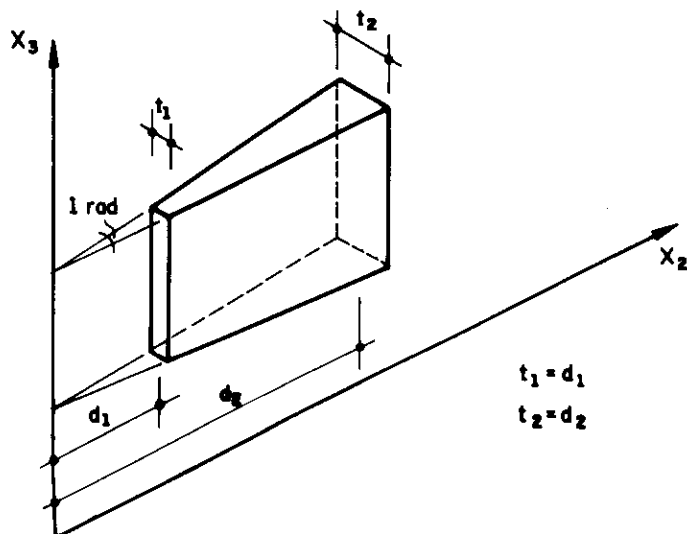


fig. 3.35 Espessuras para elemento axissimétrico

existam diferenças significativas em relação ao anterior. Nas passagens onde essas diferenças não se fizerem presentes, vai-se apenas fazer referência ao que já foi apresentado. Até mesmo em termos de bibliografia a ser referenciada, os trabalhos são exatamente os mesmos do caso anterior.

### 3.7.2. Coordenadas Locais e Globais

As coordenadas locais deste elemento são semelhantes às do caso anterior, ou seja, duas para cada nó. São as apresentadas na figura 3.36.

Quanto às coordenadas globais, neste caso elas serão as mesmas do sistema local. Isso ocorre pois os sistemas de referência são paralelos. Além disso, devido ao fato do elemento desenvolver-se no plano global  $X_2X_3$ , não há necessidade da coordenada segundo o eixo global  $X_1$ .

Desse modo, a matriz de transformação de coordenadas do sistema local para o global é a identidade. Todos os vetores e matrizes calculados em relação ao sistema local podem ser considerados em relação ao sistema global sem qualquer alteração.

### 3.7.3. Funções de Interpolação

Todas as expressões deste sub-item são idênticas às correspondentes do elemento tipo 4.

### 3.7.4. Matriz da Relação Deformações / Deslocamentos

Para a determinação da matriz que relaciona as deformações com os deslocamentos, são necessárias algumas alterações, em relação ao elemento anterior, quando se tratar de análise de um estado axissimétrico. Nesse caso, além dos valores  $\epsilon_1$ ,  $\epsilon_2$  e  $\gamma$  presentes nos estados planos de tensão e deformação, existe ainda a deformação normal ao plano do elemento,  $\epsilon_a$ . Assim sendo, a matriz B será composta pelas três linhas apresentadas na expressão 3.57 e mais uma quarta linha que pode ser determinada por :

$$B = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{h_1^{(i,j)}}{t_{ij}} & 0 & \frac{h_2^{(i,j)}}{t_{ij}} & 0 & \frac{h_3^{(i,j)}}{t_{ij}} & 0 & \frac{h_4^{(i,j)}}{t_{ij}} & 0 \end{bmatrix} \dots (3.92)$$

onde,

$t_{ij}$  : espessura do elemento no ponto de coordenadas  $r_i$  e  $s_j$  para os quais a matriz está sendo calculada.

$h_k^{(i,j)}$  : função interpoladora  $h_k$  calculada no ponto  $r_i$  e  $s_j$ .

A respeito do valor de  $t_{ij}$ , seria interessante ressaltar que, devido ao fato da fração da estrutura ser de um radiano, seu valor será sempre igual à distância do ponto ao eixo  $X_3$ , conforme a já referida figura 3.35. Nesse caso, fica evidente que pode-se calculá-lo através da expressão :

$$t_{ij} = \sum_k h_k^{(i,j)} x_1^{(k)} \dots (3.93)$$

onde,

$h_k^{(i,j)}$  = valor de  $h_k$  no ponto de coordenadas  $r_i$  e  $s_j$ .

Caso o valor de  $t_{ij}$  resulte zero, ou seja, caso o

ponto considerado esteja no eixo  $X_3$ , a deformação normal ao plano do elemento será igual à deformação  $\epsilon_1$ . Nesse caso, basta montar na quarta linha da matriz os coeficientes já calculados para a primeira linha, obtendo-se desse modo a matriz B completa de quatro linhas por oito colunas.

Entretanto, se a análise realizada pelo elemento for um estado plano de tensão ou deformação, a matriz B será obtida simplesmente pela expressão 3.57, sem qualquer alteração.

### 3.7.5. Matriz de Constantes Elásticas

Este é um ponto onde os elementos tipos 3 e 4 são completamente diferentes. Isso se dá pois o material admitido para este caso pode ser ortotrópico, e portanto a relação entre as tensões e as deformações são completamente alteradas em relação ao elemento anterior. Aliás, para cada tipo de análise que se vai realizar existe uma dada matriz de constantes elásticas.

A matriz de relação entre deformações verificadas e as tensões atuantes, ou seja a matriz C da expressão 3.59, obviamente depende de quais componentes dos estados de tensões e deformações estão sendo consideradas. Para o caso de um estado axissimétrico, pode-se escrever :

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau \\ \sigma_3 \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & 0 & C_{14} \\ C_{12} & C_{22} & 0 & C_{24} \\ 0 & 0 & C_{33} & 0 \\ C_{14} & C_{24} & 0 & C_{44} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma \\ \epsilon_3 \end{Bmatrix} \quad \dots (3.94)$$

onde,

$$A = 1 - \nu_{23}^2 (E_2/E_3)$$

$$B = \nu_{12} + \nu_{13}\nu_{23}(E_2/E_3)$$

$$C = 1 - (E_1/E_3) \left\{ \nu_{13}^2 + (E_3/E_2)(B^2/A) \right\}$$

$$D = \nu_{13} + \nu_{23}(B/A)$$

$$C_{11} = E_1/C$$

$$C_{12} = (B/A) C_{11}$$

$$C_{14} = D C_{11}$$

$$C_{22} = E_2/A + (B/A) C_{12}$$

$$C_{24} = \nu_{23}E_2/A + (B/A) C_{14}$$

$$C_{44} = E_3 + \nu_{13}C_{14} + \nu_{23}C_{24}$$

Se o estado for plano de deformação, ou seja  $\epsilon_3 = 0$ , basta eliminar a quarta linha e a quarta coluna da matriz da expressão 3.94, obtendo-se :

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{12} & C_{22} & 0 \\ 0 & 0 & C_{33} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma \end{Bmatrix} \quad \dots(3.95)$$

onde os valores  $C_{ij}$  são os mesmos da expressão 3.94.

Já para um estado plano de tensão, com a condição de que  $\sigma_3 = 0$ , é necessário operar algumas modificações na matriz da expressão 3.94, para continuar com os mesmos coeficientes  $C_{ij}$ . Após essas alterações, escreve-se :

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau \end{Bmatrix} = \begin{bmatrix} C_{11} - (C_{14}^2/C_{44}) & C_{12} - (C_{14}C_{24}/C_{44}) & 0 \\ & C_{22} - (C_{24}^2/C_{44}) & 0 \\ \text{sim.} & & C_{33} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma \end{Bmatrix} \quad \dots(3.96)$$



### 3.7.6. Matriz de Rigidez em Coordenadas Globais

Quanto à matriz de rigidez dos elementos, a única modificação deste caso em relação ao elemento tipo 4 é a consideração da espessura correta para cada ponto a ser considerado na integração numérica.

Assim sendo, no caso de um estado axissimétrico, a expressão 3.64, do elemento anterior, fica da seguinte maneira :

$$\vec{r} = \sum_{i,j} t_{ij} w_{ij} \vec{R}_{ij} \quad \dots(3.97)$$

onde,

$t_{ij}$  : espessura do elemento no ponto  $r_i$  e  $s_j$  (ver equação 3.93).

Já para os estados planos de tensão e deformação a espessura continua constante, sendo  $t_{ij} = t$  e  $w_{ij} = 1$ , respectivamente.

### 3.7.7. Vetor de Cargas

Os tipos de carregamento sobre o elemento são exatamente os mesmos do elemento anterior. As alterações a serem feitas são devidas à consideração do problema axissimétrico e ao fato de que a face "IJ" não mais se encontrar sempre sobre o eixo local  $x_1$ .

Quanto ao carregamento sobre as faces, o vetor  $f_s$  será alterado, pois as forças normais e na direção da face não estarão corretamente orientadas segundo os eixos locais. Considerando-se  $\theta$  como o ângulo entre o eixo local  $x_1$  e a direção da face "IJ", positivo no sentido anti-horário, pode-se escrever :

$$\tilde{f}_s = \begin{Bmatrix} p_1 \\ p_2 \end{Bmatrix} \quad \dots(3.98)$$

onde,

$$p_1 = p_1 \cos\theta - p_n \sin\theta$$

$$p_2 = p_1 \sin\theta + p_n \cos\theta$$

$p_1$  = força de superfície na direção da face (fig 3.20).

$p_n$  = força de superfície normal à face (fig 3.20).

Outra alteração diz respeito à variação da espessura, que no caso anterior era sempre constante. Agora, em casos de problemas axissimétricos, a espessura em função da coordenada homogênea "r" pode ser escrita :

$$t(r) = \frac{t_I}{2} (1-r) + \frac{t_J}{2} (1+r) \quad \dots(3.99)$$

onde,

$t_I$  : espessura do elemento no nó I ( igual a  $X_2(I)$  ).

$t_J$  : espessura do elemento no nó J ( igual a  $X_2(J)$  ).

Desse modo, substituindo-se o valor de t dado pela expressão 3.99 na equação 3.73, e realizando-se a devida integração, obtém-se :

$$\tilde{p}_1 = \frac{1}{6} \begin{Bmatrix} p_1 (2t_I + t_J) \\ p_2 (2t_I + t_J) \\ p_1 (t_I + 2t_J) \\ p_2 (t_I + 2t_J) \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \dots(3.100)$$

Outra parcela que também apresenta-se alterada é a correspondente aos efeitos de variação de temperatura.

Inicialmente é necessário uma alteração no vetor de deformações iniciais,  $\tilde{\epsilon}_I$ , de modo a se considerar os diferentes coeficientes de dilatação térmica nas direções  $x_1$  e  $x_2$ . Essa alteração resulta :

$$\tilde{\epsilon}_I = \Delta T \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ 0 \end{Bmatrix} \quad \dots(3.101)$$

Também a espessura, que no tipo anterior era sempre constante, precisa ser considerada variável quando tratar-se de problemas axissimétricos. Isso é simples de ser realizado, bastando a substituição do valor de  $t_{ij}$ , fornecido pela equação 3.93, na expressão 3.80. Desse modo a parcela das cargas resulta :

$$p_2 = - \sum_{i,j} t_{ij} \omega_{ij} T_{ij} \quad \dots(3.102)$$

Finalmente, menciona-se que a contribuição do peso próprio também será alterada para os casos de problemas axissimétricos, novamente por efeito da variação da espessura. No caso, essa alteração verifica-se na expressão 3.83, cálculo dos volumes associados aos nós do elemento, resultando então :

$$V_k = \sum_{i,j} t_{ij} \omega_{ij} |\det J_{ij}| h_k^{(i,j)} \quad \dots(3.103)$$

Com essas alterações, o vetor PE pode ser montado de modo idêntico ao apresentado para o elemento anterior.

### 3.7.8. Matriz da Relação Tensão / Deslocamento

A matriz que relaciona as tensões num ponto do

elemento com os deslocamentos é montada com as mesmas expressões utilizadas para o elemento anterior.

### 3.7.9. Vetor de Tensões Iniciais

O vetor de tensões iniciais também é montado com as mesmas expressões utilizadas para o elemento tipo 4. Apenas deve-se ressaltar que o vetor de deformações iniciais a ser utilizado é o fornecido pela expressão 3.101.

### 3.7.10. Modos Incompatíveis

Todo o equacionamento e alterações devidas à inclusão dos modos incompatíveis também são idênticos aos desenvolvidos para o caso anterior.

### 3.7.11. Elementos Triangulares

As condições para a utilização de elementos definidos por apenas três pontos nodais é a mesma do elemento tipo 4. Basta fornecer como dado para os nós K e L do elemento o mesmo ponto nodal da estrutura.

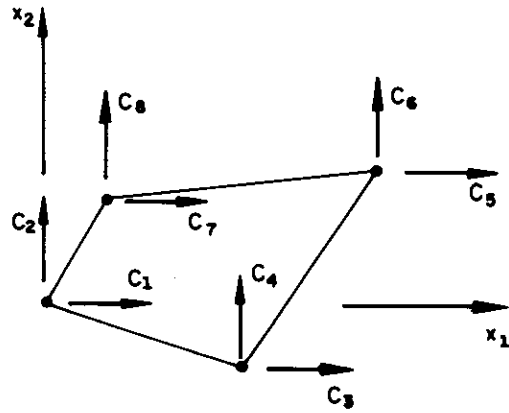


fig. 3.36 Coordenadas nos sistemas local e global

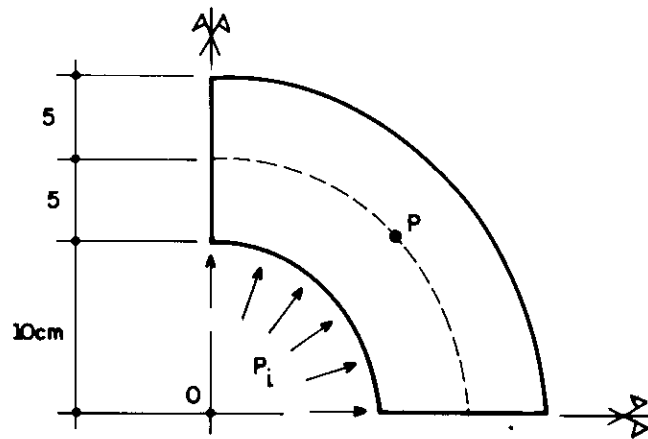


fig. 3.37 Tubo de parede grossa submetido a pressão interna

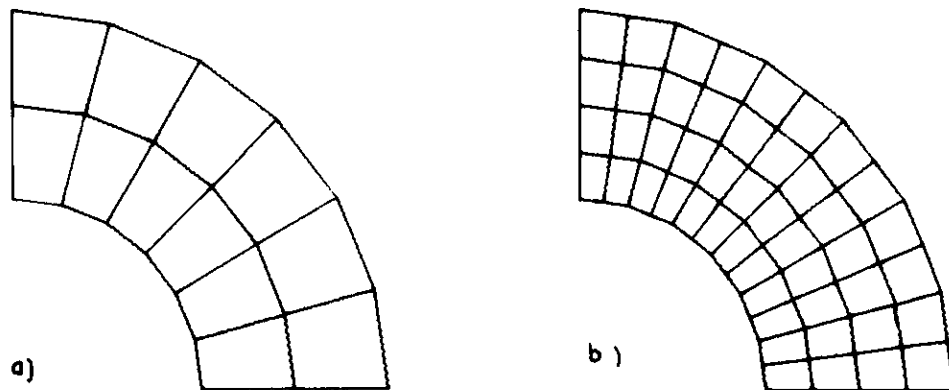


fig. 3.38 Discretizações 1 e 2

### 3.7.12. Desempenho

#### 3.7.12.1. Convergência dos Resultados

Para o elemento tipo 5, poderia ser dispensada a análise sobre a convergência de resultados. Na verdade, como as funções aproximadoras são as mesmas do elemento anterior, poderia assumir-se as conclusões lá apresentadas. Entretanto, vai-se aproveitar a oportunidade de se avaliar o comportamento do elemento para um exemplo onde a solicitação predominante não seja um momento fletor, como o foi no caso anteriormente analisado da viga em balanço.

Analisa-se, assim, o problema de um tubo de parede grossa submetido a uma pressão interna, conforme mostra-se na figura 3.37. Os dados adicionais necessários ao problema são :

$$E = 20\ 000\ \text{KN/cm}^2$$

$$\nu = 0.25$$

$$G = 8\ 000\ \text{Kn/cm}^2$$

$$p_i = 20\ \text{KN/cm}^2$$

Para um acompanhamento da convergência serão definidas quatro discretizações :

Discretização 1 : 12 elementos quadrangulares (fig. 3.38a).

Discretização 2 : 48 elementos quadrangulares (fig. 3.38b).

Discretização 3 : 24 elementos triangulares (fig. 3.39a).

Discretização 4 : 96 elementos triangulares (fig. 3.39b).

De modo semelhante ao caso anterior, também serão obtidos resultados considerando-se ou não a inclusão de modos incompatíveis.

A solução analítica pode ser obtida na referência [3.12]. Serão comparados os valores calculados para a

tensão radial, tensão tangencial e ainda o deslocamento radial, todos no ponto P marcado na figura 3.37. Quanto aos deslocamentos radiais, seus valores são apresentados pela tabela 3.6, em  $\text{cm} \times 10^2$ . As tensões radiais, em  $\text{KN/cm}^2$ , estão organizadas na tabela 3.7. Finalmente, as tensões tangenciais, também em  $\text{KN/cm}^2$ , encontram-se na tabela 3.8. Menciona-se ainda que nas tabelas referidas o cálculo com a inclusão dos modos incompatíveis aparece no título " $\tilde{n}$ comp", e sem a inclusão dos modos, no título "comp".

Deslocamentos Radiais				
Disc.	1	2	3	4
$\tilde{n}$ comp	1.3890	1.4049	1.5142	1.6679
comp	1.3820	1.4028	1.3172	1.3855
analítico	1.4861			

tabela 3.6

Tensões Radiais				
Disc.	1	2	3	4
$\tilde{n}$ comp	5.029	5.049	3.794	4.171
comp	6.838	5.472	5.787	5.284
analítico	5.185			

tabela 3.7

Tensões Tangenciais				
Disc.	1	2	3	4
ñcomp	18.075	18.295	20.395	22.680
comp	17.375	18.130	16.850	17.960
analítico	18.519			

tabela 3.8

Pela observação dos valores obtidos, pode-se concluir que realmente os quadriláteros continuam produzindo melhores resultados que os triângulos. Aliás, nos triângulos com modos incompatíveis inclusos, o resultado novamente divergiu, repetindo o ocorrido para o teste do elemento anterior. Quanto aos triângulos compatíveis, a convergência apresentou-se um pouco mais lenta que nos quadriláteros, mas ainda assim pode ser considerada razoável. Percebe-se que realmente não se deve modelar com triângulos os problemas com momento fletor predominante.

Já para os quadriláteros, a incompatibilidade continua produzindo melhores resultados, se bem que a diferença agora seja muito pequena devido ao fato da solicitação predominante ser por força normal. De qualquer modo os resultados, mesmo com a rede menos refinada, já se aproximam muito do valor teóricamente previsto pelo procedimento analítico.

#### 3.7.12.2. Tempo de Processamento

O desempenho do elemento tipo 5 quanto ao tempo de processamento pode ser considerado, em termos práticos, o mesmo do tipo anterior.



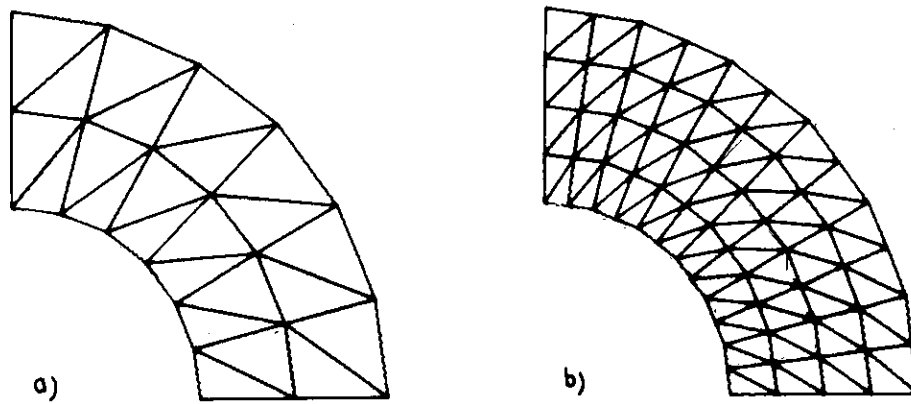


fig. 3.39 Discretizações 3 e 4

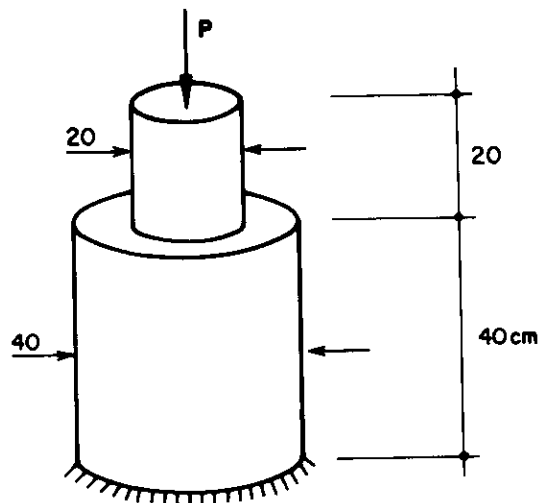


fig. 3.40 Variação de seção num pilar cilíndrico

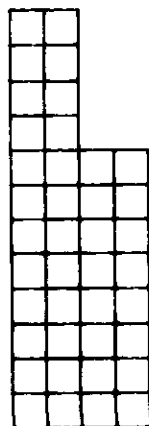


fig. 3.41 Discretização em elementos axissimétricos

### 3.7.13. Exemplo

O exemplo a ser aqui considerado trata da análise das perturbações na interface de variação de seção de um pilar circular de concreto. Para tanto será considerada a região mostrada na figura 3.40, que pelas condições particulares que apresenta pode ser muito bem discretizada através de elementos axissimétricos.

A rede utilizada aparece na figura 3.41 e alguns dados adicionais de interesse são :

$$E = 2\ 000\ \text{KN/cm}^2$$

$$\nu = 0.20$$

$$G = 833\ \text{KN/cm}^2$$

$$p = 2\ \text{KN/cm}^2$$

Como resultados, são apresentadas as tensões longitudinal e radial, respectivamente nas figuras 3.42 e 3.43. Deve-se lembrar que no caso de se considerar o problema em questão como um estado unidimensional de tensões, o valor das tensões longitudinais deve variar de 2  $\text{KN/cm}^2$  no cilindro menor para 0.5  $\text{KN/cm}^2$  no maior.

Exatamente como se processa essa variação é o que as referidas figuras estão mostrando. Inicialmente, observa-se que as tensões radiais apresentam valores relativamente baixos, aproximadamente 10 % do valor longitudinal. Ainda assim, esses valores encontram-se concentrados numa pequena região de aproximadamente 20 cm, cujo centro é exatamente o ponto de variação das seções.

Quanto às tensões longitudinais, o que pode-se perceber é que 40 cm abaixo do ponto de variação das seções seu valor já estará estabilizado no novo patamar. Antes de tudo, isso é uma confirmação do princípio de Saint Venant, que neste caso sugere que as perturbações devem ocorrer apenas até uma distância equivalente à dimensão transversal da peça.

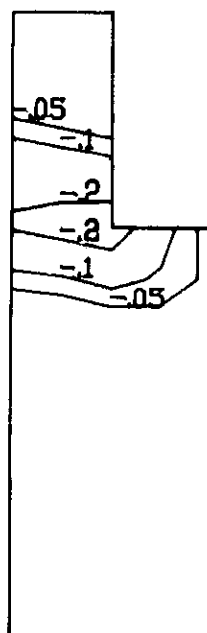


fig. 3.42 Tensão radial

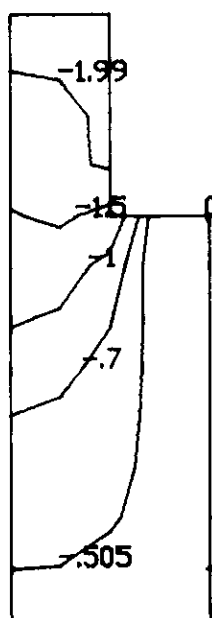


fig. 3.43 Tensão longitudinal

### 3.9. ELEMENTO TIPO 6 - SÓLIDO

#### 3.8.1. Caracterização do Elemento

O elemento sólido é utilizado para análise de estruturas tridimensionais. É definido por oito nós, formando um hexaédro com arestas retas. Essas faces podem estar em direções quaisquer do espaço. Um esboço de uma possível forma do elemento aqui analisado é apresentada pela figura 3.44.

O elemento tem três graus de liberdade por nó, no caso as três translações de um ponto no espaço definido pelos eixos globais. Não é necessária a definição de sistema de referência local. Todas as matrizes e vetores podem ser referenciados diretamente às coordenadas globais, que são apresentadas pela figura 3.45.

O material a ser definido para os elementos pode ser ortotrópico, desde que os seus eixos de simetria coincidam com o sistema de referência adotado, ou seja, o sistema global da estrutura. Desse modo, os parâmetros a serem fornecidos são :

- $\gamma$  = peso específico do material.
- $E_1$  = módulo de elasticidade longitudinal segundo  $X_1$ .
- $E_2$  = módulo de elasticidade longitudinal segundo  $X_2$ .
- $E_3$  = módulo de elasticidade longitudinal segundo  $X_3$ .
- $\nu_{12}$  = coeficiente de Poisson no plano  $X_1X_2$ .
- $\nu_{13}$  = coeficiente de Poisson no plano  $X_1X_3$ .
- $\nu_{23}$  = coeficiente de Poisson no plano  $X_2X_3$ .
- $G_{12}$  = módulo de elasticidade transversal no plano  $X_1X_2$ .
- $G_{13}$  = módulo de elasticidade transversal no plano  $X_1X_3$ .
- $G_{23}$  = módulo de elasticidade transversal no plano  $X_2X_3$ .
- $\alpha_1$  = coeficiente de dilatação térmica segundo  $X_1$ .
- $\alpha_2$  = coeficiente de dilatação térmica segundo  $X_2$ .

$\alpha_3$  = coeficiente de dilatação térmica segundo  $X_3$ .

No elemento aqui desenvolvido também podem ser incluídos modos incompatíveis de deslocamentos. No caso serão nove valores adicionais de graus de liberdade sem correspondência física com deslocamentos nos nós. Da mesma maneira que nos elementos tipos 4 e 5 estudados anteriormente, esses modos produzem um aumento sensível na rapidez de convergência, principalmente no caso de problemas submetidos predominantemente a momentos fletores.

Inclusive, para os elementos tridimensionais, essa melhora pode ser muito importante para uma redução do esforço computacional, que costuma ser significativo. Entretanto, sua utilização continua a merecer alguns cuidados, devido ao fato de se destruir a compatibilidade do elemento.

Quanto aos carregamentos permitidos, eles podem ser divididos em três grupos. Inicialmente força distribuída aplicada sobre as faces, com dois tipos de definição possíveis. A do tipo 1 é normal à superfície. A do tipo 2, fornecida já segundo os eixos globais da estrutura. Qualquer um dos dois tipos podem ser aplicados em qualquer face. Também é possível a consideração de uma variação de temperatura, calculada como sendo a diferença entre a média das temperaturas nodais e um valor de referência para cada elemento, fornecido pelo usuário. Por fim, existe a consideração automática do peso próprio, de acordo com frações segundo os eixos globais.

A utilização do presente elemento se dá em casos de estruturas que não tenham um ou dois eixos predominantes, sendo seu comportamento tipicamente tridimensional. Pode-se citar o caso de blocos de fundação, barragens e até mesmo cascas de espessura relativamente grande, desde que sejam incluídos os modos incompatíveis.

Finalmente, menciona-se que as deduções aqui apresentadas foram desenvolvidas com base nas mesmas

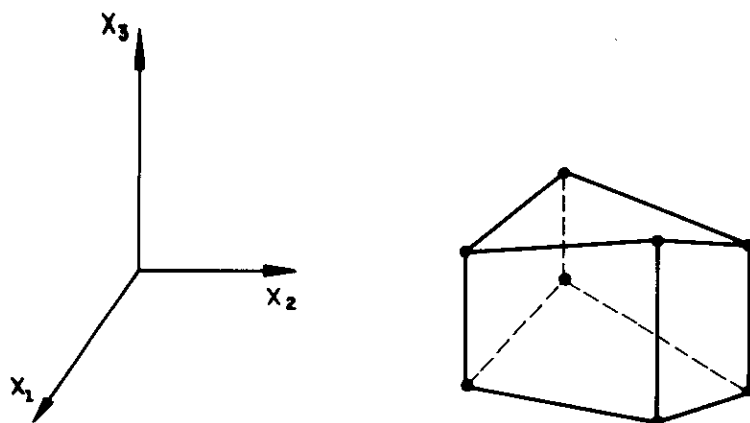


fig. 3.44 Elemento sólido

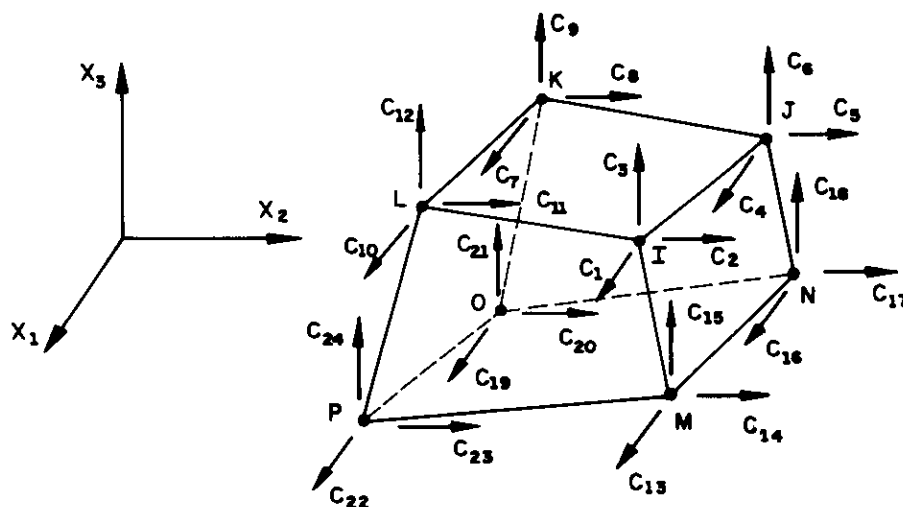


fig. 3.45 Coordenadas globais

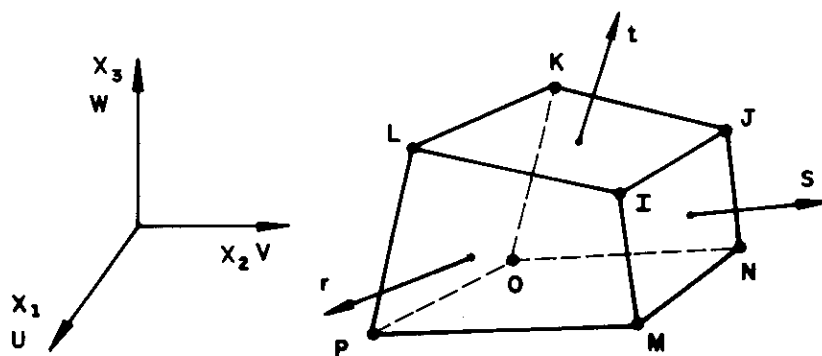


fig. 3.46 Sistema local homogêneo

referências já citadas para os dois elementos precedentes. Aliás, sendo este elemento baseado em uma formulação isoparamétrica, em muitos casos o seu desenvolvimento pode ser entendido como uma expansão para o tridimensional dos conceitos enunciados para o plano no caso dos elementos precedentes. Por essa razão, sempre que possível, serão feitas referências ao que já foi desenvolvido para os dois últimos elementos, de modo a tornar-se esta explanação mais sucinta.

### 3.8.2. Funções de Interpolação

Conforme foi mencionado, o elemento aqui desenvolvido é isoparamétrico. Para a definição das funções interpoladoras foi adotado um sistema de referência auxiliar composto por três eixos naturais. Eles aparecem na figura 3.46, juntamente com o sistema de referência e a indicação dos deslocamentos "U", "V" e "W".

Assim sendo, as coordenadas de um ponto qualquer do elemento podem ser escritas, em função dos valores nodais, de acordo com as expressões :

$$\begin{aligned}
 x_1(r,s,t) &= \sum_{i=1}^8 h_i x_1^{(i)} \\
 x_2(r,s,t) &= \sum_{i=1}^8 h_i x_2^{(i)} \quad \dots(3.104) \\
 x_3(r,s,t) &= \sum_{i=1}^8 h_i x_3^{(i)}
 \end{aligned}$$

onde,

$x_n^{(i)}$  : valor da coordenada  $x_n$  do nó "i".

$$h_1 = \frac{1}{8} (1+r)(1+s)(1+t)$$

$$\begin{aligned}
h_2 &= \frac{1}{8} (1-r)(1+s)(1+t) \\
h_3 &= \frac{1}{8} (1-r)(1-s)(1+t) \\
h_4 &= \frac{1}{8} (1+r)(1-s)(1+t) \\
h_5 &= \frac{1}{8} (1+r)(1+s)(1-t) \\
h_6 &= \frac{1}{8} (1-r)(1+s)(1-t) \\
h_7 &= \frac{1}{8} (1-r)(1-s)(1-t) \\
h_8 &= \frac{1}{8} (1+r)(1-s)(1-t)
\end{aligned}$$

De modo semelhante, também os deslocamentos serão expressos através dos valores nodais, de acordo com as equações :

$$\begin{aligned}
U(r,s,t) &= \sum_{i=1}^8 h_i U^{(i)} \\
V(r,s,t) &= \sum_{i=1}^8 h_i V^{(i)} \quad \dots(3.105) \\
W(r,s,t) &= \sum_{i=1}^8 h_i W^{(i)}
\end{aligned}$$

Desse modo, a relação entre as derivadas do sistema global e as derivadas do sistema natural pode ser escrita como sendo :

$$\begin{Bmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial s} \\ \frac{\partial}{\partial t} \end{Bmatrix} = \begin{bmatrix} \frac{\partial X_1}{\partial r} & \frac{\partial X_2}{\partial r} & \frac{\partial X_3}{\partial r} \\ \frac{\partial X_1}{\partial s} & \frac{\partial X_2}{\partial s} & \frac{\partial X_3}{\partial s} \\ \frac{\partial X_1}{\partial t} & \frac{\partial X_2}{\partial t} & \frac{\partial X_3}{\partial t} \end{bmatrix} \begin{Bmatrix} \frac{\partial}{\partial X_1} \\ \frac{\partial}{\partial X_2} \\ \frac{\partial}{\partial X_3} \end{Bmatrix} \quad \dots(3.106)$$

Ou ainda :

$$\left\{ \frac{\partial}{\partial r} \right\} = \underset{\sim}{J} \left\{ \frac{\partial}{\partial X} \right\} \quad , \quad \left\{ \frac{\partial}{\partial r} \right\} = \underset{\sim}{J}^{-1} \left\{ \frac{\partial}{\partial X} \right\} \quad \dots(3.107)$$



onde a matriz  $J$  é o jacobiano da transformação.

A determinação do jacobiano é muito simples de ser feita quando considera-se um ponto de coordenadas  $r_i$ ,  $s_j$  e  $t_k$ . Basta efetuar derivadas das expressões 3.104 em relação a "r", "s" e "t", substituindo-se então os valores das coordenadas naturais do ponto considerado. Nesse caso, será utilizada a notação  $J_{ijk}$ . Outra característica importante é o determinante do jacobiano, também de cálculo trivial, e que será notado por "det  $J_{ijk}$ ".

### 3.8.3. Matriz da Relação Deformações / Deslocamentos

A matriz que relaciona deformações com deslocamentos é construída com derivadas convenientes das funções de interpolação. No caso tridimensional, as expressões das deformações em função de derivadas dos deslocamentos podem ser escritas como :

$$\begin{aligned} \epsilon_1 &= \frac{\partial U}{\partial X_1} \\ \epsilon_2 &= \frac{\partial V}{\partial X_2} \\ \epsilon_3 &= \frac{\partial W}{\partial X_3} \\ \gamma_{12} &= \frac{\partial U}{\partial X_2} + \frac{\partial V}{\partial X_1} \\ \gamma_{13} &= \frac{\partial U}{\partial X_3} + \frac{\partial W}{\partial X_1} \\ \gamma_{23} &= \frac{\partial V}{\partial X_3} + \frac{\partial W}{\partial X_2} \end{aligned} \quad \dots (3.108)$$

Considerando-se que no presente caso a matriz  $B$  faz a relação :

$$\begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \gamma_{12} \\ \gamma_{13} \\ \gamma_{23} \end{Bmatrix} = \underset{\sim}{B} \begin{Bmatrix} U_1 \\ V_1 \\ W_1 \\ \vdots \\ \vdots \\ U_8 \\ V_8 \\ W_8 \end{Bmatrix} \quad \dots(3.109)$$

Pode-se escrever B como composta por oito submatrizes, ou seja :

$$\underset{\sim}{B} = \left[ \underset{\sim}{B}_1 \quad \underset{\sim}{B}_2 \quad \underset{\sim}{B}_3 \quad \underset{\sim}{B}_4 \quad \underset{\sim}{B}_5 \quad \underset{\sim}{B}_6 \quad \underset{\sim}{B}_7 \quad \underset{\sim}{B}_8 \right] \quad \dots(3.110)$$

onde,

$$\underset{\sim}{B}_i = \begin{bmatrix} \frac{\partial h_i}{\partial X_1} & 0 & 0 \\ 0 & \frac{\partial h_i}{\partial X_2} & 0 \\ 0 & 0 & \frac{\partial h_i}{\partial X_3} \\ \frac{\partial h_i}{\partial X_2} & \frac{\partial h_i}{\partial X_1} & 0 \\ \frac{\partial h_i}{\partial X_3} & 0 & \frac{\partial h_i}{\partial X_2} \\ 0 & \frac{\partial h_i}{\partial X_3} & \frac{\partial h_i}{\partial X_2} \end{bmatrix}$$

Convém ressaltar que as derivadas das funções interpoladoras podem ser determinadas em relação aos eixos cartesianos aos grupos, de acordo com a expressão :

$$\begin{Bmatrix} \frac{\partial h_i}{\partial X_1} \\ \frac{\partial h_i}{\partial X_2} \\ \frac{\partial h_i}{\partial X_3} \end{Bmatrix} = \underset{\sim}{J}^{-1} \begin{Bmatrix} \frac{\partial h_i}{\partial r} \\ \frac{\partial h_i}{\partial s} \\ \frac{\partial h_i}{\partial t} \end{Bmatrix} \quad \dots(3.111)$$

Já as derivadas das funções aproximadoras em relação a "r", "s" e "t" são triviais, como pode-se observar pelas funções correspondentes.

### 3.8.4. Matriz de Constantes Elásticas

A matriz de constantes elásticas, matriz C, dá a relação entre o estado de deformações e o estado de tensões para um determinado ponto. Para o presente caso, material ortotrópico com eixos de simetria coincidentes com os eixos globais, pode-se escrever :

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \tau_{12} \\ \tau_{13} \\ \tau_{23} \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \gamma_{12} \\ \gamma_{13} \\ \gamma_{23} \end{Bmatrix} \quad \dots(3.112)$$

onde,

$$A = 1 - \nu_{23}^2 (E_2/E_3)$$

$$B = \nu_{12} + \nu_{13}\nu_{23} (E_2/E_3)$$

$$C = 1 - (E_1/E_3) \left\{ \nu_{13}^2 + (E_3/E_2)(B^2/A) \right\}$$

$$D = \nu_{13} + \nu_{23}(B/A)$$

$$C_{11} = E_1/C$$

$$C_{12} = (B/A) C_{11}$$

$$C_{13} = D C_{11}$$

$$C_{22} = E_2/A + (B/A) C_{12}$$

$$C_{23} = \nu_{23} E_2/A + (B/A) C_{12}$$

$$C_{33} = E_3 + \nu_{13}C_{14} + \nu_{23}C_{24}$$

$$C_{44} = G_{12}$$

$$C_{55} = G_{13}$$

$$C_{66} = G_{23}$$

### 3.8.5. Matriz de Rigidez em Coordenadas Globais

A matriz de rigidez do elemento pode ser calculada pela expressão :

$$\tilde{R}_E = \int_{\tilde{V}} \tilde{B}^T \tilde{C} \tilde{B} dV \quad \dots(3.113)$$

Entretanto, para realizar-se a integração em função das coordenadas naturais, é necessário que se escreva :

$$dV = |\det \tilde{J}| dr ds dt \quad \dots(3.114)$$

Portanto, tem-se :

$$\tilde{R}_E = \int_{\tilde{V}} \tilde{B}^T \tilde{C} \tilde{B} |\det \tilde{J}| dr ds dt \quad \dots(3.115)$$

Considerando-se a integração numérica pelo processo de Gauss, pode-se escrever :

$$\tilde{R}_E = \sum_{i,j,k} \omega_{ijk} \tilde{R}_{ijk} \quad \dots(3.116)$$

onde,

$\omega_{ijk}$  : fator de peso para o ponto  $r_i$ ,  $s_j$  e  $t_k$ .

$$\tilde{R}_{ijk} = \tilde{B}_{ijk}^T \tilde{C} \tilde{B}_{ijk} |\det \tilde{J}_{ijk}|$$

A integração numérica, por "default", é feita por 8 pontos, utilizando-se as formulas de Gauss. Caso o elemento seja muito distorcido, o usuário pode optar ainda pela utilização de 27 ou mesmo 64 pontos. Entretanto, conforme será discutido no item correspondente, o desempenho do elemento, em termos de tempo de processamento, sofre uma queda apreciável.

### 3.8.6. Vetor de Cargas

Foi mencionado que os elementos podem receber solicitações de forças distribuídas sobre as faces, variação de temperatura e consideração automática de peso próprio.

Quanto à ação sobre as faces, ela deve ser constante. Caso seja do tipo 1, atua na direção normal à face carregada. Caso seja tipo 2, podem ser fornecidos três valores, já em relação aos eixos de referência global. Seus sentidos positivos são referenciados, no caso de tipo 1, pelos coordenadas homogêneas  $r, s$  e  $t$ , mostradas na figura 3.46. Para cada face a ação será positiva quando na direção do eixo homogêneo correspondente. No caso do tipo 2, os sentidos positivos serão os dos eixos de referência.

A consideração desse carregamento depende da face de atuação. Por exemplo, considerando-se uma ação tipo 1 positiva atuando na face  $r = + 1$ , conforme mostra-se na figura 3.47, a integração dessa tensão na área correspondente é feita de acordo com a expressão :

$$\tilde{P}_1 = \int_S \tilde{H}_S^T \tilde{f}_S dS \quad \dots(3.117)$$

A matriz  $H$  é um arranjo conveniente das funções de interpolação de modo a conseguir-se a seguinte relação :

$$\begin{Bmatrix} U \\ V \\ W \end{Bmatrix} = \underset{\sim}{H} \begin{Bmatrix} U^{(1)} \\ V^{(1)} \\ W^{(1)} \\ \vdots \\ U^{(8)} \\ V^{(8)} \\ W^{(8)} \end{Bmatrix} \quad \dots(3.118)$$

Pode-se escrevê-la :

$$\underset{\sim}{H} = \left\{ \begin{matrix} H_{\sim 1} & H_{\sim 2} & H_{\sim 3} & H_{\sim 4} & H_{\sim 5} & H_{\sim 6} & H_{\sim 7} & H_{\sim 8} \end{matrix} \right\} \dots(3.119)$$

onde,

$$H_{\sim i} = \begin{bmatrix} h_i & 0 & 0 \\ 0 & h_i & 0 \\ 0 & 0 & h_i \end{bmatrix}$$

Já o índice "s", que aparece na matriz H da expressão 3.117, apenas indica que ela será calculada para uma determinada face, no caso a face onde  $r = +1$ . Logicamente, nesse caso, as funções interpoladoras terão apenas os eixos s e t como variáveis.

Para o cálculo do vetor  $f_s$ , quando tratar-se de uma carga tipo 1, é necessário a definição de um versor normal à face, no ponto considerado. Isso pode ser feito da maneira que se segue :

$$\begin{aligned} \vec{a} &= \left\{ \begin{matrix} \frac{\partial X_1}{\partial s} & \frac{\partial X_2}{\partial s} & \frac{\partial X_3}{\partial s} \end{matrix} \right\} \\ \vec{b} &= \left\{ \begin{matrix} \frac{\partial X_1}{\partial t} & \frac{\partial X_2}{\partial t} & \frac{\partial X_3}{\partial t} \end{matrix} \right\} \quad \dots(3.120) \\ \vec{n} &= \vec{a} \times \vec{b} \quad , \quad \vec{v} = \vec{n} / |\vec{n}| \end{aligned}$$

Considerando-se  $v_1, v_2$  e  $v_3$  como as coordenadas do vetor  $\vec{v}$  e "p" como a força distribuída, pode-se escrever  $f_s$ :

$$\vec{f}_s = \begin{Bmatrix} p v_1 \\ p v_2 \\ p v_3 \end{Bmatrix} \quad \dots(3.121)$$

Para a carga tipo 2 o procedimento simplifica-se, pois os valores  $p_x$ ,  $p_y$  e  $p_z$  já são fornecidos nas direções globais. Assim sendo, o vetor  $\vec{f}_s$  é constante para toda a face e pode ser escrito como :

$$\vec{f}_s = \begin{Bmatrix} p_x \\ p_y \\ p_z \end{Bmatrix} \quad \dots(3.122)$$

Finalmente, deve-se lembrar que a integração será realizada em função das coordenadas homogêneas e portanto deve-se escrever :

$$dS = |\det J_s| ds dt$$

onde o determinante de  $J_s$  é o módulo do vetor  $\vec{n}$  apresentado nas expressões 3.120.

Após o que foi colocado, basta que se faça a integração numérica da equação 3.117, no caso com a utilização de quatro pontos de Gauss. Desse modo a expressão resulta :

$$P_{\sim 1} = \sum_{i,j} \omega_{ij} (H_s)_{ij} (f_s)_{ij} |\det (J_s)_{ij}| ds dt \quad \dots(3.123)$$

A consideração do efeito de uma variação de temperatura sobre o elemento é dada por uma equação idêntica à 3.76. Entretanto, neste caso o vetor de deformações iniciais resulta :

$$\varepsilon_I = \Delta T \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \dots(3.125)$$

Não se pode esquecer ainda que :

$$dV = |\det \tilde{J}| dr ds dt \quad \dots(3.126)$$

Desse modo, também por integração numérica, nos mesmos pontos da montagem da matriz de rigidez, obtém-se :

$$P_{\tilde{2}} = - \sum_{i,j,k} \omega_{ijk} B_{\tilde{ijk}}^T C_{\tilde{I}} \varepsilon_I |\det \tilde{J}_{ijk}| \quad \dots(3.127)$$

Finalmente, existe a consideração do peso próprio. Do mesmo modo que nos outros elementos, ele será adicionado diretamente aos nós correspondentes, multiplicado pelos fatores definidos pelo usuário para cada eixo global. Para tanto o programa calcula o volume associado a um determinado nó do elemento, de  $V_1$  a  $V_8$ , de acordo com a expressão :

$$V_1 = \sum_{i,j,k} \omega_{ijk} |\det \tilde{J}_{ijk}| h_1^{(i,j,k)} \quad \dots(3.128)$$

onde,

$h_1^{(i,j,k)}$  : função interpoladora  $h_1$  calculada para o ponto  $r_i, s_j$  e  $t_k$ .

Assim sendo, a parcela devida ao peso próprio resulta:



$$\tilde{P}_3 = \gamma \begin{Bmatrix} v_1 a_1 \\ v_1 a_2 \\ v_1 a_3 \\ \vdots \\ v_B a_1 \\ v_B a_2 \\ v_B a_3 \end{Bmatrix} \quad \dots(3.129)$$

onde,

$a_1$ ,  $a_2$  e  $a_3$  : frações do peso próprio a serem consideradas segundo os eixos  $X_1$ ,  $X_2$  e  $X_3$ .

Obtendo-se para o vetor de cargas do elemento :

$$\tilde{P}_E = \tilde{P}_1 + \tilde{P}_2 + \tilde{P}_3 \quad \dots(3.130)$$

### 3.8.7. Matriz da Relação Tensão / Deslocamento

A matriz que relaciona as tensões locais com os deslocamentos globais é montada levando-se em conta em quais pontos do elemento os resultados serão apresentados. Isso é feito por uma expressão muito semelhante à considerada nos elementos tipos 4 e 5. Entretanto, neste caso, todos os vetores e matrizes deduzidos já referem-se diretamente ao sistema global. Desse modo, não existe a matriz de transformação  $\lambda$ , utilizada nos outros casos. Assim sendo, as tensões num ponto qualquer do elemento resultam :

$$\tilde{\sigma} = C B U \quad \dots(3.131)$$

No caso do elemento tipo 6, o usuário pode optar por até quatro pontos onde as tensões serão apresentadas. Assim

sendo, cada vez que a equação 3.131 for aplicada para um determinado ponto escolhido, estarão sendo montadas seis linhas da matriz de relação das tensões com os deslocamentos. Essa matriz tem, portanto, dimensão variável, podendo apresentar de 6 a 24 linhas, sempre com 24 colunas colunas. Para o caso de tensões apresentadas em apenas um ponto, de coordenadas  $r_i$ ,  $s_j$  e  $t_k$ , pode-se escrever :

$$\tilde{TD} = \tilde{C} \tilde{B}_{ijk} \quad \dots(3.132)$$

### 3.8.8. Vetor de Tensões Iniciais

De forma semelhante aos dois elementos anteriores, o vetor de tensões iniciais contém apenas o efeito da variação da temperatura. Desse modo, para cada carregamento definido, pode-se escrever :

$$\tilde{TI} = - \tilde{C} \tilde{\epsilon}_I \quad \dots(3.133)$$

onde,

$\tilde{\epsilon}_I$  : deformações iniciais fornecidas pela expressão 3.125.

Também aqui essas tensões iniciais são constantes para todo o elemento. Finalmente, menciona-se que de modo semelhante à matriz TD, para cada ponto onde os resultados sejam apresentados é necessário montar seis posições no vetor TI.

### 3.8.9. Modos Incompatíveis

O procedimento de inclusão de modos incompatíveis neste elemento tipo 6 é bastante análogo ao definido para os elementos tipos 4 e 5. Desde a alteração das funções interpoladoras, até a posterior eliminação desses graus de liberdade adicionais em determinadas matrizes e vetores, o procedimento adotado é bastante semelhante.

Entretanto, aqui serão em número de nove os termos adicionados nas funções de interpolação, sendo três para cada eixo homogêneo. Desse modo, as referidas funções resultam [3.13] :

$$\begin{aligned}U(r,s,t) &= \sum_{i=1}^8 h_i U^{(i)} + \alpha_1(1-r^2) + \alpha_2(1-s^2) + \alpha_3(1-t^2) \\V(r,s,t) &= \sum_{i=1}^8 h_i V^{(i)} + \beta_1(1-r^2) + \beta_2(1-s^2) + \beta_3(1-t^2) \\W(r,s,t) &= \sum_{i=1}^8 h_i W^{(i)} + \delta_1(1-r^2) + \delta_2(1-s^2) + \delta_3(1-t^2) \\& \dots(3.134)\end{aligned}$$

A inclusão desses modos incompatíveis pode melhorar de forma notável a convergência dos resultados, principalmente para casos onde se verifica a atuação de momentos fletores. Considerando-se que o esforço computacional a ser desenvolvido para a análise de uma estrutura bem discretizada é bastante grande, a inclusão desses graus de liberdade adicionais pode ser muito importante.

### 3.8.10. Desempenho

#### 3.8.10.1. Convergência de Resultados

Para examinar a convergência do elemento sólido será considerado o problema da viga em balanço mostrada na figura 3.22. Inclusive os dados de características elásticas serão os mesmos do item 3.6.12.1.

O acompanhamento dos resultados obtidos será realizado para três diferentes discretizações, cujas redes são definidas da maneira que se segue :

Discretização 1 : 8 elementos ( fig 3.48 ).

Discretização 2 : 16 elementos ( fig 3.49 ).

Discretização 3 : 128 elementos ( fig 3.50 ).

Como serão utilizados somente elementos paralelepípedicos em um problema onde a ação predominante é um momento fletor, a análise será feita exclusivamente com a inclusão dos modos incompatíveis. Além disso, como já foi feito no item 3.6.12.1, vai-se controlar os valores do deslocamento na extremidade livre, apresentados em cm na tabela 3.9, a tensão normal na extremidade superior de uma seção próxima ao engaste, em  $\text{KN}/\text{cm}^2$  na tabela 3.10, e ainda da tensão tangencial no eixo da peça, também em  $\text{KN}/\text{cm}^2$  na tabela 3.11.

A observação dos resultados anotados nas referidas tabelas mostra, de forma absolutamente clara, que a discretização mais apurada, para o problema em questão, somente tem utilidade no cálculo da tensão de cisalhamento. Essa tendência aliás já havia sido notada desde que o mesmo exemplo foi resolvido com o elemento membrana quadrangular não conforme. Aqui esta tendência torna-se ainda mais clara, podendo-se perceber quase que uma degradação dos resultados devido ao excesso de graus de liberdade

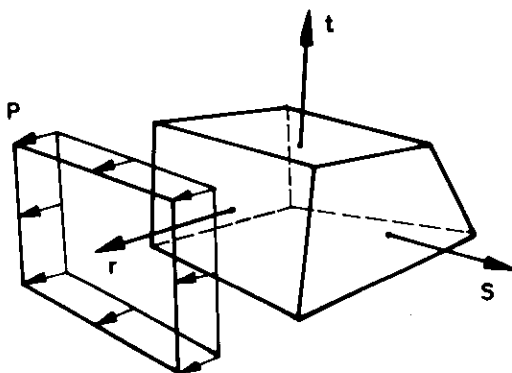


fig. 3.47 Força de superfície do tipo 1 na face  $r = + 1$

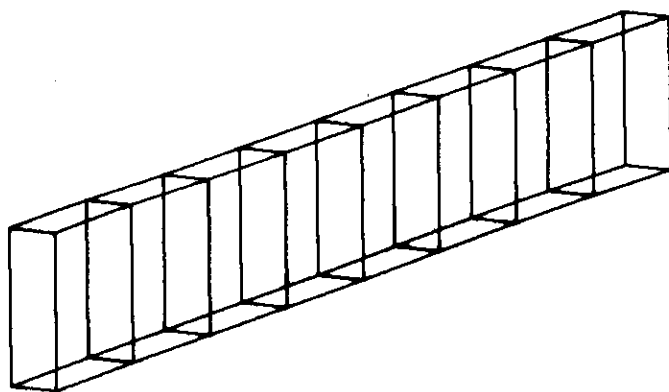


fig. 3.48 Discretização 1

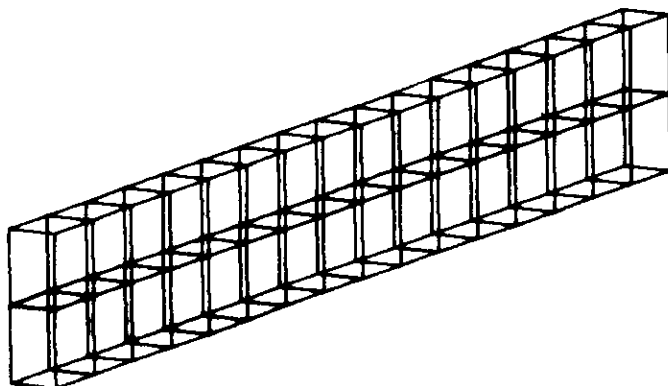


fig. 3.49 Discretização 2

utilizados na solução. Apenas para a tensão de cisalhamento, constante em  $2.500 \text{ KN/cm}^2$  para as discretizações 1 e 2, é que percebe-se uma melhora significativa.

---

Deslocamento na extremidade livre

---

Disc.	1	2	3
M.E.F	1.53379	1.53754	1.54502
Viga	1.53600		

---

tabela 3.9

---

Tensão Normal na Extremidade Superior da Seção

---

Disc.	1	2	3
M.E.F.	97.50	101.23	106.97
Viga	97.50	101.25	106.88

---

tabela 3.10

---

Tensão Tangencial no Eixo da Viga

---

Disc.	1	2	3
M.E.F.	2.500	2.500	3.550
Viga	3.750		

---

tabela 3.11

### 3.8.10.2. Tempo de Processamento

Quanto ao desempenho em relação aos tempos de processamento, é necessário considerar-se que o elemento tipo 6 pode apresentar valores maiores ou menores para a montagem de suas matrizes e vetores, dependendo do número de pontos de Gauss utilizados no cálculo da matriz de rigidez. Assim sendo, esses tempos estão anotados na tabela 3.12, sendo considerados os valores possíveis de 8, 27 ou 64 pontos de Gauss.

Núm. de Ptos	8	27	64
Tempo (seg)	1.332	2.930	6.157

tabela 3.12

Como pode-se perceber pelos resultados anotados, o tempo de montagem das matrizes e vetores para o elemento sólido praticamente dobra ao se considerar um outro nível de integração. Essa informação deve ser levada em conta pelo usuário, de modo a não tornar excessivamente demorada uma análise com a consideração desse tipo de elemento.

Já quanto ao cálculo das tensões, o tempo também varia em relação ao número de pontos nos quais essa saída está sendo providenciada. Entretanto, essa variação é pouco significativa e para o caso médio de saída em dois pontos, pode-se adotar o valor de 0.75 seg.

### 3.8.11. Exemplo

Como exemplo para o elemento sólido vai-se considerar um caso bastante diferente dos que têm sido analisados até

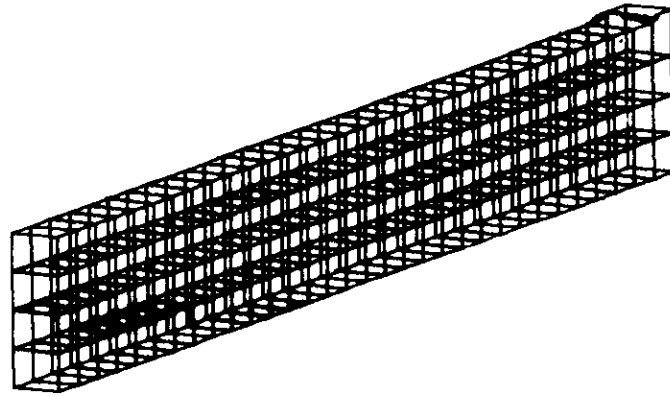


fig. 3.50 Discretização 3

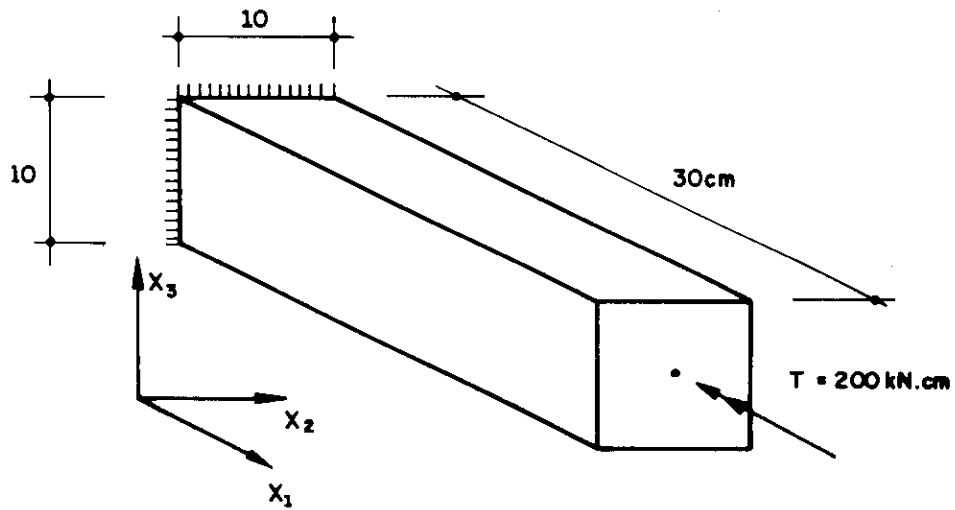


fig. 3.51 Eixo submetido a momento torçor

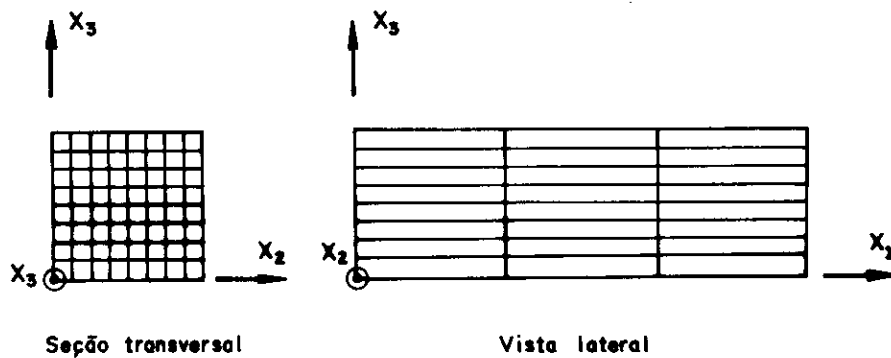


fig. 3.52 Discretização adotada



aqui. Trata-se de uma torção atuando num eixo de seção quadrada, conforme mostra-se na figura 3.51.

Devido à grande variação da tensão de cisalhamento nas seções transversais, é necessário uma discretização relativamente fina no plano  $X_2X_3$ . Já ao longo do eixo, portanto segundo  $X_1$ , a discretização pode ser pobre que os resultados não serão afetados. De acordo com essas informações é que foi montada a rede de discretização mostrada na figura 3.52.

Como características adicionais pode-se citar :

$$E = 20\ 000\ \text{KN/cm}^2.$$

$$\nu = 0.25$$

$$G = 8\ 000\ \text{KN/cm}^2$$

Os resultados interessantes de serem examinados são exatamente as já referidas tensões tangenciais. Mesmo com a aplicação do carregamento na seção da extremidade livre e a consideração do engaste na outra extremidade, esses valores pouco variam ao longo do comprimento da peça. Entretanto, por uma questão de maior fidelidade, vai-se tomar os valores obtidos para os elementos médios em relação ao comprimento. Esses valores encontram-se desenhados em gráficos nas figuras 3.53, tensão tangencial no plano  $X_1X_2$ , e 3.54, tensão tangencial no plano  $X_1X_3$ .

O que se pode perceber pela análise desses resultados é que realmente os maiores valores de sollicitação ocorrem nos pontos de meio de lado da seção, sendo os cantos da mesma os pontos de extremidade menos sollicitados. Evidentemente que no centro da seção a tensão é rigorosamente zero. Por fim, é interessante mencionar que pela observação das curvas, pode-se concluir que o máximo valor da tensão tangencial na seção deve ser da ordem de  $0.90\ \text{KN/cm}^2$ . Esse valor está próximo do máximo previsto pela solução por série, que resulta em torno de  $0.95\ \text{KN/cm}^2$ .

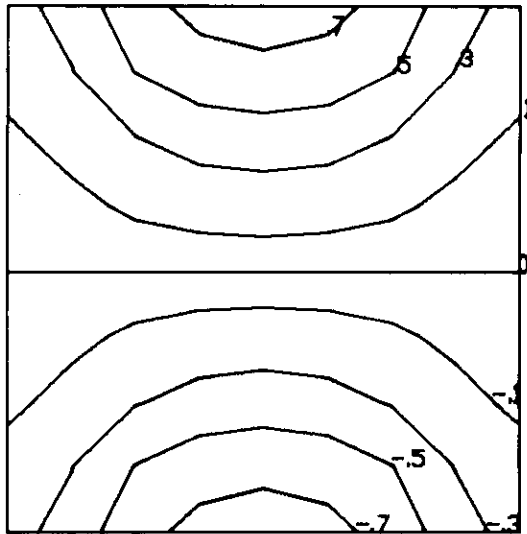


fig. 3.53 Tensão tangencial no plano  $x_1x_2$  ( horizontal )

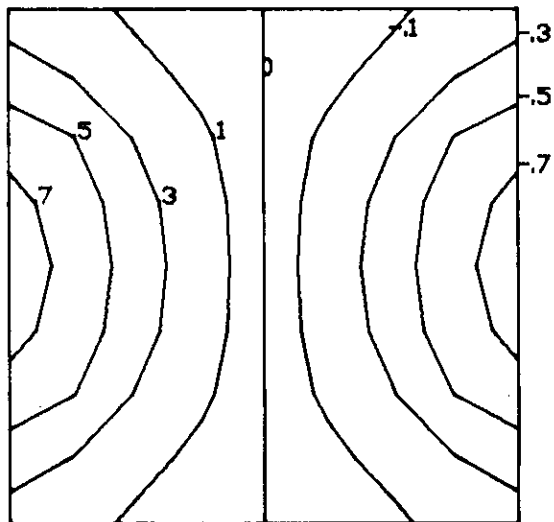


fig. 3.54 Tensão tangencial no plano  $x_1x_3$  ( vertical )

### 3.9. ELEMENTO TIPO 7 - PLACA OU CASCA

#### 3.9.1. Caracterização do Elemento

O elemento tipo 7 pode ser usado para modelagens de placas ou cascas de espessura relativamente pequena. Esse elemento, no caso do programa LS, nada mais é que a composição de um elemento utilizado para modelagem de placas fletidas e um elemento utilizado para discretização de estados planos de tensão.

Desse modo, as solicitações que forem normais ao plano do elemento são resistidas pela rigidez fornecida pelo elemento placa. Já as que estiverem contidas no próprio plano, pela rigidez de membrana. Essa composição equivale a definir, sobre uma malha de nós, elementos dos dois tipos com os mesmos nós de extremidade.

Entretanto, se o desejo do usuário é a utilização de um elemento para modelagem de estruturas que possam ser definidas exclusivamente como placas, basta que o programa, avisado dessa condição, deixe de somar às equações do problema a rigidez do elemento membrana. Esta é a razão de se mencionar o elemento tipo 7 como sendo adequado à modelagem de placas ou cascas.

O elemento pode ser triangular ou quadrangular, estando definido em qualquer posição do espaço. No caso das cascas, cinco graus de liberdade são definidos para cada nó: duas translações no próprio plano, devidas ao componente membrana, e mais uma translação normal ao plano e duas rotações, devidas ao componente placa. Já para o caso de um elemento exclusivamente placa, obviamente somente restarão os graus de liberdade a ele associados, no caso o deslocamento normal ao plano e as rotações, perfazendo um total de três graus de liberdade para cada nó.

Tanto para o elemento triangular como para o quadrangular são definidos os sistemas locais de referência mostrados na figura 3.55. Essa referência é composta por um eixo  $x_1$ , definido do nó "I" para o nó "J", e ainda  $x_2$  e  $x_3$ , respectivamente no plano e normal ao plano do elemento, formando um sistema dextrorso.

Os materiais a serem definidos para os elementos podem ser ortotrópicos, desde que seus eixos de simetria coincidam com os eixos locais de referência dos elementos. Nesse caso, serão os seguintes os parâmetros a serem fornecidos :

- $\gamma$  : peso específico do material (opcional).
- $E_1$  : módulo de elasticidade longitudinal segundo  $x_1$ .
- $E_2$  : módulo de elasticidade longitudinal segundo  $x_2$ .
- $\nu_{12}$  : coeficiente de Poisson no plano  $x_1 x_2$ .
- $G$  : módulo de elasticidade transversal.
- $\alpha_1$  : coeficiente de variação térmica segundo  $x_1$  (opcional).
- $\alpha_2$  : coeficiente de variação térmica segundo  $x_2$  (opcional).

O elemento tipo 7, quando utilizado na modelagem de cascas, pode receber quatro tipos de carregamento. Em termos de forças, pode-se definir uma carga distribuída constante, normal ao plano do elemento, e consideração automática de peso próprio, de acordo com os detalhes já definidos para os outros tipos. Além disso, são permitidas ações térmicas de variação de temperatura no plano do elemento e um gradiente de temperatura na direção normal ao plano.

A utilização do presente elemento está muito ligada a pavimentos de edifícios. Pavimentos em estrutura convencional, com lajes apoiadas ou engastadas, e pavimentos em laje cogumelo constituem as principais utilizações. Além disso, existem ainda as cascas propriamente ditas, como as de revolução, de translação, parabolóides, hiperbolóides, etc.

### 3.9.2. Considerações Gerais Sobre a Formulação do Elemento

Ao contrário dos elementos anteriormente definidos, neste caso existe uma diferenciação bastante acentuada entre os elementos triangulares ou quadrangulares. Em última análise o programa utiliza-se exclusivamente de elementos triangulares, montando os quadrângulos a partir da junção de quatro triângulos, com posterior condensação estática do nó central. A figura 3.56 ilustra esse procedimento.

O usuário define o elemento quadrangular normalmente, através dos pontos nodais "I", "J", "K" e "L". O programa calcula a posição do centróide do quadrilátero fornecido, "O", e divide a área em quatro triângulos. Então são calculadas as matrizes e vetores para cada um desses triângulos. Posteriormente, faz-se a condensação estática dos graus de liberdade do nó "O", obtendo-se as matrizes e vetores agora apenas para os nós "I", "J", "K" e "L".

O elemento a ser utilizado para a determinação da rigidez de membrana é o CST ("Constant Strain Triangle"). Trata-se de um elemento isoparamétrico cuja formulação foi deduzida no item 3.6. Entretanto, naquele caso a dedução das matrizes e vetores do elemento era feita genericamente para o elemento quadrangular, obtendo-se o triangular por degeneração. Como aqui tem-se especificamente o interesse voltado para o triangular e se está num bloco de comandos diferente, o CST será deduzido através de uma formulação direta. Como referências para esse desenvolvimento, podem ser citadas as mesmas publicações já apresentadas para os outros elementos isoparamétricos deduzidos.

Já o elemento adotado para a placa é conhecido como LCCT-9 ("Linear Curvature Compatible Triangle" de 9 graus de liberdade). Como o próprio nome indica é um elemento perfeitamente conforme de nove parâmetros nodais, um deslocamento e duas rotações para cada nó. Seu

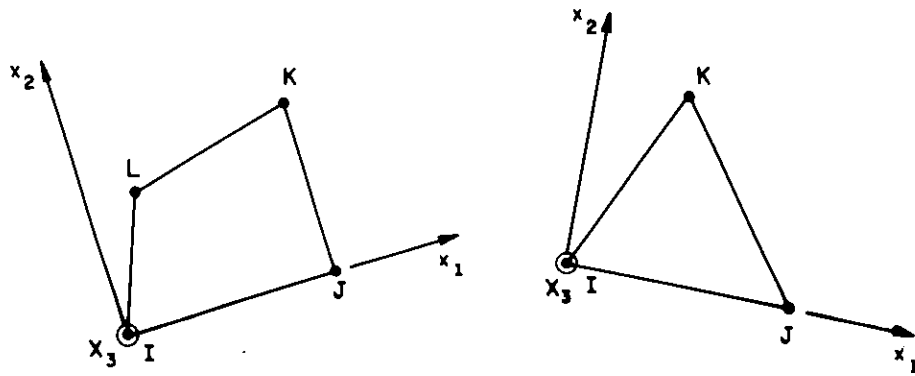


fig. 3.55 Sistemas locais de referência

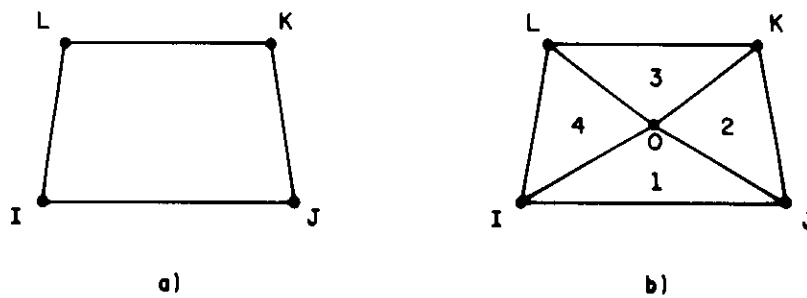


fig. 3.56 Divisão do quadrilátero em triângulos

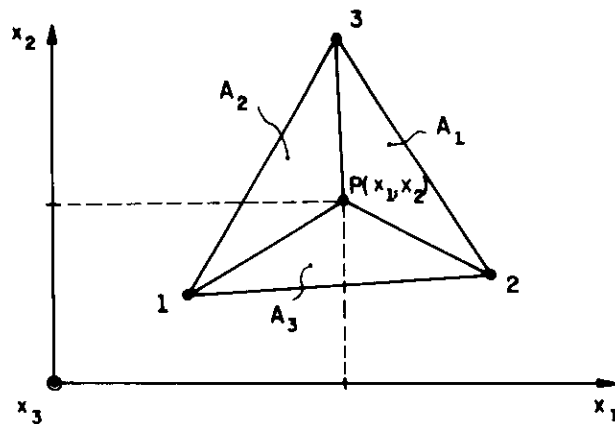


fig. 3.57 Coordenadas triangulares

desenvolvimento está baseado na referência [3.14]. Encontra-se também deduzido em [3.15], onde é chamado de HCT.

### 3.9.3. Coordenadas Homogêneas Triangulares

Tanto no desenvolvimento do elemento triangular para membranas, quanto para placas, é interessante trabalhar com coordenadas homogêneas triangulares. Considere-se a figura 3.57, onde é definido um triângulo com os nós numerados em um sentido conveniente. Para o ponto P  $(x_1, x_2)$  as coordenadas triangulares podem ser definidas como sendo :

$$h_i = \frac{A_i}{A} \quad \dots(3.135)$$

onde,

$A_i$  = área do triângulo Pjk  
 $A$  = área total 123

Como pode-se perceber, as referidas coordenadas são valores adimensionais obtidos pela relação de áreas convenientes. Por uma questão de clareza, a figura 3.58 apresenta o desenvolvimento das funções  $h_1$ ,  $h_2$  e  $h_3$  em todo o domínio triangular do elemento

Resta agora explicitar, em função de coordenadas dos nós do elemento, como calcular essas funções para um determinado ponto de coordenadas  $x_1$  e  $x_2$ . Considerando-se que  $x^{(n)}$  represente uma determinada coordenada do nó "n", pode-se escrever :

$$h_i = \frac{1}{2A} \left[ c_i + b_i x_1 + a_i x_2 \right] \quad \dots(3.136)$$

onde,

$$a_i = x_1^{(k)} - x_1^{(j)}$$

$$b_i = x_2^{(j)} - x_2^{(k)}$$

$$c_i = x_1^{(j)} x_2^{(k)} - x_1^{(k)} x_2^{(j)}$$

$$A = \text{área do triângulo} = \frac{1}{2} (b_1 a_2 - b_2 a_1)$$

Ressalta-se que para as expressões anteriores está sendo considerada uma permutação cíclica dos índices  $i, j$  e  $k$ , de modo que :

$$i = 1, j = 2 \text{ e } k = 3$$

$$i = 2, j = 3 \text{ e } k = 1$$

$$i = 3, j = 1 \text{ e } k = 2$$

É interessante deixar claro que as coordenadas homogêneas definidas são as funções de interpolação da geometria dos elementos triangulares a serem desenvolvidos. Pode-se perceber isso com facilidade pela simples observação da figura 3.58, onde essas funções aparecem desenhadas. Assim sendo, as coordenadas de um ponto qualquer do elemento podem ser determinadas em função dos valores nodais através das expressões :

$$x_1 = h_1 x_1^{(1)} + h_2 x_1^{(2)} + h_3 x_1^{(3)} \quad \dots(3.137)$$

$$x_2 = h_1 x_2^{(1)} + h_2 x_2^{(2)} + h_3 x_2^{(3)}$$

Para finalizar esta apresentação das coordenadas triangulares, é importante mencionar alguns parâmetros que interessarão diretamente à dedução do elemento LCCT-9. Eles encontram-se apresentados na figura 3.59. Trata-se dos valores de  $L_1, L_2, L_3$ , comprimentos dos lados do triângulo,  $d_1, d_2$  e  $d_3$ , distâncias ao ponto  $P$  medidas sobre esses lados, e ainda  $H_1, H_2$  e  $H_3$ , dimensões do triângulo em direções convenientes. Com esses valores é possível ainda



definir :

$$\lambda_i = d_i / L_i \quad \dots(3.138)$$

$$\mu_i = 1 - \lambda_i$$

### 3.9.4. Matriz de Constantes Elásticas

Rigorosamente falando, as matrizes de constantes elásticas para os elementos de membrana e placa seriam diferentes. Entretanto, para o caso das placas, desprezam-se as tensões que não estejam no plano do elemento. Desse modo, para um ou outro caso, a matriz resulta a mesma.

Considerando-se o material ortotrópico definido no item 3.9.1, pode-se escrever a matriz de relação entre as tensões e as deformações, que aparece na equação 3.59, como sendo :

$$\tilde{C} = \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{12} & C_{22} & 0 \\ 0 & 0 & C_{33} \end{bmatrix} \quad \dots(3.139)$$

onde,

$$e = E_2 / E_1$$

$$C_{11} = \frac{E_1}{(1 - e\nu_{12}^2)}$$

$$C_{12} = e\nu_{12} C_{11}$$

$$C_{22} = \frac{E_2}{(1 - e\nu_{12}^2)}$$

$$C_{33} = G$$

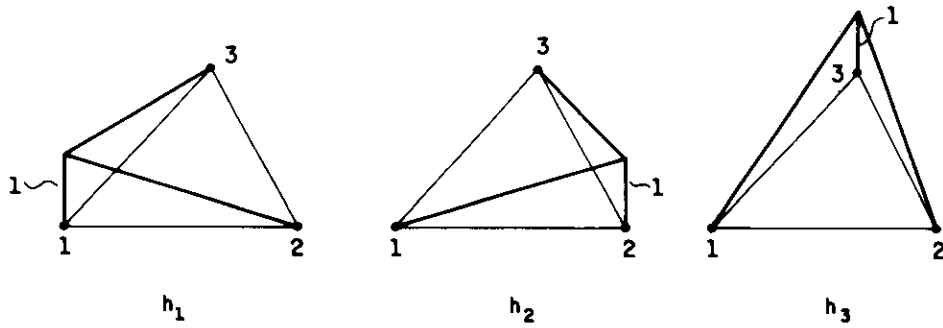


fig. 3.58 Funções  $h_1$ ,  $h_2$  e  $h_3$

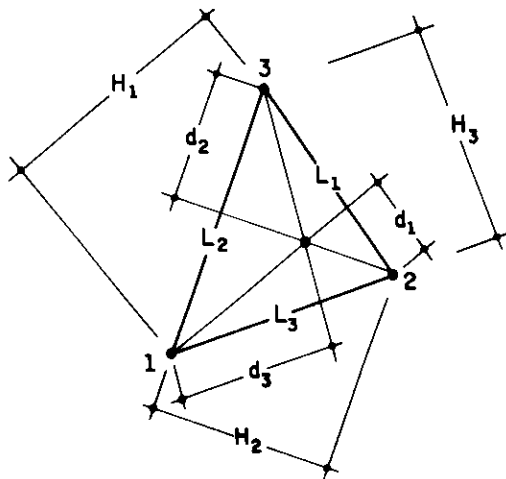


fig 3.59 Parâmetros para uso no elemento placa

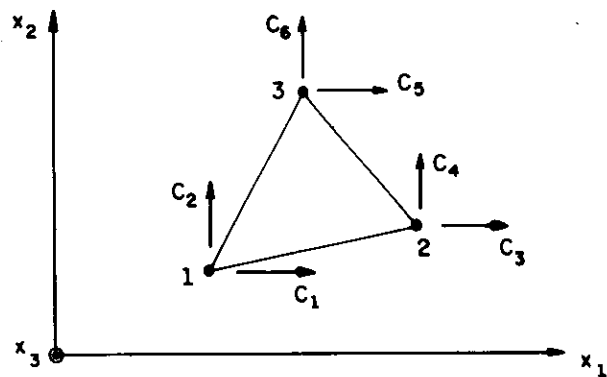


fig. 3.60 Coordenadas locais no elemento membrana

### 3.9.5. Elemento Triangular de Membrana

#### 3.9.5.1. Coordenadas e Funções Interpoladoras

Conforme foi mencionado, o elemento a ser utilizado para avaliação das propriedades de membrana será o CST. Tomando-se por referência o sistema local definido no item 3.9.1, as coordenadas do elemento são as apresentadas na figura 3.60.

Como trata-se de um elemento isoparamétrico, as funções interpoladoras a serem utilizadas serão as mesmas apresentadas para a geometria. Desse modo, pode-se escrever os deslocamentos "u" e "v" para um ponto qualquer do elemento através das expressões :

$$u = h_1 u^{(1)} + h_2 u^{(2)} + h_3 u^{(3)} \quad \dots(3.140)$$

$$v = h_1 v^{(1)} + h_2 v^{(2)} + h_3 v^{(3)}$$

#### 3.9.5.2. Matriz de Rigidez

As relações entre as deformações e os deslocamentos são as mesmas apresentadas pelas equações 3.56. Desse modo, a matriz B que relaciona essas grandezas pode ser escrita como :

$$\tilde{B} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & 0 & \frac{\partial h_2}{\partial x_1} & 0 & \frac{\partial h_3}{\partial x_1} & 0 \\ 0 & \frac{\partial h_1}{\partial x_1} & 0 & \frac{\partial h_2}{\partial x_1} & 0 & \frac{\partial h_3}{\partial x_1} \\ \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_1} & \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_1} & \frac{\partial h_3}{\partial x_1} & \frac{\partial h_3}{\partial x_1} \end{bmatrix} \quad \dots(3.141)$$

ou seja,

$$\tilde{B} = \frac{1}{2A} \begin{bmatrix} b_1 & 0 & b_2 & 0 & b_3 & 0 \\ 0 & a_1 & 0 & a_2 & 0 & a_3 \\ a_1 & b_1 & a_2 & b_2 & a_3 & b_3 \end{bmatrix} \quad \dots(3.142)$$

onde,  $a_i$  e  $b_i$  tem suas expressões apresentadas no item 3.9.3.

A matriz de rigidez pode então ser determinada por um expressão totalmente análoga à 3.61. Analisando-se as grandezas que estão sendo integradas, verifica-se que todas são constantes no domínio do elemento. Assim sendo tem-se :

$$\tilde{r} = \tilde{B}^T \tilde{C} \tilde{B} \int_V dV \quad \dots(3.143)$$

Lembrando-se que a espessura do elemento "t" é também uma constante, conclui-se que a matriz de rigidez resulta :

$$\tilde{r} = \left( \tilde{B}^T \tilde{C} \tilde{B} \right) A t \quad \dots(3.144)$$

onde,

A : área do elemento

Portanto, é muito simples obterem-se expressões explícitas para os termos da matriz de rigidez, matriz quadrada de ordem seis e simétrica. Neste trabalho, devido à necessidade de se resumir esta apresentação, deixa-se de apresentar essas 21 expressões.

### 3.9.5.3. Vetor de cargas

A única solicitação que deve ser levada em consideração para o cálculo do vetor de cargas consistente do elemento membrana é a variação de temperatura ao longo

do elemento. A expressão básica para esse cálculo é a 3.76, sendo o vetor de tensões iniciais dado pela equação 3.101.

Devido ao fato da matriz B ser uma constante, o vetor de cargas resulta :

$$\underset{\sim}{p} = - \left[ \underset{\sim}{B}^T \underset{\sim}{C} \underset{\sim}{\varepsilon}_I \right] A t \quad \dots(3.145)$$

ou, na forma explícita,

$$\underset{\sim}{p} = - \frac{\Delta T t}{2} \left\{ \begin{array}{l} b_1(C_{11}\alpha_1 + C_{12}\alpha_2) \\ a_1(C_{12}\alpha_1 + C_{22}\alpha_2) \\ b_2(C_{11}\alpha_1 + C_{12}\alpha_2) \\ a_2(C_{12}\alpha_1 + C_{22}\alpha_2) \\ b_3(C_{11}\alpha_1 + C_{12}\alpha_2) \\ a_3(C_{12}\alpha_1 + C_{22}\alpha_2) \end{array} \right\} \quad \dots(3.146)$$

#### 3.9.5.4. Matriz da Relação Tensões / Deslocamentos

Conforme foi mencionado o elemento apenas possui saída de resultados para o seu centróide. A expressão que possibilita esse cálculo é :

$$\underset{\sim}{\sigma} = \underset{\sim}{C} \underset{\sim}{B} \underset{\sim}{u} \quad \dots(3.147)$$

Portanto, percebe-se que a matriz que se procura determinar, e que será chamada de "td" devido ao fato de referir-se a coordenadas locais, pode ser encontrada simplesmente multiplicando-se a matriz de constantes elásticas pela matriz B. Em termos explícitos, esse procedimento resulta :

$$\tilde{t}_d = \frac{1}{2A} \begin{bmatrix} C_{11}b_1 & C_{12}a_1 & C_{11}b_2 & C_{12}a_2 & C_{11}b_3 & C_{12}a_3 \\ C_{12}b_1 & C_{22}a_1 & C_{12}b_2 & C_{22}a_2 & C_{12}b_3 & C_{22}a_3 \\ C_{33}a_1 & C_{33}b_1 & C_{33}a_2 & C_{33}b_2 & C_{33}a_3 & C_{33}b_3 \end{bmatrix} \quad \dots(3.148)$$

### 3.9.5.5. Vetor de Tensões Iniciais

Também aqui apenas o efeito da variação<sup>N</sup> de temperatura deve ser levado em conta. Isso é muito simples de ser feito, bastando seguir-se a equação 3.89 com o vetor de deformações iniciais dado pela expressão 3.101. Esse cálculo resulta :

$$\tilde{t}_i = -\Delta T \begin{Bmatrix} C_{11}\alpha_1 + C_{12}\alpha_2 \\ C_{12}\alpha_1 + C_{22}\alpha_2 \\ 0 \end{Bmatrix} \quad \dots(3.149)$$

### 3.9.6. Elemento Triangular de Placa

#### 3.9.6.1. Elemento Completo e Sub-triângulos Componentes

Para desenvolver o elemento LCCT-9, adotado para a obtenção da rigidez de placa, é necessário dividir o elemento em mais três sub-triângulos. Para se evitar confusões, sempre que se referir ao elemento todo mencionar-se-á a expressão "elemento completo". Já quando a menção dizer respeito às partes, serão utilizadas as expressões "sub-triângulos" ou "sub-elementos".

Os deslocamentos e rotações para o triângulo completo são as apresentadas na figura 3.61a, onde também são mostrados os mencionados sub-triângulos definidos pelos

pontos nodais e o centróide do elemento completo. Nesses sub-triângulos também estarão presentes os deslocamentos e rotações já mostrados para o elemento completo, além de mais três valores no centróide, conforme são apresentados na figura 3.61b.

Para que as relações geométricas deduzidas no item 3.9.3 continuem a ser utilizadas para cada sub-triângulo, é necessário uma renumeração dos pontos nodais. Essa renumeração, mostrada entre parênteses na própria figura 3.61b, é tal que inicialmente são renumerados os nós externos e posteriormente o centróide.

### 3.9.6.2. Funções Interpoladoras

Dentro de cada sub-elemento, os deslocamentos serão aproximados por valores de deslocamentos e rotações definidos nos pontos nodais. Desse modo, pode-se escrever :

$$\tilde{w}^{[i]} = \tilde{H}_I^{[i]} \tilde{u}^{[i]} \quad \dots(3.150)$$

onde,

$\tilde{w}^{[i]}$  : deslocamento  $w$  no sub-triângulo "i".

$\tilde{H}_I^{[i]}$  : vetor de funções interpoladoras.

$\tilde{u}^{[i]}$  : vetor de graus de liberdade do sub-elemento "i".

Por exemplo , para o sub-triângulo 1 , pode-se escrever :

$$\tilde{u}^{[1]T} = \left\{ w_2 \quad \theta_{12} \quad \theta_{22} \quad w_3 \quad \theta_{13} \quad \theta_{23} \quad w_0 \quad \theta_{10} \quad \theta_{20} \quad \theta_5 \right\}$$

... (3.151)

onde,

$w_i$  : deslocamento do nó "i".  
 $\theta_{ij}$  : rotação, em torno de  $x_i$ , no nó "j".

Já as funções de interpolação, que compõem o vetor  $H_I^{[i]}$ , podem ser encontradas na página 407 da referência [3.14] e deixam de ser aqui reproduzidos para não alongar-se demais a presente explanação. Apenas menciona-se que são dez expressões para cada sub-elemento e baseiam-se nas funções  $h_i$  apresentadas pela expressão 3.136.

Agora, os deslocamentos  $w^{[i]}$  podem ser referenciados a um vetor que contenha todos os graus de liberdade presentes nos sub-elementos. Esse vetor é dado por :

$$\tilde{u}^T = \left\{ \begin{array}{ccccccccc} w_1 & \theta_{11} & \theta_{21} & w_2 & \theta_{12} & \theta_{22} & w_3 & \theta_{13} & \theta_{23} \\ & & & \theta_4 & \theta_5 & \theta_6 & w_0 & \theta_{10} & \theta_{20} \end{array} \right\} \dots(3.152)$$

ou,

$$\tilde{u} = \left\{ \begin{array}{c} \tilde{u}_e \\ \tilde{u}_0 \end{array} \right\} \dots(3.153)$$

onde,

$\tilde{u}_e$  : deslocamentos e rotações externos.

$\tilde{u}_0$  : deslocamentos e rotações no centróide.

Nesse caso, considerando-se o sub-triângulo 1, os deslocamentos resultariam :

$$w^{[1]} = \left\{ \begin{array}{cc} H_e^{[1]} & H_0^{[1]} \end{array} \right\} \left\{ \begin{array}{c} \tilde{u}_e \\ \tilde{u}_0 \end{array} \right\} \dots(3.154)$$

Ressalta-se que os sub-vetores  $H_e^{[1]}$  e  $H_0^{[1]}$  são compostos pelas mesmas funções que compõem  $H_I^{[1]}$ , com as adaptações necessárias ao novo arranjo do graus de



liberdade.

Equações semelhantes podem ser montadas para os sub-triângulos 2 e 3, resultando no seguinte conjunto :

$$\begin{Bmatrix} w^{[1]} \\ w^{[2]} \\ w^{[3]} \end{Bmatrix} = \begin{bmatrix} H_{\tilde{e}}^{[1]} & H_{\tilde{o}}^{[1]} \\ H_{\tilde{e}}^{[2]} & H_{\tilde{o}}^{[2]} \\ H_{\tilde{e}}^{[3]} & H_{\tilde{o}}^{[3]} \end{bmatrix} \begin{Bmatrix} u_{\tilde{e}} \\ u_{\tilde{o}} \end{Bmatrix} \quad \dots(3.155)$$

### 3.9.6.3. Compatibilidade de Rotações Internas

A expressão 3.155 não representa um estado de deslocamentos compatíveis. Apesar dos deslocamentos transversais serem os mesmos, quando se trata de um lado comum entre dois sub-triângulos, as rotações normais diferem. É necessário portanto alterar as funções de interpolação para que essa compatibilidade seja garantida.

Considere-se, por exemplo, a junção entre os sub-elementos 1 e 2. Verifica-se, na figura 3.61a, que o ponto nodal 9 está contido nessa interface. Nesse caso, a rotação normal a essas faces pode ser calculada através de relações desenvolvidas para cada um dos sub-triângulos adjacentes. Considerando-se o sub-elemento 1, tem-se :

$$\theta_9^{[1]} = \begin{Bmatrix} d_{\tilde{e}}^{[1]} & d_{\tilde{o}}^{[1]} \end{Bmatrix} \begin{Bmatrix} u_{\tilde{e}} \\ u_{\tilde{o}} \end{Bmatrix} \quad \dots(3.156)$$

onde,

$$d_{\tilde{e}}^{[1]} = \frac{\partial H_{\tilde{e}}^{[1]}}{\partial n} \quad \text{e} \quad d_{\tilde{o}}^{[1]} = \frac{\partial H_{\tilde{o}}^{[1]}}{\partial n} \quad , \text{ calculados para o nó 9.}$$

Já com a consideração do sub-elemento 2 tem-se :

$$\theta_9^{[3]} = \left\{ \begin{matrix} d_7^{[3]} & d_{90}^{[3]} \end{matrix} \right\} \left\{ \begin{matrix} u_e \\ u_o \end{matrix} \right\} \quad \dots(3.157)$$

Entretanto, para que a compatibilidade seja mantida, é necessário impor  $\theta_9^{(1)} = -\theta_9^{(3)}$ . Considerando-se que o mesmo tratamento deve ser dispensado aos nós 7 e 8, pode-se organizar essas expressões em forma matricial da maneira que se segue :

$$\left[ \begin{matrix} (d_7^{[1]} + d_7^{[3]}) & (d_{70}^{[1]} + d_{70}^{[3]}) \\ (d_8^{[1]} + d_8^{[3]}) & (d_{80}^{[1]} + d_{80}^{[3]}) \\ (d_9^{[1]} + d_9^{[3]}) & (d_{90}^{[1]} + d_{90}^{[3]}) \end{matrix} \right] \left\{ \begin{matrix} u_e \\ u_o \end{matrix} \right\} = \underset{\sim}{0} \quad \dots(3.158)$$

ou ainda,

$$\left[ \begin{matrix} D & \\ \underset{\sim}{D} & \underset{\sim}{D}_o \end{matrix} \right] \left\{ \begin{matrix} u_e \\ u_o \end{matrix} \right\} = \underset{\sim}{0} \quad \dots(3.159)$$

Portanto, os valores de  $u_o$  que levam à compatibilidade desejada podem ser expressos por :

$$u_o = -D_o^{-1} D_e u_e = \underset{\sim}{L} u_e \quad \dots(3.160)$$

Desse modo, a equação 3.155 pode ser reescrita, agora apenas em função dos deslocamentos e rotações  $u_e$ , da seguinte maneira :

$$\left\{ \begin{matrix} w^{[1]} \\ w^{[2]} \\ w^{[3]} \end{matrix} \right\} = \left[ \left\{ \begin{matrix} \underset{\sim}{H}_e^{[1]} \\ \underset{\sim}{H}_e^{[2]} \\ \underset{\sim}{H}_e^{[3]} \end{matrix} \right\} + \left\{ \begin{matrix} \underset{\sim}{H}_o^{[1]} \\ \underset{\sim}{H}_o^{[2]} \\ \underset{\sim}{H}_o^{[3]} \end{matrix} \right\} \underset{\sim}{L} \right] \underset{\sim}{u}_e \quad \dots(3.161)$$

ou ainda,

$$\begin{Bmatrix} w^{[1]} \\ w^{[2]} \\ w^{[3]} \end{Bmatrix} = \begin{bmatrix} H_c^{[1]} \\ H_c^{[2]} \\ H_c^{[3]} \end{bmatrix} \tilde{u} \quad \dots(3.162)$$

onde,

$$\tilde{u}^T = \left\{ w_1 \theta_{11} \theta_{21} w_2 \theta_{12} \theta_{22} w_3 \theta_{13} \theta_{23} \theta_4 \theta_5 \theta_6 \right\}$$

Portanto as funções de interpolação  $H_c^{[i]}$  são totalmente compatíveis e permitem que se escreva o campo de deslocamentos em função dos graus de liberdade do elemento completo. Essas funções, que demandam um grande trabalho algébrico para serem explicitadas, também são apresentadas pela referência [3.14], na página 440. De modo semelhante às interpolações não compatíveis, também não se reproduzirá essas expressões neste trabalho.

#### 3.9.6.4. Eliminação dos Nós de Meio de Lado

Após a obtenção da expressão 3.162, ainda é necessário eliminarem-se os graus de liberdade  $\theta_4$ ,  $\theta_5$  e  $\theta_6$  para obter-se o elemento LCCT-9. Essa modificação é feita com muita facilidade escrevendo-se os graus de liberdade a serem eliminados em função das rotações dos nós vizinhos. Desse modo, tem-se :

$$\begin{aligned} \theta_4 &= \frac{1}{2} (\theta_{11} + \theta_{12}) \frac{a_3}{L_3} - \frac{1}{2} (\theta_{21} + \theta_{22}) \frac{b_3}{L_3} \\ \theta_5 &= \frac{1}{2} (\theta_{12} + \theta_{13}) \frac{a_1}{L_1} - \frac{1}{2} (\theta_{22} + \theta_{23}) \frac{b_1}{L_1} \quad \dots(3.163) \\ \theta_6 &= \frac{1}{2} (\theta_{13} + \theta_{11}) \frac{a_2}{L_2} - \frac{1}{2} (\theta_{23} + \theta_{21}) \frac{b_2}{L_2} \end{aligned}$$

onde,  $a_i$  e  $L_i$  são os parâmetros para o elemento completo calculados no item 3.9.3.

Com essas expressões, volta-se à equação 3.162 e eliminam-se as três últimas colunas da matriz de funções de interpolação. Após essa eliminação, a equação que permite o cálculo do campo de deslocamentos em função dos graus de liberdade dos nós 1, 2 e 3 pode ser escrita como sendo :

$$\tilde{w} = \tilde{H} \tilde{u} \quad \dots(3.164)$$

onde,

$$\tilde{w}^T = \left\{ w^{[1]} \quad w^{[2]} \quad w^{[3]} \right\}$$

$\tilde{H}$  : matriz de funções interpoladoras compatíveis em relação aos graus de liberdade nos nós 1, 2 e 3.

$$\tilde{u}^T = \left\{ w_1 \quad \theta_{11} \quad \theta_{21} \quad w_2 \quad \theta_{12} \quad \theta_{22} \quad w_3 \quad \theta_{13} \quad \theta_{23} \right\}$$

### 3.9.6.5. Campo de Curvaturas Interno

Para a determinação da rigidez e outros parâmetros que dizem respeito ao elemento em questão, é necessário que se conheça a matriz que relaciona os deslocamentos e rotações nos nós de extremidade com as curvaturas num determinado ponto interior do elemento. Essas curvaturas podem ser estabelecidas pela diferenciação do campo de deslocamentos da maneira que se segue :

$$\tilde{x}^{[i]} = \tilde{T}^{[i]} \tilde{u} \quad \dots(3.165)$$

onde,

$$\tilde{x}^{[i]T} = \left[ \frac{\partial^2 w^{[i]}}{\partial x_1^2} \quad \frac{\partial^2 w^{[i]}}{\partial x_2^2} \quad 2 \frac{\partial^2 w^{[i]}}{\partial x_1 \partial x_2} \right]$$

$$\tilde{T}^{[i]} = \begin{bmatrix} \frac{\partial^2 H^{[i]}}{\partial x_1^2} & \frac{\partial^2 H^{[i]}}{\partial x_2^2} & 2 \frac{\partial^2 H^{[i]}}{\partial x_1 \partial x_2} \end{bmatrix}$$

Encontrado o campo de curvaturas para cada sub-triângulo, pode-se calcular as curvaturas nodais desses sub-triângulos. Para isso basta desenvolver a matriz  $T^{[i]}$  com as respectivas coordenadas desses pontos. Assim fazendo, pode-se escrever :

$$\tilde{x}_n^{[i]} = \tilde{T}_n^{[i]} u_n \quad \dots(3.166)$$

Agora, as curvaturas internas podem ser expressas em função dos valores nodais por meio das funções de interpolação desenvolvidas no item 3.9.3. Matricialmente pode-se escrever :

$$\tilde{x}^{[i]} = \tilde{\phi} \tilde{x}_n^{[i]} \quad \dots(3.167)$$

onde,

$$\tilde{\phi} = \begin{bmatrix} h_1 & h_2 & h_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_1 & h_2 & h_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 \end{bmatrix}$$

Ressalta-se que, da maneira como encontra-se organizada a matriz  $\phi$ , as curvaturas nodais estão dispostas de modo que nas primeiras posições estão as derivadas em  $x_1$ , para os três nós, logo após as derivadas em  $x_2$  e finalmente as derivadas em  $x_1$  e  $x_2$ .

#### 3.9.6.6. Matriz de Rigidez

Para o cálculo da matriz de rigidez, deve-se lembrar que a energia acumulada é obtida pela integração, na área do elemento, das curvaturas multiplicadas pelos momentos

fletores. Já os momentos fletores, quando o elemento tem espessura constante "t", podem ser escritos em função das próprias curvaturas da maneira que se segue :

$$\underset{\sim}{m}^{[i]} = \underset{\sim}{D} \underset{\sim}{\kappa}^{[i]} \quad \dots(3.168)$$

onde,

$$\underset{\sim}{m}^{[i]T} = \left\{ m_{11}^{[i]} \quad m_{22}^{[i]} \quad m_{12}^{[i]} \right\}$$

$$\underset{\sim}{D} = \frac{t^3}{12} \underset{\sim}{C}$$

$\underset{\sim}{C}$  : matriz de constantes elásticas do item 3.9.4.

Portanto, desenvolvendo-se a integração mencionada, obtém-se como contribuição do sub-triângulo "i" a seguinte expressão :

$$\underset{\sim}{r}^{[i]} = \underset{\sim}{T}^{[i]T} \underset{\sim}{G}^{[i]} \underset{\sim}{T}^{[i]} \quad \dots(3.169)$$

A matriz  $G^{[i]}$  é quadrada de ordem 9, simétrica e formada por sub-matrizes de ordem 3,  $G_{jk}^{[i]}$ , cuja expressão resulta :

$$G_{jk}^{[i]} = \frac{A_i t^3}{144} C_{jk} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad \dots(3.170)$$

onde, além dos parâmetros já vistos,  $A_i$  é a área do sub-triângulo "i".

Desse modo, a matriz de rigidez do elemento triangular completo pode ser escrita pela simples adição das contribuições de cada sub-elemento, ou seja :

$$\tilde{r} = \tilde{r}^{[1]} + \tilde{r}^{[2]} + \tilde{r}^{[3]} \quad \dots(3.171)$$

### 3.9.6.7. Vetor de Cargas

O único carregamento a ser levado aos nós do elemento de forma consistente é o gradiente de temperatura segundo o eixo normal ao plano do elemento. A expressão utilizada para esse procedimento é a que se segue :

$$\tilde{p} = - \int_A \tilde{B}^T \tilde{m}_0 dA \quad \dots(3.172)$$

onde,

$$\tilde{B} = \phi \tilde{T}_{\tilde{n}}$$

$$\tilde{m}_0 = D \tilde{\kappa}_0$$

É interessante ressaltar que esse cálculo será feito para cada sub-elemento, somando-se as contribuições para encontrar-se o vetor de cargas do elemento completo.

O vetor  $\tilde{\kappa}_0$  representa o campo de curvaturas iniciais, provocado pelo gradiente  $T_g$ . Essas curvaturas podem ser encontradas simplesmente pela multiplicação do gradiente fornecido pelo coeficiente de dilatação térmica correspondente. Assim sendo, tem-se :

$$\tilde{\kappa}_0 = T_g \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ 0 \end{Bmatrix} \quad \dots(3.173)$$

Então o momento  $\tilde{m}_0$  resulta :

$$\tilde{m}_0 = \frac{T_g t^3}{12} \begin{Bmatrix} C_{11}\alpha_1 + C_{12}\alpha_2 \\ C_{12}\alpha_1 + C_{22}\alpha_2 \\ 0 \end{Bmatrix} \quad \dots(3.174)$$

Portanto, na equação 3.172, a única matriz não

constante é a  $\phi$ . Logo, pode-se escrever :

$$\tilde{p}^{(i)} = - \tilde{T}^{(i)T} \left[ \int_{A_i} \phi^T dA_i \right] \tilde{m}_0 \quad \dots(3.175)$$

onde,

$$\int_{A_i} \phi^T dA_i = \frac{A_i}{3} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

### 3.9.6.8. Matriz da Relação Momentos / Deslocamentos

Para a determinação da matriz de relação momentos com deslocamentos será necessário trabalhar por sub-triângulos. Determinar-se-á, para cada um desses sub-elementos, o vetor dos momentos no centróide. Para o cálculo em relação ao elemento completo, basta realizar uma média dos valores determinados.

Considerando-se as expressões 3.166, 3.167 e 3.168, pode-se escrever :

$$\tilde{m}^{(i)} = \tilde{D} \tilde{\phi} \tilde{T}^{(i)} \tilde{u} \quad \dots(3.176)$$

onde,

$$\tilde{t}_d^{(i)} = \tilde{D} \tilde{\phi} \tilde{T}^{(i)}$$

Ressalta-se que na matriz  $\phi$ , os valores de  $h_1$ ,  $h_2$  e  $h_3$  são os dos centróide, ou seja  $2A_i/3$ . Portanto, a matriz para o elemento completo resulta :

$$\tilde{t}_d = \frac{1}{3} \left[ \tilde{t}_d^{[1]} + \tilde{t}_d^{[2]} + \tilde{t}_d^{[3]} \right] \quad \dots(3.177)$$



### 3.9.6.9. Vetor de Momentos Iniciais

O vetor de momentos iniciais somente receberá contribuição do gradiente de temperatura. Sua determinação é trivial, podendo-se escrever :

$$\tilde{t}_i = - \tilde{m}_o \quad \dots(3.178)$$

onde  $\tilde{m}_o$  é fornecido pela equação 3.174.

### 3.9.7. Matrizes e Vetores em Coordenadas Globais

Após a montagem dos elementos triangulares de membrana e placa, ou apenas placa se esse for o caso, é necessário verificar se o elemento definido pelo usuário era triangular ou quadrangular.

No caso de triângulo, o procedimento é um pouco mais simples, bastando ao programa transformar as matrizes e vetores desenvolvidos em relação às coordenadas locais para as coordenadas globais. Essas coordenadas globais são as mesmas já consideradas, por exemplo, para o elemento tipo 2. No caso genérico são três translações e três rotações para cada nó do elemento. Neste ponto talvez seja interessante ressaltar que o elemento casca tem efetivamente apenas cinco graus de liberdade. Entretanto, para efeito de compatibilização com o restante do programa é como se fossem considerados seis graus de liberdade. No caso a rotação segundo o eixo  $x_3$  fica sem contribuição.

A transformação das matrizes e vetores escritos em relação às coordenadas locais para as coordenadas globais é feita por meio da mesma matriz  $\lambda$  já mencionada em outros elementos. No presente caso, considerando-se  $l_i$ ,  $m_i$  e  $n_i$ , respectivamente cossenos diretores do eixo local  $x_i$  em

relação aos eixos globais  $X_1$ ,  $X_2$  e  $X_3$  pode-se escrever a matriz de transformação para os elementos definidos com três nós :

$$\lambda_{\sim} = \begin{bmatrix} \lambda_0 & 0 & 0 \\ \sim & \sim & \sim \\ 0 & \lambda_0 & 0 \\ \sim & \sim & \sim \\ 0 & 0 & \lambda_0 \\ \sim & \sim & \sim \end{bmatrix} \quad \dots(3.179)$$

onde,

$$\lambda_0_{\sim} = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix}$$

A matriz de transformação será utilizada para alterar a matriz de rigidez segundo a expressão 3.17. Quanto ao vetor de cargas, a equação a ser utilizada é a 3.33. Por fim, no caso da matriz de relação momentos/deslocamentos a equação será a seguinte :

$$TD_{\sim} = td_{\sim} \lambda_{\sim} \quad \dots(3.180)$$

Menciona-se ainda que o vetor de momentos iniciais não necessita de alteração.

Caso o elemento considerado seja um quadrilátero, é necessário que o nó central, definido pelo programa no centróide do elemento, seja eliminado antes da transformação de suas matrizes e vetores para as coordenadas globais. Essa eliminação é feita de forma muito simples através de uma condensação estática, [3.11], procedimento já discutido para os elementos tipos 4,5 e 6.

Naqueles casos, interessava a eliminação dos modos incompatíveis que tinham sido introduzidos nas funções de interpolação. Aqui interessa a eliminação de graus de liberdade de um nó criado pelo programa. Entretanto, do ponto de vista matemático, a situação é rigorosamente a

mesma, assim como o será o procedimento utilizado nessa eliminação. Basta que se faça uma triangularização parcial das matrizes de rigidez e momentos/deformações e do vetor de carga, eliminando-se as seis últimas linhas, que correspondem ao quinto nó criado.

Depois da eliminação, a transformação para as coordenadas globais será realizada através da seguinte matriz  $\lambda$  :

$$\lambda = \begin{bmatrix} \lambda_0 & 0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & \lambda_0 & 0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & \lambda_0 & 0 \\ \sim & \sim & \sim & \sim \\ 0 & 0 & 0 & \lambda_0 \\ \sim & \sim & \sim & \sim \end{bmatrix} \quad \dots(3.181)$$

onde  $\lambda_0$  é a mesma submatriz da expressão 3.179.

### 3.9.8. Desempenho

#### 3.9.8.1. Considerações Sobre a Convergência de Resultados

Para o presente elemento, na verdade uma composição de um elemento membrana e um elemento placa, a convergência dos resultados deve considerar separadamente a convergência de cada componente.

Entretanto, para o componente membrana, alguns estudos já foram realizados quando do desenvolvimento do elemento tipo 4. Isso porque o teste de convergência lá realizado, quando considerou-se a discretização em elementos triangulares sem a inclusão de modos incompatíveis, é exatamente o que seria aqui considerado, posto que os elementos triangulares são os mesmos. Apenas a ressaltar

que, neste caso, quando consideram-se elementos quadrangulares isso equivale a definir quatro triângulos, conforme está apresentado na figura 3.56b.

Já para o componente placa, elemento LCCT-9 isolado ou composto em quadriláteros, nenhum estudo foi ainda realizado neste trabalho, sendo o que se fará no próximo sub-item.

### 3.9.8.2. Convergência do Componente Placa

Para se examinar a convergência do elemento placa aqui desenvolvido foram considerados quatro casos clássicos : placa apoiada com carga distribuída, placa engastada com carga distribuída, placa apoiada com carga concentrada e placa engastada com carga concentrada. Devido à dupla simetria, foi discretizado apenas o quarto inferior esquerdo da placa completa.

O parâmetro controlado foi o deslocamento central e as características de dimensões, valor do carregamento e parâmetros elásticos adotados não interessam diretamente pois serão apresentados valores de coeficientes adimensionais  $\alpha$  e  $\beta$ , que podem ser definidos por :

$$\alpha = \frac{w D}{q l^4} \quad \dots(3.182)$$

$$\beta = \frac{w D}{P l^4}$$

onde,

$\alpha, \beta$  : parâmetros adimensionais, respectivamente para carga distribuída e concentrada.

w : deslocamento do ponto central.

D : fator de rigidez à flexão da placa.

q, P : cargas, respectivamente distribuída e concentrada.

l : dimensão do quarto de placa discretizado.

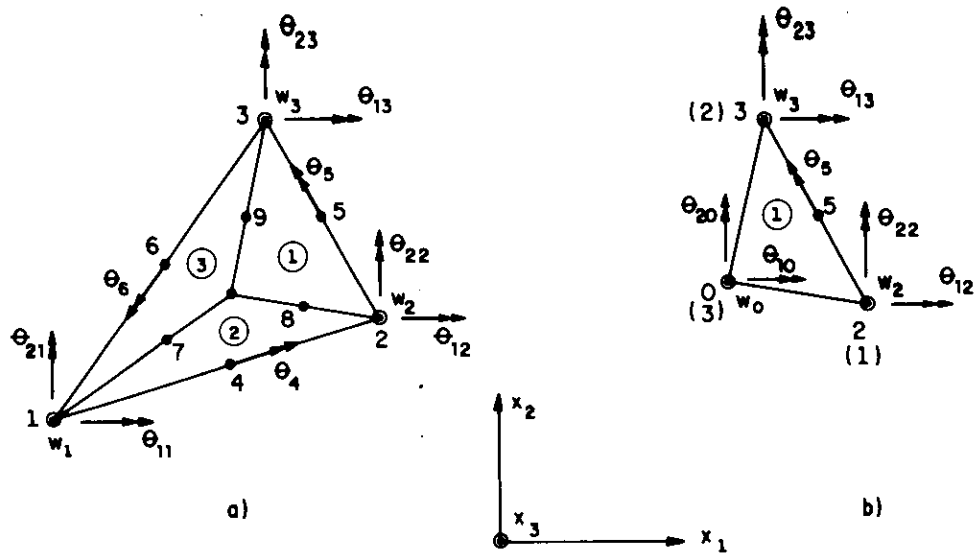


fig. 3.61 Deslocamentos e rotações no elemento placa

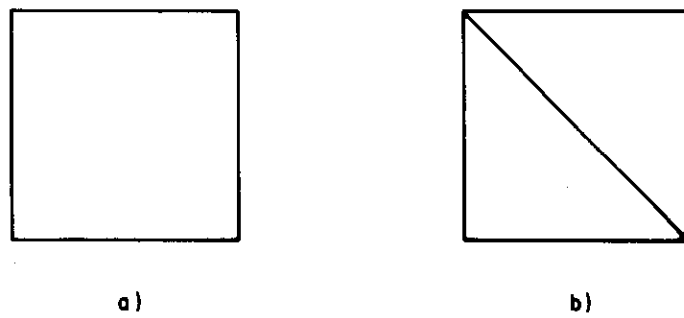


fig. 3.62 Discretizações 1 e 4

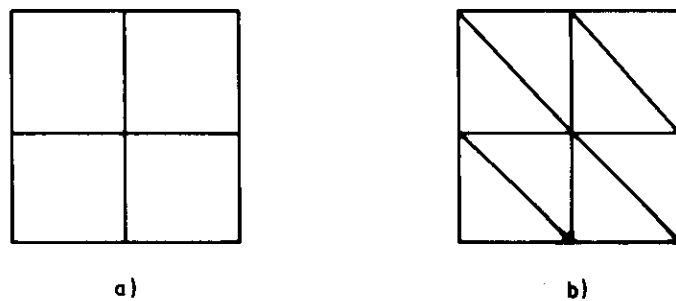


fig. 3.63 Discretizações 2 e 5

Para um bom acompanhamento da convergência, foram consideradas seis discretizações :

Discretização 1 : 1 elemento retangular ( fig 3.62a ).

Discretização 2 : 4 elementos retangulares ( fig 3.63a ).

Discretização 3 : 16 elementos retangulares ( fig 3.64a ).

Discretização 4 : 2 elementos triangulares ( fig 3.62b ).

Discretização 5 : 8 elementos triangulares ( fig 3.63b ).

Discretização 6 : 32 elementos triangulares ( fig 3.64b ).

Os resultados encontram-se organizados nas tabelas 3.13 a 3.16. Nelas são apresentados os valores obtidos para os diversos exemplos considerados e ainda o valor da solução exata, para efeito de comparação.

Pela observação desses valores pode-se perceber que o elemento considerado apresenta sempre uma convergência perfeitamente monotônica. Além disso, o quadrilátero mostra uma convergência relativamente rápida para o resultado exato. Já os triângulos apresentam uma convergência mais lenta, necessitando de redes mais refinadas para a obtenção de bons resultados.

De qualquer maneira, tomando-se a discretização 2, que pode ser considerada relativamente pobre, verifica-se que no pior dos casos o quadrilátero afasta-se apenas 8.5 % do valor exato. Já para os dois elementos triangulares a situação é sensivelmente pior, situando-se essa diferença na casa dos 30 % para a discretização 5.

Placa apoiada sob carga distribuída						
Disc.	1	2	3	4	5	6
$10^3 \alpha$	3.951	3.935	4.035	3.440	3.915	4.025
Exato	4.060					

tabela 3.13

Placa engastada sob carga distribuída						
Disc.	1	2	3	4	5	6
$10^3 \alpha$	1.130	1.166	1.233	0.185	1.007	1.194
Exato	1.265					

tabela 3.14

Placa apoiada sob carga concentrada						
Disc.	1	2	3	4	5	6
$10^3 \alpha$	10.802	10.963	11.391	8.820	10.483	11.230
Exato	11.600					

tabela 3.15

Placa engastada sob carga concentrada						
Disc.	1	2	3	4	5	6
$10^3 \alpha$	4.519	5.154	5.399	1.008	4.215	5.159
Exato	5.600					

tabela 3.16

### 3.9.8.3. Tempo de Processamento

Quanto ao desempenho em relação aos tempos de processamento, o elemento quadrangular <sup>iv</sup>gasta 1.575 seg para

a montagem e gravação das suas matrizes e vetores. Para o elemento triangular esse tempo reduz-se a 0.45 seg.

Já no cálculo das tensões, o quadrangular gasta 0.055 seg e o triângulo puro 0.04 seg. É importante ressaltar que os tempos fornecidos foram verificados para elementos onde consideraram-se as rigidezes de membrana e placa. No caso de elementos exclusivamente placa, observa-se uma redução de aproximadamente 20 % no tempo de montagem e gravação das matrizes e vetores.

### 3.9.9. Exemplos

#### 3.9.9.1. Pavimento de Edifício em Laje Cogumelo

O pavimento de edifício aqui considerado é do Edifício Maison Mouette, SP, e sua planta de forma é mostrada na figura 4.28, no capítulo seguinte. Trata-se de uma laje cogumelo que possui algumas vigas para travamento quanto a esforços horizontais. A rede utilizada para sua discretização é feita por elementos placa e barra e encontra-se apresentada na figura 3.65.

Como dados adicionais para o problema, mencionam-se :

$e$  = espessura da laje = 0.15 m

$E$  =  $1.70 \times 10^7$  KN/m<sup>2</sup>

$\nu$  = 0.20

Ressalta-se ainda que o valor do módulo de elasticidade transversal  $G$  foi fornecido com seu valor multiplicado por  $10^{-3}$ , de modo a praticamente eliminarem-se os momentos volventes. Já o carregamento atuante é o tradicional para pavimentos desse tipo, compreendendo cargas permanentes e acidentais definidas por procedimentos



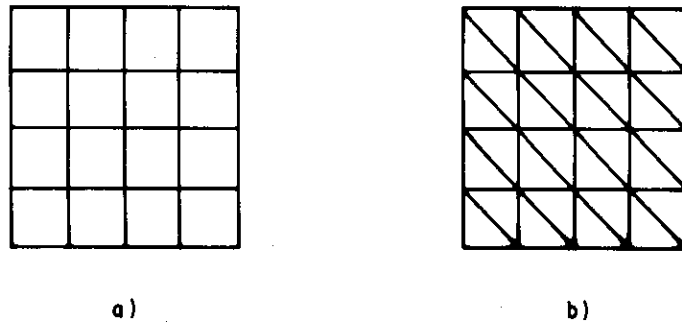


fig. 3.64 Discretizações 3 e 6

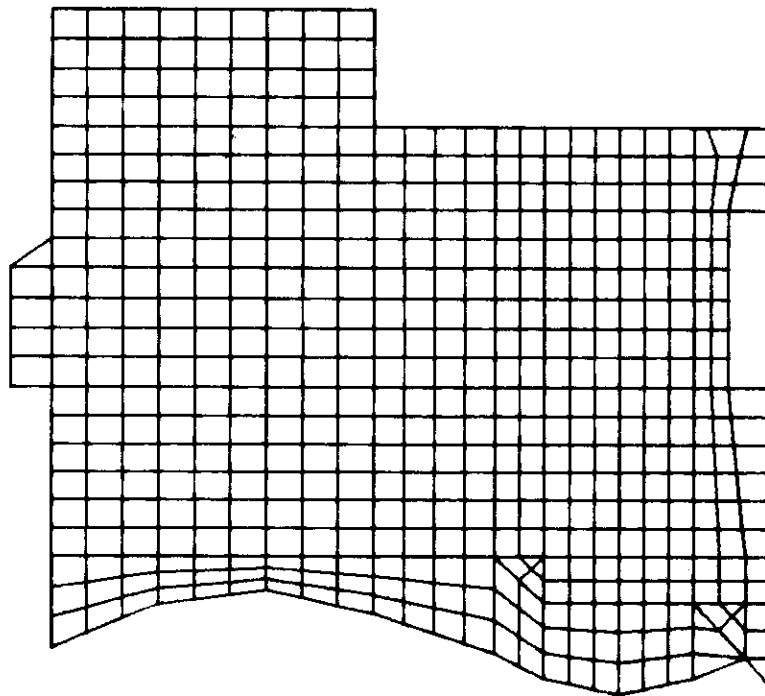


fig. 3.65 Discretização do pavimento

convencionais.

Como resultados, são apresentadas nas figuras 3.66 e 3.67 as curvas de isomomentos, em  $\text{KN} \times \text{m} / \text{m}$  e segundo os eixos locais  $x_1$  e  $x_2$ , respectivamente.

É interessante de se observar por essas curvas que a existência de pilares e vigas destaca as regiões que num procedimento de cálculo tradicional seriam consideradas como lajes independentes. Entretanto, em outros pontos do pavimento, verifica-se uma situação mais complexa, podendo-se prever dificuldades para o dimensionamento que se baseasse nessas hipóteses tradicionais.

#### 3.9.9.2. Parabolóide Hiperbólico

Para este exemplo, onde demonstra-se a utilização de elementos para análise de cascas, foi escolhido um parabolóide hiperbólico em concreto armado. Trata-se de uma estrutura que foi projetada para a orla marítima de Vitória, ES, servindo de cobertura para bares e lanchonetes.

Suas dimensões são apresentadas na figura 3.68. Já para a discretização da estrutura, mostrada em perspectiva na figura 3.69, foram utilizados 256 elementos casca. Como dados adicionais do cálculo, podem ser citados :

$$e = \text{espessura da casca} = 0.08 \text{ m}$$

$$E = 1.70 \times 10^7 \text{ KN/m}^2$$

$$\nu = 0.20$$

$$G = 0.71 \times 10^7 \text{ KN/m}^2$$

O carregamento para este exemplo é composto apenas pelo peso próprio mais uma carga uniformemente distribuída, segundo o eixo  $x_3$ , de  $1 \text{ KN/m}^2$ .

Como resultados, apresentam-se nas figuras 3.70 e 3.71 as tensões normais, respectivamente segundo os eixos locais

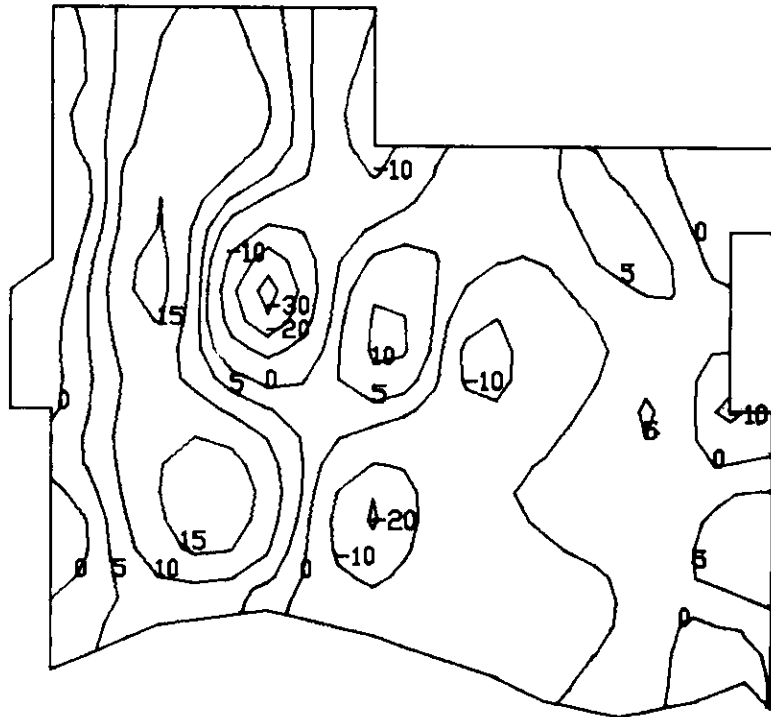


fig. 3.66 Momentos fletores segundo  $X_1$

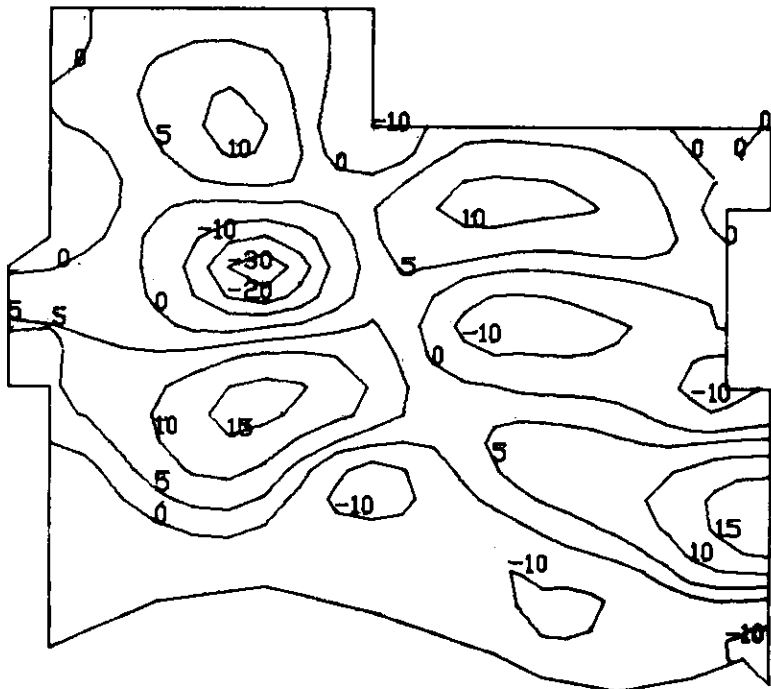


fig. 3.67 Momentos fletores segundo  $X_2$

$x_1$  e  $x_2$ , e na figura 3.72 o momento fletor, segundo o eixo local  $x_1$ . Ressalta-se que as tensões são fornecidas em MPa e o momento fletor em KN x m / m.

Pela observação dos resultados apresentados, verifica-se que realmente a maior solicitação para a estrutura diz respeito às tensões normais, indicando que o comportamento de membrana é preponderante. Isso é bastante interessante pois para uma estrutura de concreto armado a predominância das tensões de compressão é um fato benéfico. Além disso, percebe-se que a simplificação do cálculo através de expressões que consideram apenas o comportamento de membrana não foge muito da realidade.

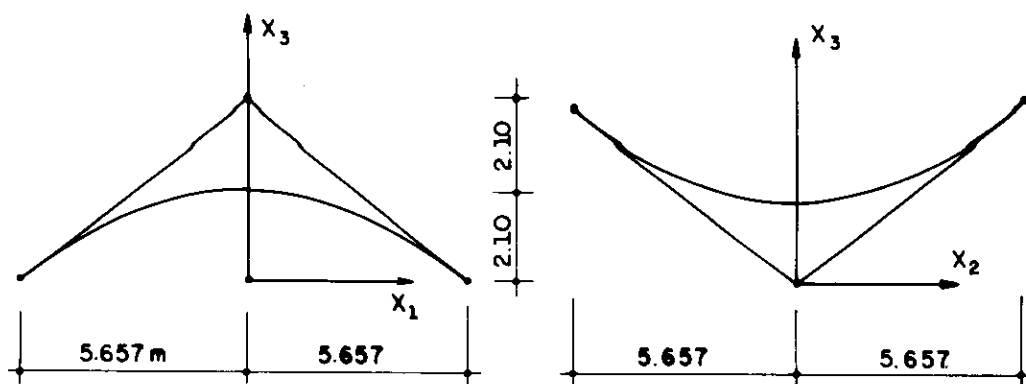


fig. 3.68 Parabolóide hiperbólico

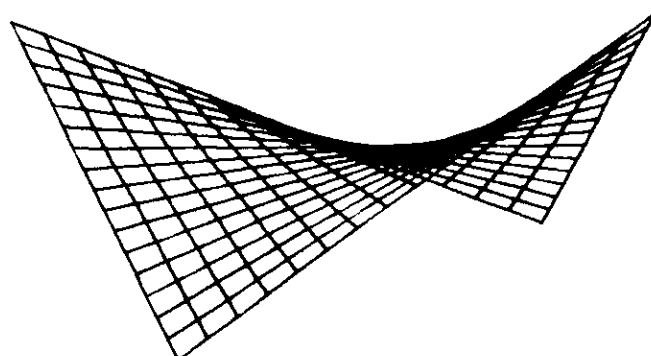


fig. 3.69 Discretização da casca

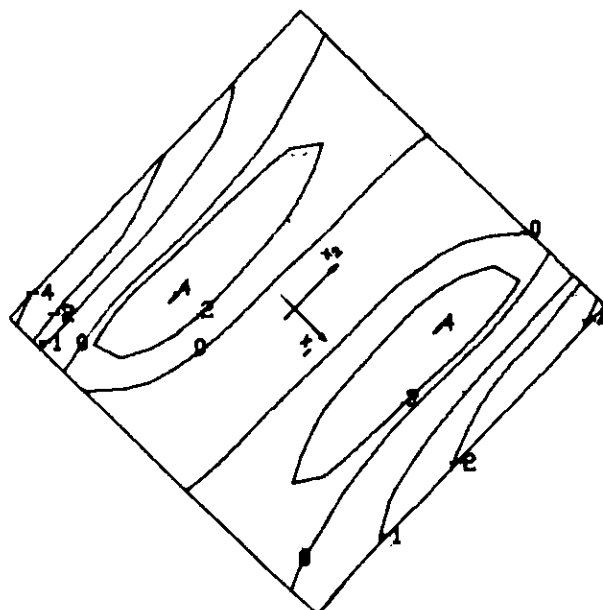


fig. 3.70 Tensão normal segundo eixo local  $x_1$

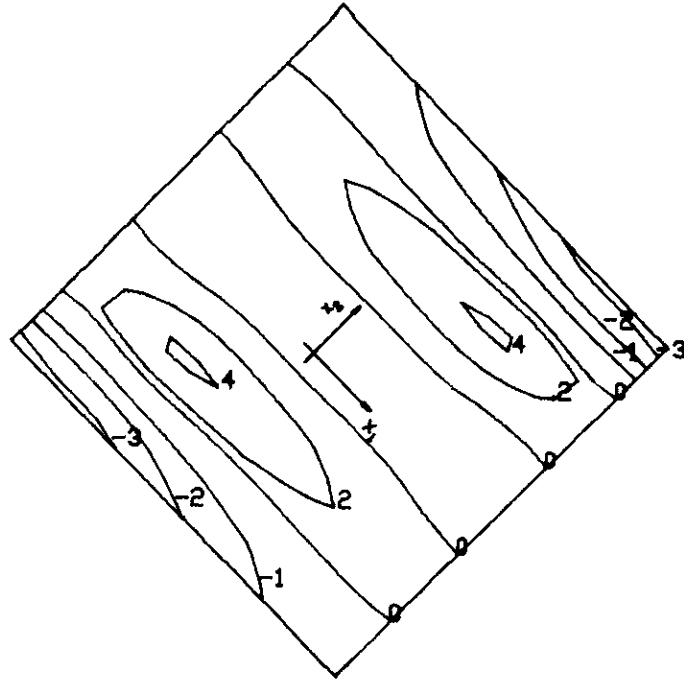


fig. 3.71 Tensão normal segundo eixo local  $x_2$

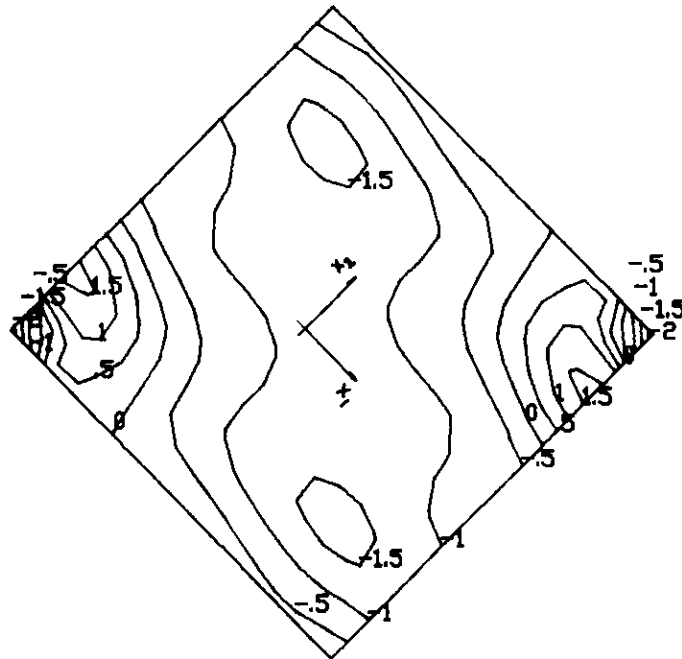


fig. 3.72 Momento fletor segundo eixo local  $x_1$

### 3.10 REFERÊNCIAS BIBLIOGRÁFICAS

- [3.1] Meek, J.L.; "Matrix Structural Analysis"; McGraw-Hill Kogakusha, Ltd; Tokio; 1971.
- [3.2] Przemieniecki, J.S.; "Theory Of Matrix Structural Analysis"; Ed. McGraw-Hill Kogakusha, Ltd; Tokio; 1971.
- [3.3] Gere, J.M.; Weaver, W.; "Análise de Estruturas Reticuladas"; Ed. Guanabara Dois; Rio de Janeiro; 1981.
- [3.4] Moreira, D.F.; "Análise Matricial de Estruturas"; Ed. da Universidade de São Paulo; S.Paulo; 1977.
- [3.5] Schiel, F.; "Introdução à Resistência dos Materiais"; Ed. Harper & Row do Brasil; S. Paulo; 1984.
- [3.6] Bathe, K; "Finite Element Procedures in Engineering Analysis"; Ed. Prentice-Hall, Inc.; Englewood Clifs, N.J.; 1982.
- [3.7] Brebbia, C.A.; Connor, J.J; "Metodos de los Elementos Finitos en la Ingenieria Civil"; Ed. Colegio Oficial de Ingenieros de Caminos, Canales y Puertos; Madrid; 1975.
- [3.8] Zienkiewicz, O.C.; "The Finite Element Method"; Ed. Mc-Graw-Hill; London; 1985.
- [3.9] Hinton, E.; Owen, D.R.J.; "Finite Element Programming"; Ed. Academic Press; London; 1977.
- [3.10] Strang, G.; Fix, G.; "An Analysis of the Finite Element Method"; Ed. Prentice-Hall, Englewood Clifs, N.J.; 1973.

[3.11] Rubinstein, M.F.; Rosen, R.; "Substructure Analysis by Matrix Decomposition"; Journal of Structural Division, Proc. ASCE; New York; 1970.

[3.12] Jaeger, C.; "Rock Mechanics and Engineering"; Ed. Cambridge University Press; Cambridge; 1972.

[3.13] "Numerical and Computer Methods in Structural Mechanics"; Ed. S. Fennes, N. Perrone, A. Robinson e W. Schnobrich; Academic Press; London; 1973.

[3.14] Clough, R.W.; Felippa, C.A.; "A Refined Quadrilateral Element for Analysis of Plate Bending"; Proc. 2nd Conference on Matrix Methods in Structural Mechanics; Wright Peterson Air Force Base, Ohio; 1968.

[3.15] Clough, R.W.; Tocher, J.L.; "Finite Element Stiffness Matrices for the Analysis of Plate Bending"; Proc. Conference on Matrix Methods in Structural Mechanics; Wright Peterson Air Force Base, Ohio; 1968.



## CAPÍTULO 4

### Elemento Sapata Rígida

#### 4.1. INTRODUÇÃO

O objetivo deste capítulo é fornecer detalhes a respeito da formulação do elemento tipo 8, mencionado no capítulo anterior com a denominação de "elemento sapata".

Inicialmente é apresentado um breve resumo do Método dos Elementos de Contorno, que serve de base ao desenvolvimento do elemento mencionado. Esse resumo inicia-se com as equações governantes da teoria da elasticidade para meios homogêneos e seu tratamento para a formulação da equação de Navier. A seguir são apresentadas algumas soluções fundamentais para casos de domínios tridimensionais. Finalmente são deduzidas equações integrais para pontos do contorno e do interior do domínio.

Quanto à formulação do elemento propriamente dito, inicialmente são discutidos alguns detalhes sobre sua definição, como por exemplo, forma e função aproximadora adotada. Logo após é apresentada a dedução das matrizes e vetores envolvidos na discretização dos domínios considerados. Então, após o cálculo da matriz de rigidez das sapatas, são estudados alguns últimos detalhes, como translação e simetrização desses valores.

Depois, são discutidos resultados obtidos com o elemento aqui desenvolvido, abrangendo aspectos como : número de pontos de Gauss necessários às integrações, tempos de processamento, interação entre sapatas e influência da consideração da flexibilidade da fundação em algumas superestruturas de edifícios.

Para finalizar o capítulo, são apresentadas algumas breves conclusões gerais a respeito dos tópicos discutidos.

## 4.2. MÉTODO DOS ELEMENTOS DE CONTORNO

### 4.2.1. Teoria da Elasticidade - Equações Governantes

Pode-se supor um corpo  $\Omega + \Gamma$ , onde  $\Omega$  é uma região tridimensional aberta e  $\Gamma$  é seu contorno, conforme apresenta-se na figura 4.1. O material que constitui esse corpo, suposto elástico linear e isotrópico, pode ser definido através de duas constantes independentes: E, módulo de elasticidade, e  $\nu$ , coeficiente de Poisson. Para simplificar o equacionamento, define-se mais duas constantes: G, módulo de elasticidade transversal, e  $\lambda$ , constante de Lamé, que podem ser escritas em função das duas primeiras da maneira que se segue :

$$G = \frac{E}{2(1+\nu)} \quad \dots(4.1)$$

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \quad \dots(4.2)$$

Visando a obtenção de expressões mais compactas, adotou-se neste item a notação indicial [4.1]. Por questão de clareza, define-se a seguir a convenção utilizada para a derivação e somatório:

$$f_{i,j} = \frac{\partial f_i}{\partial x_j}$$

$$a_i b_i = a_1 b_1 + a_2 b_2 + a_3 b_3$$

A partir dessa introdução, inicia-se o equacionamento proposto [4.2], apresentando-se as equações de equilíbrio:

$$\sigma_{ij,j} + b_i = 0 \quad \dots(4.3)$$

onde  $\sigma_{ij}$  são componentes de tensão e  $b_i$  forças de volume.

As relações entre os deslocamentos e as deformações específicas podem ser escritas como :

$$\epsilon_{ij} = \frac{1}{2} \left( u_{i,j} + u_{j,i} \right) \quad \dots(4.4)$$

onde  $\epsilon_{ij}$  são as deformações específicas e  $u_i$  os deslocamentos.

Já a lei de Hooke, relação entre as tensões e deformações, resulta :

$$\sigma_{ij} = \lambda \epsilon_{kk} \delta_{ij} + 2G \epsilon_{ij} \quad \dots(4.5)$$

onde, além das grandezas já mencionadas, destaca-se o delta de Kronecker,  $\delta_{ij}$ , cujo valor é zero se  $i \neq j$  e 1 se  $i=j$ .

As condições de contorno para um ponto  $Q \in \Gamma$  podem ser escritas como :

$$u_i(Q) = \bar{u}_i(Q) \quad \dots(4.6)$$

$$p_i(Q) = \bar{p}_i(Q) \quad \dots(4.7)$$

onde  $\bar{u}_i$  e  $\bar{p}_i$  são as componentes de deslocamentos e forças de superfície prescritas no contorno. Ressalta-se ainda que  $p_i = \sigma_{ij} n_j$ , onde  $n_j$  são os cossenos diretores da normal ao plano tangente à superfície no ponto  $Q$  considerado, conforme mostra-se na figura 4.2.

O conjunto de equações apresentado, 4.3 a 4.7, define perfeitamente um problema elástico linear. Pode-se continuar o equacionamento proposto pela simples combinação das expressões já apresentadas. Usando-se a equação 4.4, relação deformação/deslocamento, a lei de Hooke e a relação  $p_i = \sigma_{ij} n_j$ , pode-se escrever as tensões e forças de

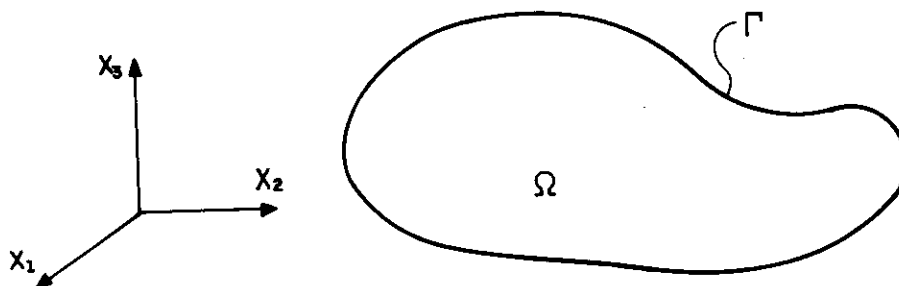


fig. 4.1 Corpo de volume  $\Omega$  e contorno  $\Gamma$

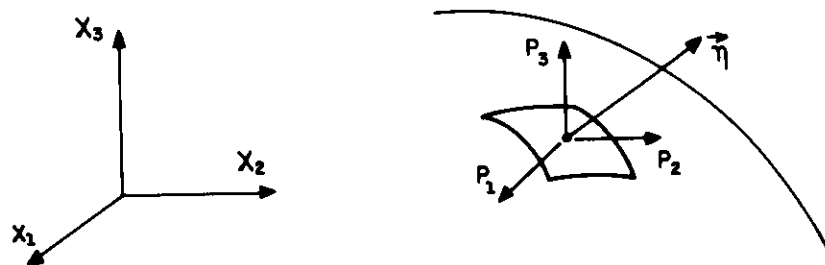


fig. 4.2 Componentes das forças de superfície

superfície em termos dos deslocamentos:

$$\sigma_{ij} = \lambda u_{k,k} \delta_{ij} + G \left( u_{i,j} + u_{j,i} \right) \quad \dots(4.8)$$

$$p_i = G \left( u_{i,n} + u_{j,i} n_j + \frac{2\nu}{1-2\nu} u_{j,j} n_i \right) \quad \dots(4.9)$$

onde  $u_{i,n} = u_{i,j} n_j$ .

Substituindo-se a relação tensão/deformação, equação 4.5, na equação de equilíbrio 4.3 e depois utilizando-se a relação 4.4, pode-se obter a chamada equação de Navier :

$$u_{j,ii} + \frac{1}{1-2\nu} u_{i,ij} + \frac{1}{G} b_j = 0 \quad \dots(4.10)$$

Note-se que a solução da equação 4.10, equação diferencial do problema elástico linear em termos de deslocamentos, permite o cálculo de todos os parâmetros importantes para o caso de domínio tridimensional. Depois de calculados os deslocamentos, pode-se determinar as componentes de tensões pela equação 4.8, forças de superfície pela 4.9 e até mesmo deformações específicas pela 4.4.

Evidentemente a solução analítica da equação 4.10 só é possível para alguns casos particulares. Para se contornar este problema é que foram desenvolvidos os métodos numéricos de solução, como o Método dos Elementos de Contorno. Os próximos itens irão explicitar esse processo.

## 4.2.2. Soluções Fundamentais

### 4.2.2.1. Equacionamento básico

Na formulação das equações integrais do problema

elástico é necessária a utilização de uma solução fundamental da equação diferencial 4.10 [4.3]. Considerando-se os pontos "s" e "q" pertencentes a um domínio  $\Omega^*$ , que contém o domínio  $\Omega$ , essa solução pode ser entendida fisicamente como as respostas em "q" pela aplicação de forças concentradas  $F_i^*$  em "s". As forças concentradas são supostas na direção dos eixos cartesianos de referência, conforme mostra-se na figura 4.3.

Note-se que para cada domínio  $\Omega^*$ , com condições de contorno próprias, deve-se obter uma solução fundamental. Por exemplo, para o domínio infinito, tem-se a solução de Kelvin, para o semi-infinito as soluções de Mindlin e Boussinesq-Cerruti, e assim por diante.

O problema pode ser matematicamente colocado através da substituição das forças de volume pela expressão :

$$b_j^*(q) = \Delta(s,q) F_j^*(s) \quad \dots(4.11)$$

onde  $\Delta(s,q)$ , Delta de Dirac, é definido como :

$$\begin{aligned} \Delta(s,q) &= 0 \quad \text{se } s \neq q \\ \Delta(s,q) &= \infty \quad \text{se } s \equiv q \\ \text{e } \int_V \Delta(s,q) dv &= 1 \end{aligned} \quad \dots(4.12)$$

Substituindo-se as relações 4.11 em 4.3, as equações de equilíbrio para o problema fundamental resultam :

$$\sigma_{ij,j}^* + \Delta(s,q) F_i^*(s) = 0 \quad \dots(4.13)$$

Para as relações deformação/deslocamento têm-se :

$$\epsilon_{ij}^* = \frac{1}{2} \left[ u_{i,j}^* + u_{j,i}^* \right] \quad \dots(4.14)$$

A lei de Hooke resulta :

$$\sigma_{ij}^* = \lambda \epsilon_{kk}^* \delta_{ij} + 2G \epsilon_{ij}^* \quad \dots(4.15)$$

Para as forças de superfície pode-se escrever :

$$p_i^* = G \left( u_{i,n}^* + u_{j,i}^* n_j + \frac{2\nu}{1-2\nu} u_{j,j}^* n_i \right) \quad \dots(4.16)$$

Já a equação de Navier do problema fundamental pode ser escrita como sendo :

$$u_{j,ii}^* + \frac{1}{1-2\nu} u_{i,ij}^* + \frac{1}{G} \Delta(s,q) F_j^* \quad \dots(4.17)$$

Neste trabalho os deslocamentos  $u_i^*$  e as forças de superfície  $p_i^*$  que acontecem no ponto "q" devidos às forças concentradas do ponto "s" serão apresentados na sua forma tensorial [4.4] :

$$u_i^*(q) = U_{ji}^*(s,q) F_j^*(s) \quad \dots(4.18)$$

$$p_i^*(q) = P_{ji}^*(s,q) F_j^*(s)$$

onde nos tensores  $U_{ji}^*$  e  $P_{ji}^*$ , o primeiro índice corresponde à direção da carga unitária e o segundo à componente do deslocamento ou força, conforme se mostra nas figuras 4.4. e 4.5.

Na forma matricial, o tensor dos deslocamentos fundamentais pode ser escrito como :

$$U_{ji}^*(s,q) = \begin{bmatrix} U_{11}^* & U_{12}^* & U_{13}^* \\ U_{21}^* & U_{22}^* & U_{23}^* \\ U_{31}^* & U_{32}^* & U_{33}^* \end{bmatrix}$$

Já o tensor das forças de superfície resulta :

$$P_{ji}^*(s,q) = \begin{bmatrix} P_{11}^* & P_{12}^* & P_{13}^* \\ P_{21}^* & P_{22}^* & P_{23}^* \\ P_{31}^* & P_{32}^* & P_{33}^* \end{bmatrix}$$

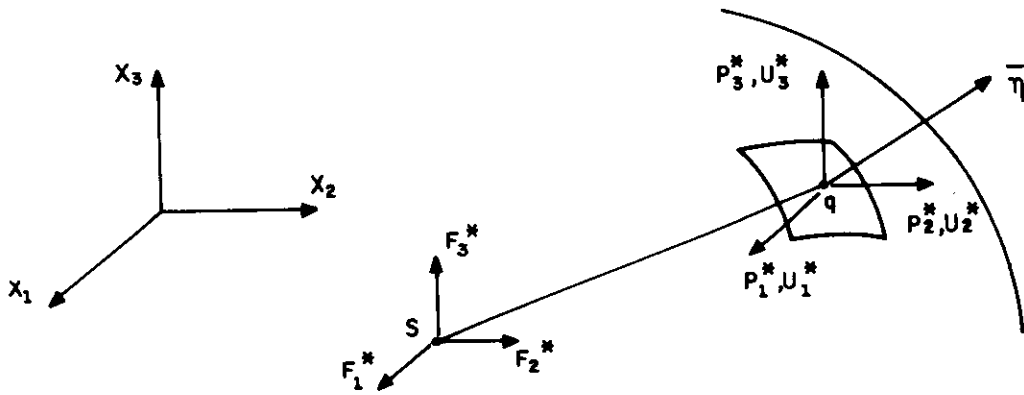


fig. 4.3 Respostas em "q" pela aplicação de  $F^*$  em "s"

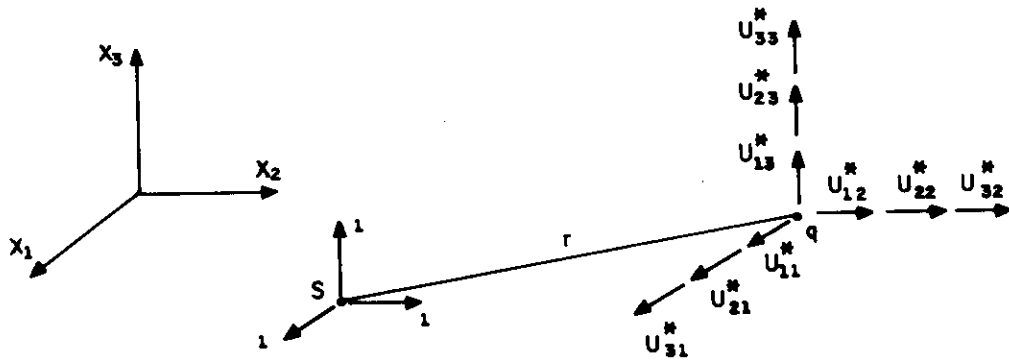


fig 4.4 Componentes do tensor de deslocamentos fundamentais

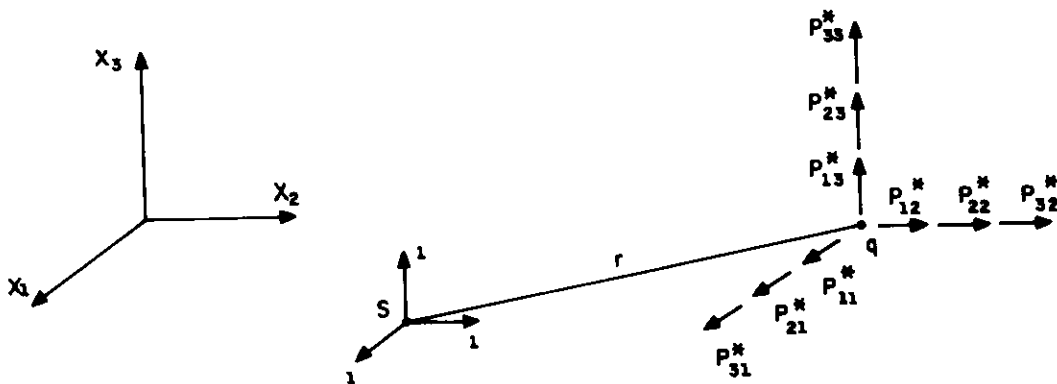


fig. 4.5 Componentes do tensor de forças de superfície fundamentais



Nos próximos itens serão discutidas as soluções particulares que se pode ter para esses tensores, em suma, soluções da equação 4.17. Mais especificamente para os interesses deste trabalho, essa discussão se limitará às soluções para domínios tridimensionais.

#### 4.2.2.2. Solução Fundamental de Kelvin

A clássica solução fundamental de Kelvin é obtida da resolução da equação 4.17 para um domínio tridimensional, elástico-linear, homogêneo e infinito [4.5]. As soluções para os deslocamentos e forças de superfície resultam :

$$\begin{aligned}
 U_{ij}^* &= \frac{1+\nu}{8\pi E(1-\nu)r} \left[ (3-4\nu)\delta_{ij} + r_{,i}r_{,j} \right] \\
 P_{ij}^* &= \frac{-1}{8\pi(1-\nu)r^2} \left\{ r_{,n} \left[ (1-2\nu)\delta_{ij} + 3r_{,i}r_{,j} \right] + \right. \\
 &\quad \left. - (1-2\nu) ( n_j r_{,i} - n_i r_{,j} ) \right\} \dots(4.19)
 \end{aligned}$$

onde  $r$  é a distância entre os pontos "s" e "q", sendo importante salientar que as derivações são relativas ao ponto "q", ou seja, ponto onde são observados os efeitos considerados. Esses parâmetros são mostrados na figura 4.6.

#### 4.2.2.3. Solução Fundamental de Mindlin

Para o domínio semi-infinito, ou seja quando existe um plano livre de tensões, nas mesmas condições de homogeneidade e elasticidade do item anterior, a solução foi dada por Mindlin [4.6]. O ponto "s", onde aplicam-se as cargas concentradas, estará mergulhado no domínio de uma

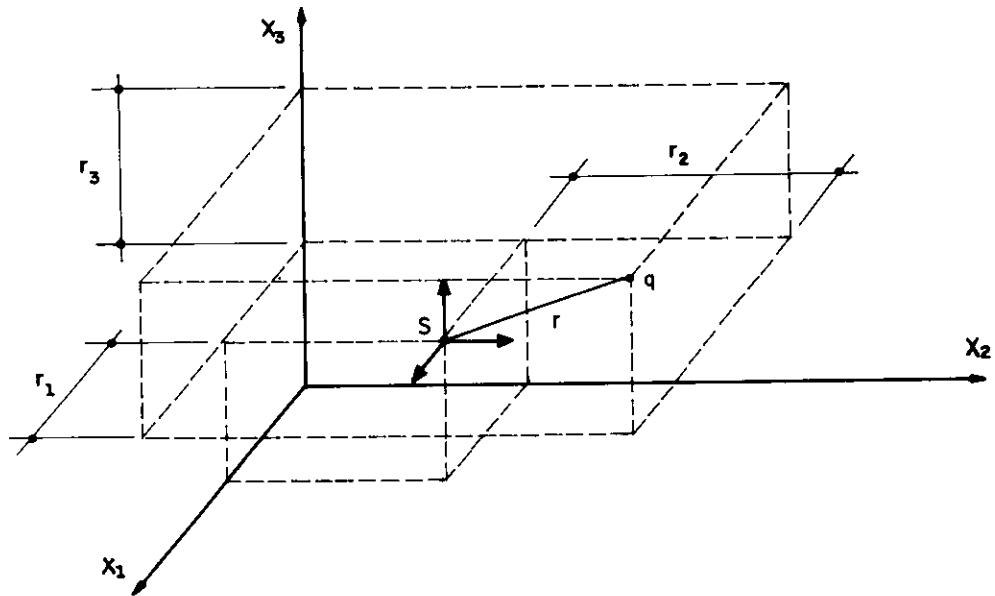


fig. 4.6 Espaço infinito de Kelvin

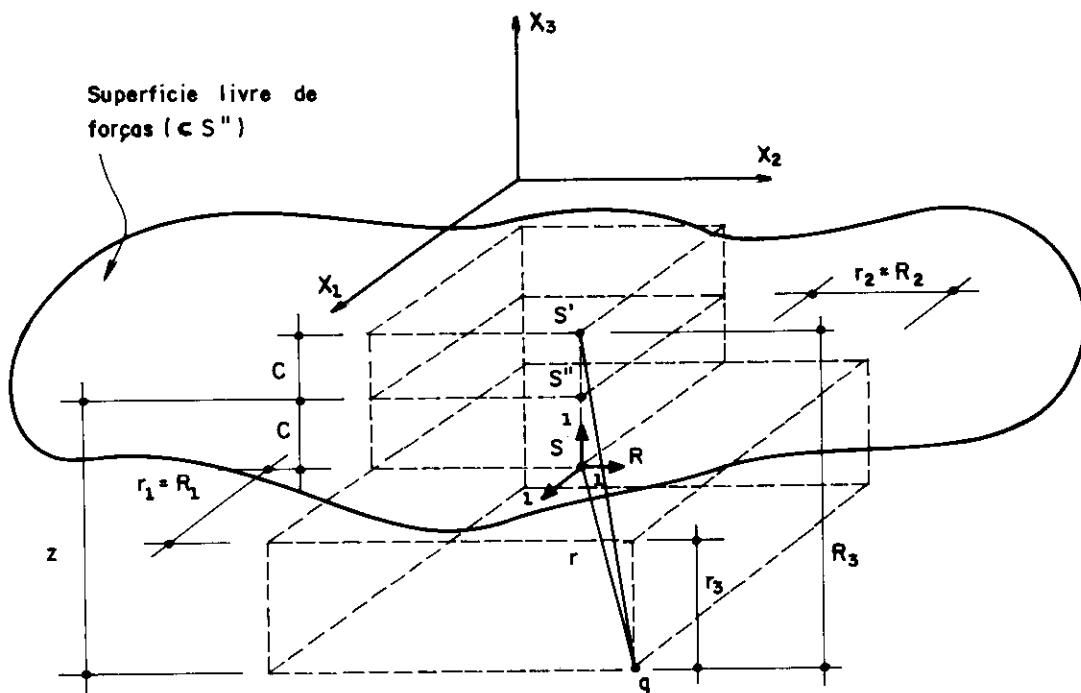


fig. 4.7 Espaço semi-infinito de Mindlin

distância  $C$ , conforme apresenta-se na figura 4.7. Nesse caso, as expressões para as componentes do tensor de deslocamentos podem ser escritas como:

$$\begin{aligned}
 U_{11}^* &= K_d \left\{ \frac{3-4\nu}{r} + \frac{1}{R} + \frac{r_1^2}{r^3} + \frac{(3-4\nu)r_1^2}{R^3} + \right. \\
 &\quad \left. + \frac{4(1-\nu)(1-2\nu)}{R+R_3} \left[ 1 - \frac{r_1^2}{R(R+R_3)} \right] \right\} \\
 U_{12}^* &= K_d r_1 r_2 \left[ \frac{1}{r^3} + \frac{3-4\nu}{R^3} - \frac{4(1-\nu)(1-2\nu)}{R(R+R_3)^2} \right] \\
 U_{13}^* &= -K_d r_1 \left[ \frac{r_3}{r^3} + \frac{(3-4\nu)r_3}{R^3} + \frac{4(1-\nu)(1-2\nu)}{R(R+R_3)} \right] \\
 U_{21}^* &= U_{12}^* \qquad \dots (4.20)
 \end{aligned}$$

$$\begin{aligned}
 U_{22}^* &= K_d \left\{ \frac{3-4\nu}{r} + \frac{1}{R} + \frac{r_2^2}{r^3} + \frac{(3-4\nu)r_2^2}{R^2} + \right. \\
 &\quad \left. + \frac{4(1-\nu)(1-2\nu)}{R+R_3} \left[ 1 - \frac{r_1^2}{R(R+R_3)} \right] \right\}
 \end{aligned}$$

$$U_{23}^* = \frac{r_2}{r_1} U_{13}^*$$

$$U_{31}^* = -K_d r_1 \left[ \frac{r_3}{r^3} + \frac{(3-4\nu)r_3}{R^3} - \frac{4(1-\nu)(1-2\nu)}{R(R+R_3)} \right]$$

$$U_{32}^* = \frac{r_2}{r_1} U_{31}^*$$

$$U_{33}^* = K_d \left[ \frac{3-4\nu}{r} + \frac{8(1-\nu)^2 - (3-4\nu)}{R} + \frac{r_3^2}{r^3} + \frac{(3-4\nu)R_3^2}{R^3} \right]$$

onde :

$$r_i = x_i(q) - x_i(s)$$

$$R_i = x_i(q) - x_i(s')$$

$$C = \frac{|x_3(s) - x_3(s')|}{2}$$

$$Z = |x_3(s'') - x_3(q)|$$

$$K_d = \frac{1+\nu}{8\pi E(1-\nu)}$$

$$K_s = \frac{1}{8\pi(1-\nu)}$$

Deve-se ressaltar que o tensor de forças de superfície não será apresentado pois, como se mostra no item 4.3.4, não tem importância prática para este trabalho. Assim sendo, essas expressões bastante extensas, encontradas na referência [4.6], serviriam apenas para alongar este resumo, que se pretende objetivo e breve.

#### 4.2.2.4. Solução Fundamental de Boussinesq-Cerruti

Caso o domínio considerado seja semi-infinito, portanto idêntico ao item anterior, mas o ponto "s" esteja na superfície livre de forças, a solução fundamental torna-se muito mais simples. Pode ser obtida das equações 4.20 quando se faz  $C = 0$ ,  $Z = 0$ ,  $r = R$  e  $r_3 = R_3$ , conforme mostra-se na figura 4.8.

Com essas condições, as expressões resultantes para os deslocamentos podem ser escritas como [4.7] :

$$U_{11}^* = K \left[ (1-\nu) + \nu r_{,1}^2 \right]$$

$$U_{12}^* = K \nu r_{,1} r_{,2}$$

$$U_{13}^* = -K (0.5-\nu) r_{,1}$$

$$U_{21}^* = U_{12}^*$$

$$U_{22}^* = K \left[ (1-\nu) + \nu r_{,2}^2 \right] \quad \dots(4.21)$$

$$U_{23}^* = -K (0.5-\nu) r_{,2}$$

$$U_{31}^* = -U_{13}^*$$

$$U_{32}^* = -U_{23}^*$$

$$U_{33}^* = K (1-\nu)$$

Também aqui o tensor das forças de superfície não será apresentado, devido ao fato já mencionado no item anterior.

#### 4.2.3. Equações Integrais para Pontos Interiores e do Contorno

Vai-se considerar um domínio elástico-linear, isotrópico e homogêneo  $\Omega$ , definido por um contorno  $\Gamma = \Gamma_1 + \Gamma_2$ , onde se desenvolvem estados de deslocamentos, deformações e tensões pela aplicação de determinadas ações. As representações integrais desses estados podem ser obtidas a partir de equações de resíduos ponderados [4.8]. Mais especificamente, o erro da equação 4.3, quando a solução exata for substituída por uma solução aproximada, permite escrever por ponderação em todo o domínio e imposição de deslocamentos e forças de superfície prescritos no contorno a seguinte expressão :

$$\int_{\Omega} (\sigma_{jk,j} + b_k) u_k^* d\Omega = \int_{\Gamma_1} (u_k - \bar{u}_k) p_k^* d\Gamma + \\ - \int_{\Gamma_2} (p_k - \bar{p}_k) u_k^* d\Gamma = 0 \quad \dots(4.22)$$

onde  $u_k^*$  e  $p_k^*$ , soluções fundamentais para deslocamentos e forças, são as funções ponderadoras. O contorno  $\Gamma_1$  refere-se à parte de  $\Gamma$  onde os deslocamentos são prescritos. Já no contorno  $\Gamma_2$  as forças de superfície é que o são.

A equação 4.22, após duas integrações por partes, resulta:

$$\int_{\Omega} b_k u_k^* d\Omega - \int_{\Omega} \sigma_{jk,j}^* u_k d\Omega = - \int_{\Gamma_1} p_k u_k^* d\Gamma - \int_{\Gamma_2} \bar{p}_k u_k^* d\Gamma + \\ + \int_{\Gamma_1} \bar{u}_k p_k^* d\Gamma + \int_{\Gamma_2} u_k p_k^* d\Gamma \quad \dots(4.23)$$

Pela soma das parcelas correspondentes, a equação 4.23 pode ser reescrita na forma :

$$\int_{\Omega} u_k^*(s,q) b_k(q) d\Omega(q) + \int_{\Gamma_1} u_k^*(s,Q) p_k(Q) d\Gamma(Q) = \\ - \int_{\Omega} \sigma_{jk,j}^*(s,q) u_k(q) d\Omega(q) + \int_{\Gamma} p_k^*(s,Q) u_k(q) d\Gamma(Q) \quad \dots(4.24)$$

onde "q" é um ponto do domínio  $\Omega$  e "Q" do contorno  $\Gamma$ .

Pela consideração da equação de equilíbrio 4.3, da distribuição Delta de Dirac e equações 4.18 com  $F_i^* = 1$ , a expressão 4.24 pode ser escrita como :

$$u_i(s) = - \int_{\Gamma} P_{ik}^*(s,Q) u_k(Q) d\Gamma(Q) + \int_{\Gamma} U_{ik}^*(s,Q) p_k(Q) d\Gamma(Q) + \\ + \int_{\Omega} U_{ik}^*(s,q) b_k(q) d\Omega(q) \quad \dots(4.25)$$

onde "s" e "q"  $\in \Omega$  e "Q"  $\in \Gamma$ .

A equação 4.25 é conhecida por Identidade Somigliana [4.9]. Ela fornece para o ponto interior "s" os deslocamentos, segundo os eixos de referência, dados pela integração dos valores dos deslocamentos e forças de superfície no contorno  $\Gamma$  e forças volumétricas no domínio  $\Omega$ .

Entretanto, para um equacionamento mais eficiente dos problemas a serem resolvidos, seria interessante escrever uma equação integral que relacionasse deslocamentos e forças de superfície apenas para pontos do contorno. Para conseguir-se esse intento, far-se-á uma ampliação do domínio original  $\Omega + \Gamma$ , considerando-se um acréscimo infinitesimal  $\Omega_\epsilon$ , de raio  $\epsilon$ , com contorno  $\Gamma_\epsilon$ . Dessa maneira, o ponto de contorno "S" passa a ser considerado como ponto interno [4.10], conforme apresenta-se na figura 4.9.

Para o novo domínio  $\Omega + \Omega_\epsilon$ , com contorno  $\Gamma - \bar{\Gamma} + \Gamma_\epsilon$ , pode-se reescrever a Identidade Somigliana, obtendo-se a representação integral dos deslocamentos da maneira como se segue:

$$\begin{aligned}
 u_i(S) = & - \int_{\Gamma - \bar{\Gamma}} P_{ik}^*(S, Q) u_k(Q) d\Gamma(Q) - \int_{\Gamma_\epsilon} P_{ik}^*(S, Q) u_k(Q) d\Gamma(Q) + \\
 & + \int_{\Gamma - \bar{\Gamma}} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q) + \int_{\Gamma_\epsilon} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q) + \\
 & + \int_{\Omega} U_{ik}^*(S, q) b_k(q) d\Omega(q) + \int_{\Omega} U_{ik}^*(S, q) b_k(q) d\Omega(q)
 \end{aligned}
 \dots(4.26)$$

Obviamente, após a extensão do domínio que visou transformar "S" em ponto interior, deve-se calcular o limite da equação 4.26 quando  $\epsilon$ , raio de  $\Omega_\epsilon$ , tende a zero. Para isso, serão analisados separadamente os 6 termos da

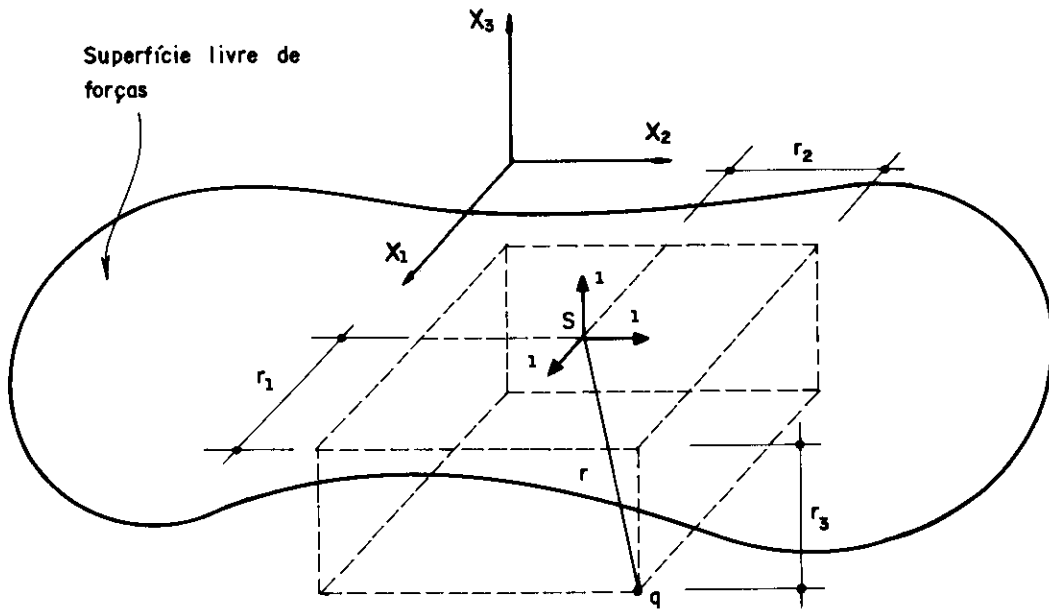


fig 4.8 Espaço semi-infinito de Boussinea-Cerruti

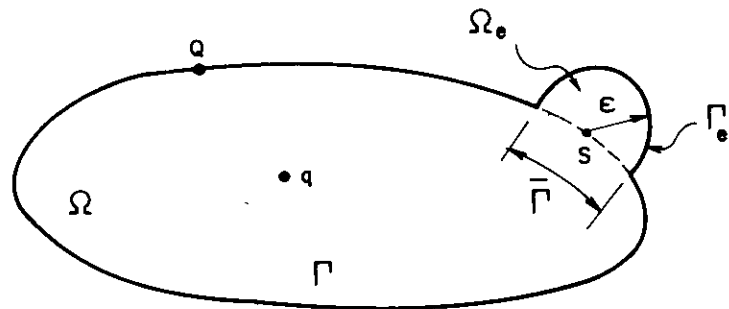


fig. 4.9 Domínio adicional  $\Omega$  no ponto "S"



referida equação. Para a primeira integral, quando o referido limite é considerado, torna-se evidente que [4.11]:

$$\lim_{\epsilon \rightarrow 0} \left[ \int_{\Gamma-\bar{\Gamma}} P_{ik}^*(S,Q) u_k(Q) d\Gamma(Q) \right] = \int_{\Gamma} P_{ik}^*(S,Q) u_k(Q) d\Gamma(Q)$$

notando-se, entretanto, que devido ao tipo de singularidade de  $P_{ik}^*$  essa integral necessariamente deve ser analisada no sentido de valor principal.

Já a segunda parcela precisa ser analisada com maior cuidado. Em primeiro lugar, a integração do tensor  $P_{ik}^*$  no contorno  $\Gamma_\epsilon$  depende do domínio  $\Omega$  considerado. Considerando o domínio infinito, os valores de  $P_{ik}^*$  são da ordem de  $1/\epsilon^2$ , enquanto que o resultado da integração sobre a superfície é da ordem de  $\epsilon^2$ . Conseqüentemente, a integral não tem valor definido quando  $\epsilon \rightarrow 0$ , mas produz um termo independente que pode ser calculado, pela substituição das equações 4.19 e a consideração de que o ponto "S" está numa superfície sem angulosidade. Assim sendo pode-se escrever [4.11] :

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \left[ \int_{\Gamma_\epsilon} P_{ik}^*(S,Q) u_k(Q) d\Gamma(Q) \right] &= \\ &= u_k(Q) \lim_{\epsilon \rightarrow 0} \left[ \int_{\Gamma-\bar{\Gamma}} P_{ik}^*(S,Q) d\Gamma(Q) \right] = \\ &= u_k(Q) \left[ -\frac{1}{2} \delta_{ik} \right] = -\frac{1}{2} \delta_{ik} u_k(Q) \end{aligned}$$

A terceira parcela, de modo semelhante à primeira, resulta :

$$\lim_{\varepsilon \rightarrow 0} \left[ \int_{\Gamma - \tilde{\Gamma}} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q) \right] = \int_{\Gamma} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q)$$

Com a quarta parcela, ainda considerando-se o domínio infinito, os valores de  $U_{ik}^*$  são da ordem de  $1/\varepsilon$ , enquanto a integração no contorno  $\tilde{\Gamma}_\varepsilon$  produz  $\varepsilon^2$ . Assim pode-se concluir que :

$$\lim_{\varepsilon \rightarrow 0} \left[ \int_{\tilde{\Gamma}_\varepsilon} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q) \right] = 0$$

Quanto às duas últimas parcelas, verifica-se facilmente que quando  $\varepsilon \rightarrow 0$  a integral em  $\Omega$  representa todo o conjunto pois a parcela em  $\tilde{\Omega}_\varepsilon$  tende a zero.

Desse modo, a equação 4.26 pode ser generalizada para diversos domínios e pontos colocados no interior ou contorno da seguinte forma :

$$\begin{aligned} c_{ik}(S) u_i(S) = & - \int_{\Gamma} P_{ik}^*(S, Q) u_k(Q) d\Gamma(Q) + \\ & + \int_{\Gamma} U_{ik}^*(S, Q) p_k(Q) d\Gamma(Q) + \\ & + \int_{\Omega} U_{ik}^*(S, q) b_k(q) d\Omega(q) \quad \dots(4.27) \end{aligned}$$

onde  $c_{ik}(S)$ , para os casos que interessam mais especificamente a este trabalho, pode ser definido como [4.12] :

Solução de Kelvin ou Mindlin

$$\begin{aligned} S \in \Omega & \Rightarrow c_{ik}(S) = \delta_{ik} \\ S \in \Gamma & \Rightarrow c_{ik}(S) = 1/2 \delta_{ik} \quad \dots(4.28) \end{aligned}$$

Solução de Boussinesq-Cerruti

$$S \in \Omega \Rightarrow c_{ik}(S) = \delta_{ik}$$

$$S \in \Gamma \Rightarrow c_{ik}(S) = \delta_{ik}$$

...(4.29)

ressaltando-se que nos pontos do contorno deve haver necessariamente um único plano tangente.

Obviamente, a resolução analítica da expressão 4.27 é muito trabalhosa e não representa um caminho interessante para a solução de um problema elástico. Muito mais interessante é a subdivisão do contorno de um determinado domínio em um número finito de regiões, definindo-se uma função aproximadora, de modo que essas integrações possam ser realizadas com facilidade. Então, a aplicação da equação integral a pontos dessa região produzirá um sistema de equações lineares. Pela imposição das condições de contorno em cada ponto, ou os deslocamentos ou as forças de superfície são conhecidos, o sistema obtido terá o mesmo número de incógnitas e equações. Assim sendo, sua resolução permitirá o conhecimento de todos os parâmetros de interesse para os problemas elásticos.

Quando procede-se da maneira assim descrita, está-se tratando da solução numérica da equação 4.27. Este é o conceito básico do Método dos Elementos de Contorno, que se desenvolverá para o domínio semi-infinito nos próximos itens.

### 4.3. A FORMULAÇÃO DO ELEMENTO SAPATA RÍGIDA

#### 4.3.1. Introdução

Para a definição do elemento sapata, duas condições foram de fundamental importância : o tempo de processamento e a facilidade na entrada dos dados. Isso porque o objetivo principal deste elemento, realizar a interação solo-estrutura de edifícios sobre fundação direta, tem um enfoque bastante prático. Este enfoque estaria comprometido se a entrada dos dados requeresse muito trabalho e o tempo computacional de execução do programa crescesse demais. Deve-se lembrar que exatamente essas condições são que inviabilizam a utilização do elemento sólido tipo 6 na consideração da interação desejada. Assim sendo, procurou-se com a definição deste elemento, através do Método dos Elementos de Contorno, uma forma simples, e sobretudo rápida, de considerar-se a mencionada interação.

A característica fundamental do elemento mencionado diz respeito à sua adaptação a um programa de Elementos Finitos, sem que haja um comprometimento do desempenho desse programa para todas as suas outras aplicações. Desse modo, optou-se por um processo pouco ortodoxo, mas eficiente, de se utilizar o Método dos Elementos de Contorno para a montagem da matriz de rigidez associada ao ponto onde se liga a sapata. Mais explicitamente, os passos principais desse processo podem ser resumidos nas seguintes etapas :

a) Inicialmente define-se o elemento como composto de uma ou mais sapatas, cada uma ligada a um ponto nodal da superestrutura. Caso o elemento tenha mais de uma sapata, automaticamente será considerada a interação entre elas. Caso apenas uma sapata tenha sido definida para o elemento,

seu comportamento será considerado independente das demais.

b) O programa automaticamente realiza a discretização dessas sapatas, calculando todos os parâmetros necessários para a montagem de um sistema de equações através do Método dos Elementos de Contorno. O solo será considerado como um domínio semi-infinito, isotrópico, homogêneo e perfeitamente elástico e a sapata como sendo perfeitamente rígida.

c) É calculada, pela definição, a matriz de rigidez correspondente a cada centróide de sapata. Isso se faz pela aplicação de deslocamentos e rotações unitários em cada um desses centróides, mas sempre em relação ao sistema global de referência. Esses deslocamentos e rotações são então transformados em deslocamentos para os nós que discretizam as sapatas, encontrando-se como resposta as forças de superfície nesses mesmos nós. A integração dessas forças de superfície produz como resultado uma matriz de coeficientes elásticos para cada um dos centróides.

d) As matrizes de coeficientes encontradas no item anterior são então transportadas para os pontos nodais da superestrutura ligados a cada sapata. Depois as matrizes são simetrizadas e montadas nas devidas posições do sistema de equações globais da superestrutura, como é feito com todos os outros tipos de elementos. Então, a estrutura é calculada de forma convencional, apenas ressaltando-se que nos resultados obtidos se encontra automaticamente considerada a interação com o solo, da maneira mencionada.

Exatamente essas etapas é que estarão descritas, com detalhes, nos próximos itens deste capítulo.

#### 4.3.2. Forma e Função Aproximadora

Pode-se considerar que duas especificações são fundamentais para a definição deste elemento. A primeira é sua forma e a segunda a função aproximadora a ser utilizada.

Quanto à primeira, considerando-se que a entrada de dados de maneira simples e objetiva é uma característica fundamental e que a grande maioria das sapatas usualmente utilizadas são retangulares, foi essa exatamente a forma adotada. Assim, para uma definição completa da geometria do elemento são necessários os seguintes parâmetros :

- $dx_1, dx_2, dx_3$  : distâncias entre o nó da estrutura e o centróide da sapata, segundo os eixos globais  $X_1, X_2$  e  $X_3$ .
- $\beta$  : ângulo entre o eixo global  $X_1$  e o local  $x_1$ , positivo no sentido horário.
- $C_1, C_2$  : dimensões da sapata segundo os eixos locais  $x_1$  e  $x_2$ .

Todos esses parâmetros são apresentados na figura 4.10. Como pode-se observar na referida figura, os eixos locais são definidos com origem no centróide da sapata e orientados segundo direções paralelas aos seus lados. Ressalta-se ainda que, pelos parâmetros mencionados, a sapata é considerada contida num plano paralelo ao plano formado pelos eixos globais  $X_1$  e  $X_2$ , podendo apenas apresentar-se rotacionada em relação a esses eixos.

Menciona-se ainda que, como sistema de referência homogêneo auxiliar, foram adotados os eixos "r" e "s", coincidentes com o sistema local de referência  $x_1$  e  $x_2$  e apresentados na figura 4.11.

Já quanto à função aproximadora, adotou-se que tanto a geometria como os deslocamentos "u" e forças de superfície

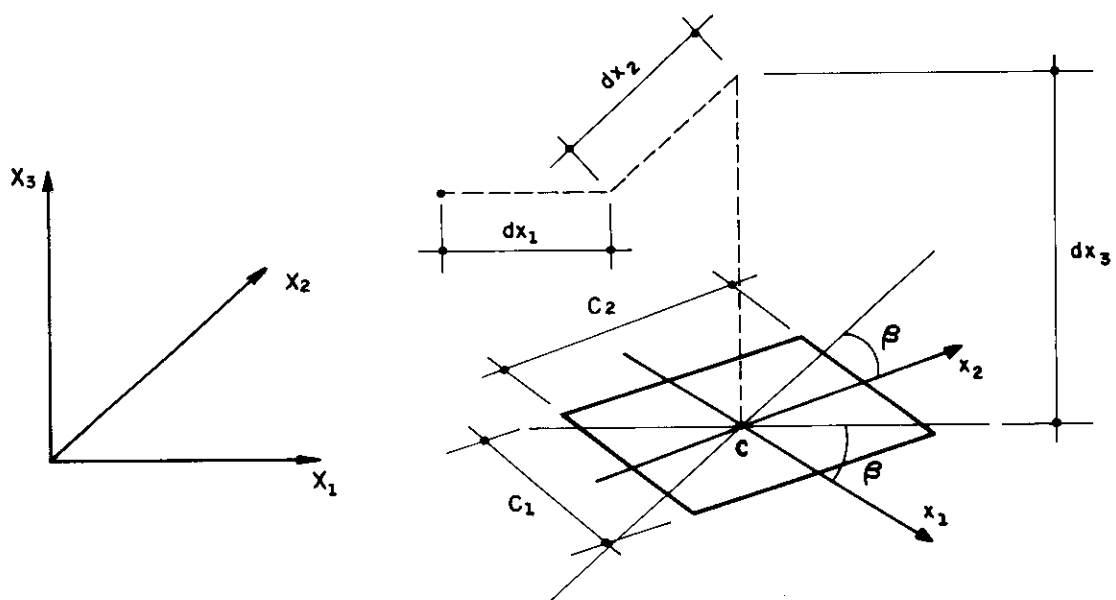


fig. 4.10 A geometria da sapata

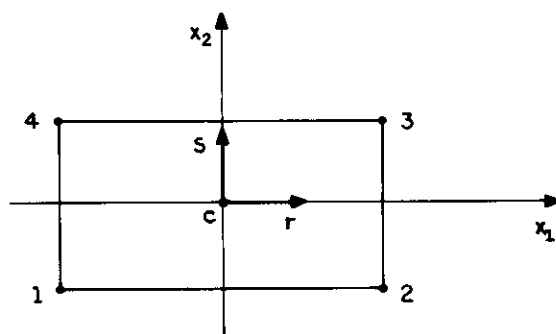


fig. 4.11 Numeração dos nós e sistema homogêneo

"p" são aproximados pela mesma função. Além disso, a sapata é discretizada por apenas quatro nós, colocados em seus cantos e numerados de acordo com o esquema apresentado na já referida figura 4.11.

Qualquer parâmetro de um ponto interno pode ser calculado em função dos valores desse parâmetro nos nós da sapata. Para isso, basta que se defina funções apropriadas a estas transformações. Essas funções são ponderadores cujo valor resulta 1 para o nó base da função e zero para todos os outros.

Existe toda uma família de funções para os diversos tipos de elementos retangulares de diferentes números de nós. Essa família é conhecida como família Serendípita [4.13]. Como foi adotado que o elemento aqui desenvolvido é discretizado por apenas quatro nós, o conjunto das quatro funções mencionadas pode ser escrito, em relação ao sistema homogêneo, da seguinte maneira :

$$\begin{aligned}
 h^1 &= \frac{1}{4} (1-r) (1-s) \\
 h^2 &= \frac{1}{4} (1+r) (1-s) \\
 & \dots (4.30) \\
 h^3 &= \frac{1}{4} (1+r) (1+s) \\
 h^4 &= \frac{1}{4} (1-r) (1+s)
 \end{aligned}$$

Graficamente, as funções 4.30 podem ser representadas conforme mostra-se na figura 4.12.

#### 4.3.3. Parâmetros Internos em Função de Valores Nodais

Em primeiro lugar, é interessante ressaltar que a notação matricial, apesar de produzir expressões mais extensas, tem a vantagem de explicitar bem as operações



desenvolvidas. Assim sendo, a partir deste ponto, volta-se a utilizá-la em paralelo à notação indicial, sempre que se considerar necessário imprimir maior clareza às deduções efetuadas.

As coordenadas de um ponto interno ao elemento podem ser escritas em função das coordenadas dos pontos nodais através da expressão :

$$X_i = h^j X_i^j \quad \dots(4.31)$$

onde,

$X_i$  : coordenada de um ponto da sapata segundo o eixo "i".

$h^j$  : funções interpoladoras definidas nas expressões 4.30.

$X_i^j$  : é a coordenada, segundo o eixo "i", do ponto nodal "j".

Matricialmente, pode-se escrever :

$$X = H X_n \quad \dots(4.32)$$

onde  $X$  é o vetor de coordenadas do ponto, ou seja :

$$X = \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \end{Bmatrix}$$

$H$  é a matriz de transformação, montada com as funções de interpolação da maneira que se segue :

$$H = \begin{bmatrix} h^1 & h^2 & h^3 & h^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h^1 & h^2 & h^3 & h^4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h^1 & h^2 & h^3 & h^4 \end{bmatrix}$$

$X_n$  é um vetor que agrupa as coordenadas nodais do elemento

da seguinte maneira :

$$\tilde{x}_n = \begin{Bmatrix} x_1^1 \\ x_1^2 \\ x_1^3 \\ x_1^4 \\ \cdot \\ \cdot \\ \cdot \\ x_3^1 \\ x_3^2 \\ x_3^3 \\ x_3^4 \end{Bmatrix}$$

onde  $x_1^j$  é a coordenada do nó "j" segundo o eixo "i".

Para os deslocamentos e forças de superfície, aproximações semelhantes podem ser utilizadas. Nesse caso as expressões resultariam :

$$U_i = h^j U_i^j \quad \dots(4.33)$$

$$P_i = h^j P_i^j \quad \dots(4.34)$$

onde,

$U_i$  e  $P_i$  : componentes dos deslocamentos e forças de superfície para um ponto interno.

$U_i^j$  e  $P_i^j$  : valores nodais dos deslocamentos e forças de superfície.

De forma semelhante ao que se mostrou para as coordenadas, matricialmente as expressões resultam :

$$\vec{U} = H \vec{U}_n \quad \dots(4.35)$$

$$\vec{P} = H \vec{P}_n \quad \dots(4.36)$$

onde,

$\vec{U}$  e  $\vec{P}$  : vetores que contém, respectivamente, os deslocamentos e forças de superfície, segundo os três eixos coordenados, para um ponto interior.

$\vec{U}_n$  e  $\vec{P}_n$  : vetores que agrupam, respectivamente, os deslocamentos e forças de superfície nodais de forma semelhante ao vetor  $X_n$  da expressão 4.32.

Com as aproximações definidas, pode-se passar à discretização da equação integral 4.27. Essa discretização é que resultará no sistema de equações que resolvido permite o conhecimento dos parâmetros envolvidos.

#### 4.3.4. Discretização da Equação Integral

No final do item 4.2.3, chegou-se à uma equação integral que relaciona deslocamentos e forças de superfície para pontos do contorno de um dado domínio. Essa equação, pode ser simplificada pela consideração de duas circunstâncias particulares.

Em primeiro lugar, pode ser desprezada a parcela referente às forças volumétricas, sem importância para as aplicações a serem aqui desenvolvidas.

Depois, deve-se considerar que a parcela que depende de  $P^*$  será sempre nula para os casos aqui tratados, devido às considerações que se faz a seguir.

Quando a sapata a ser considerada apoiar-se sobre o plano livre de tensões, solução de Boussinesq-Cerruti, o valor das componentes do tensor  $P^*$  será zero. Este

inclusive é o caso mais comumente encontrado na prática, no qual a sapata apóia-se na cota do terreno escavado, desprezando-se o efeito de algum possível reaterro por considerar-se que essa providência não reconstitui o terreno original.

Caso a cota de apoio da sapata seja considerada realmente dentro do domínio semi-infinito, solução de Mindlin, deve-se lembrar que na verdade a definição do contorno do elemento envolve a definição de seis planos, de  $\Gamma_1$  a  $\Gamma_6$ , conforme apresenta-se na figura 4.13. Se for adotada a espessura da sapata igual a zero, isto é  $\Gamma_1$  e  $\Gamma_2$  ocupando a mesma posição, as duas integrais restantes da equação 4.27 anulam-se para os contornos  $\Gamma_3$  a  $\Gamma_6$ . Entretanto, para os contornos  $\Gamma_1$  e  $\Gamma_2$  anulam-se apenas as parcelas relativas a  $P^*$ , pois trata-se da integração de funções de mesmo valor e sinais contrários na mesma área.

Quanto à matriz  $c_{ik}(S)$ , a soma de  $1/2 \delta_{ik}$  de uma superfície com  $1/2 \delta_{ik}$  da outra produzirá o próprio  $\delta_{ik}$ , ou seja a matriz identidade. Portanto, seja usada a solução de Boussinesq-Cerruti ou Mindlin, a equação integral a ser discretizada resume-se ao seguinte :

$$u_i(S) = \int_{\Gamma} U_{ik}^*(S,Q) p_k(Q) d\Gamma(Q) \quad \dots(4.37)$$

Com a utilização da expressão 4.33 e considerando-se que se tenha "L" sapatas para um determinado elemento obtém-se :

$$U_i^j = \sum_{l=1}^L \left[ \int_{\Gamma_1} U_{ik}^* h^j p_k^j d\Gamma_1 \right] \quad \dots(4.38)$$

onde,

$U_i^j$  : é o deslocamento do ponto nodal "j" na direção "i".

Retirando-se da integral os valores nodais, que não

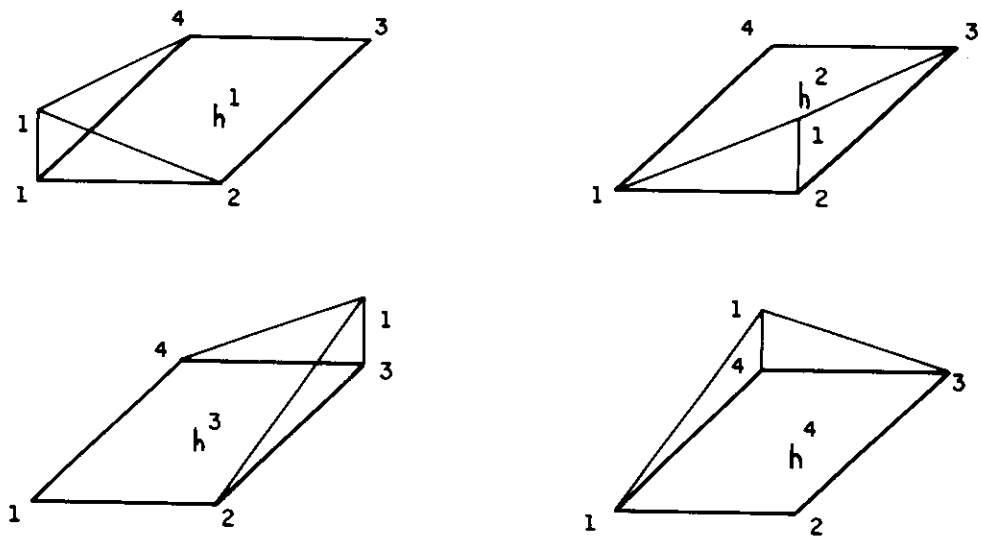


fig. 4.12 Funções interpoladoras

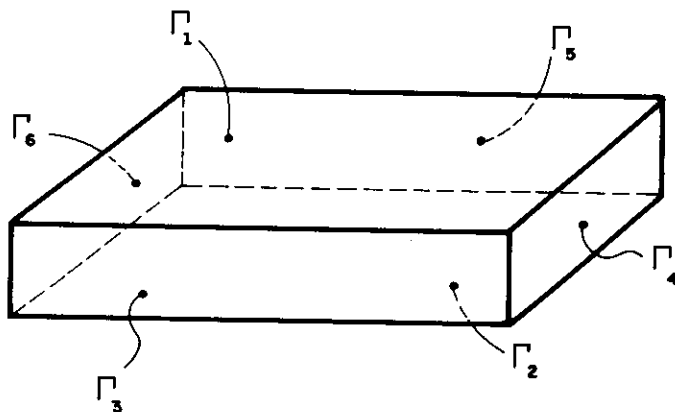


fig. 4.13 Contornos para discretização da sapata

variam em relação à posição no elemento, obtém-se :

$$U_i^j = \sum_{l=1}^L \left[ \int_{\Gamma_1} U_{ik}^* h^j d\Gamma_1 \right] P_k^j \quad \dots(4.39)$$

A integral no contorno  $\Gamma_1$  pode ser calculada numericamente por Gauss [4.14], através da consideração de "M" pontos. Desse modo, pode-se escrever :

$$U_i^j = \sum_{l=1}^L \left[ |J| \sum_{m=1}^M \left[ \omega_m \left( U_{ik}^* h^j \right) \right] \right] P_k^j \quad \dots(4.40)$$

onde,

$|J|$  : é o jacobiano da transformação de coordenadas de  $d\Gamma$  para  $drds$  [4.15]. Esse valor, no caso da sapata retangular, é igual a  $C_1 C_2 / 4$

$\omega_m$  : é o fator de peso fornecido pelo processo de integração de Gauss. No caso, o ponto "m" tem sua posição definida de acordo com o número de pontos a serem considerados para a integração.

Aplicando-se a equação 4.40 a todos os pontos nodais de um elemento, obtém-se um sistema de equações lineares que resolvido fornece os valores das forças de superfície para os pontos onde os deslocamentos são conhecidos. Para maior clareza, pode-se expressar esse sistema em termos matriciais da maneira que se segue :

$$\begin{Bmatrix} \sim U^1 \\ \sim U^2 \\ \sim U^3 \\ \vdots \\ \sim U^n \end{Bmatrix} = \begin{bmatrix} G_{\sim 11} & G_{\sim 12} & G_{\sim 13} & \cdot & \cdot & \cdot & G_{\sim 1n} \\ G_{\sim 21} & G_{\sim 22} & G_{\sim 23} & \cdot & \cdot & \cdot & G_{\sim 2n} \\ G_{\sim 31} & G_{\sim 32} & G_{\sim 33} & \cdot & \cdot & \cdot & G_{\sim 3n} \\ \vdots & \vdots & \vdots & & & & \vdots \\ G_{\sim n1} & G_{\sim n2} & G_{\sim n3} & \cdot & \cdot & \cdot & G_{\sim nn} \end{bmatrix} \begin{Bmatrix} P^1 \\ \sim P^2 \\ \sim P^3 \\ \vdots \\ \sim P^n \end{Bmatrix} \quad \dots(4.41)$$

Os vetores  $U^i$  contém os deslocamentos do nó "i" segundo as três direções coordenadas, ou seja :

$$\tilde{U}^i = \begin{pmatrix} U_1^i \\ U_2^i \\ U_3^i \end{pmatrix} \quad \dots(4.42)$$

Os vetores  $P^i$  são semelhantes a  $U^i$ , só que contendo as forças de superfície do nó "i". Já  $G_{ij}$  é uma matriz de ordem três que pode ser escrita como :

$$G_{ij} = |J| \sum_{m=1}^M \left[ \omega_m \left( \tilde{U}^* h^j \right)_{im} \right] \quad \dots(4.43)$$

onde,

$\tilde{U}^*$  : é a matriz de ordem três que contém a solução fundamental calculada do nó "i" ao ponto de Gauss "m".

Após as deduções aqui colocadas deve estar perfeitamente claro que se pode, através da imposição de deslocamentos para os pontos nodais, calcular as forças de superfície para esses mesmos pontos. Esse procedimento permitirá o cálculo da rigidez das sapatas pela definição, o que se encontra explicitado no próximo item.

#### 4.3.5. Cálculo da Rigidez das Sapatas

Após a montagem do sistema de equações 4.41, o objetivo deste trabalho é o de calcular coeficientes que simulem a presença do solo e possam ser montados na matriz de rigidez global da superestrutura. Mais especificamente, para o sistema computacional aqui desenvolvido, trata-se de

determinar os referidos parâmetros de rigidez associados aos seis graus de liberdade de cada nó com sapata associada. Isso produz uma matriz de ordem  $6N$ , sendo "N" o número de nós da superestrutura, ou de sapatas, presentes no elemento.

Inicialmente a rigidez será determinada sempre em relação ao centróide da sapata, mas de acordo com o sistema global de referência. Portanto, a transferência dessas rigidezes para o nó da superestrutura envolverá apenas as transformações relativas às translações entre o centróide e o nó. Essas transformações serão realizadas em estágio posterior e encontram-se detalhadas no próximo item.

Como foi mencionado, existem seis graus de liberdade a serem considerados para cada sapata : três translações, segundo  $X_1, X_2$  e  $X_3$ , e três rotações, em torno de  $X_1, X_2$  e  $X_3$ . Para o cálculo dos coeficientes de rigidez, deve-se impor a cada um desses graus de liberdade, de forma independente, um movimento unitário, conforme mostra-se na figura 4.14,

Cada movimento será então transformado em três deslocamentos para cada um dos quatro nós que definem a sapata. Os vetores para os seis carregamentos mencionados podem ser expressos, em sua forma transposta, da maneira que se segue :

$$U_{\sim 1}^T = \left\{ \begin{array}{cccccccccccc} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{array} \right\}$$

$$U_{\sim 2}^T = \left\{ \begin{array}{cccccccccccc} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{array} \right\}$$

$$U_{\sim 3}^T = \left\{ \begin{array}{cccccccccccc} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{array} \right\}$$

$$U_{\sim 4}^T = \left\{ \begin{array}{cccccccccccc} 0 & 0 & A_1 & 0 & 0 & A_2 & 0 & 0 & A_3 & 0 & 0 & A_4 \end{array} \right\}$$



$$\tilde{U}_5^T = \left\{ \emptyset \quad \emptyset \quad B_1 \quad \emptyset \quad \emptyset \quad B_2 \quad \emptyset \quad \emptyset \quad B_3 \quad \emptyset \quad \emptyset \quad B_4 \right\}$$

$$\tilde{U}_6^T = \left\{ C_1 \quad D_1 \quad \emptyset \quad C_2 \quad D_2 \quad \emptyset \quad C_3 \quad D_3 \quad \emptyset \quad C_3 \quad D_3 \quad \emptyset \right\}$$

...(4.44)

onde

$\tilde{U}_i$  : é o vetor correspondente ao  $i$ -ésimo carregamento de uma determinada sapata.

Já os parâmetros A, B, C e D são distâncias dadas pelas expressões :

$$A_i = X_2(\text{nó } i) - X_2(\text{centróide})$$

$$B_i = X_1(\text{centróide}) - X_1(\text{nó } i) \quad \dots(4.45)$$

$$C_i = X_2(\text{centróide}) - X_2(\text{nó } i)$$

$$D_i = X_1(\text{nó } i) - X_1(\text{centróide})$$

Após essa etapa, o conjunto formado pelos deslocamentos obtidos para os nós da sapata considerada e mais três valores nulos para cada nó de outras sapatas, constitui um vetor que deve ser levado ao sistema 4.41. Quando todos os movimentos unitários tiverem sido considerados, ter-se-á um total de  $6N$  vetores.

A solução do sistema é realizada pelo processo de triangularização de Gauss, com os  $6N$  vetores independentes considerados simultaneamente. Assim, pode-se conhecer os valores das forças de superfície para cada nó de todas as sapatas, valores esses organizados em um vetor para cada movimento unitário. A integração numérica desses valores, sempre em relação ao centróide e tendo como referência o sistema global, produz a matriz de rigidez do elemento.

Após o procedimento aqui descrito, resta apenas operar

as últimas transformações na referida matriz e realizar sua montagem na matriz de rigidez global da superestrutura.

#### 4.3.6. Translação da Matriz de Rigidez

Uma rigidez calculada para um determinado ponto pode ser transferida para outro através de transformações convenientes [4.16]. Supondo os pontos  $O$  e  $\bar{O}$  separados por distâncias  $dx_1$ ,  $dx_2$  e  $dx_3$  e referenciados respectivamente pelos sistemas  $X_1 X_2 X_3$  e  $\bar{X}_1 \bar{X}_2 \bar{X}_3$ , conforme mostra-se na figura 4.15, pode-se escrever para a transferência de forças e momentos atuantes em  $\bar{O}$  para  $O$  :

$$\begin{Bmatrix} P_1 \\ P_2 \\ P_3 \\ M_1 \\ M_2 \\ M_3 \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & -dx_3 & dx_2 & 1 & 0 & 0 \\ dx_3 & 0 & -dx_1 & 0 & 1 & 0 \\ -dx_2 & dx_1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \bar{P}_1 \\ \bar{P}_2 \\ \bar{P}_3 \\ \bar{M}_1 \\ \bar{M}_2 \\ \bar{M}_3 \end{Bmatrix} \quad \dots (4.46)$$

Resumidamente, tem-se :

$$\underset{\sim}{P} = \underset{\sim}{T} \underset{\sim}{\bar{P}} \quad \dots (4.47)$$

onde,

$\underset{\sim}{\bar{P}}$  : são as forças e momentos atuantes em  $\bar{O}$ .

$\underset{\sim}{P}$  : são as forças e momentos atuantes em  $O$ .

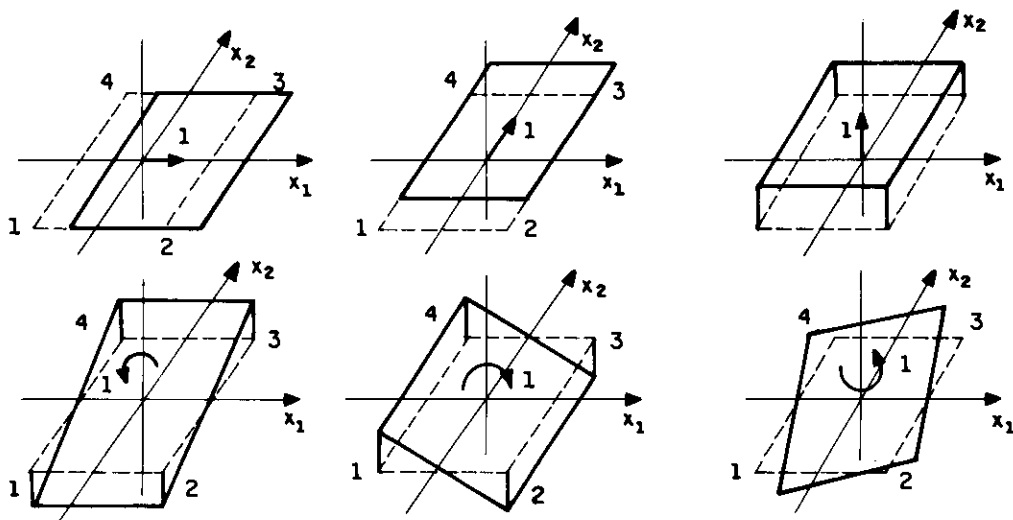


fig. 4.14 Movimentos unitarios da sapata

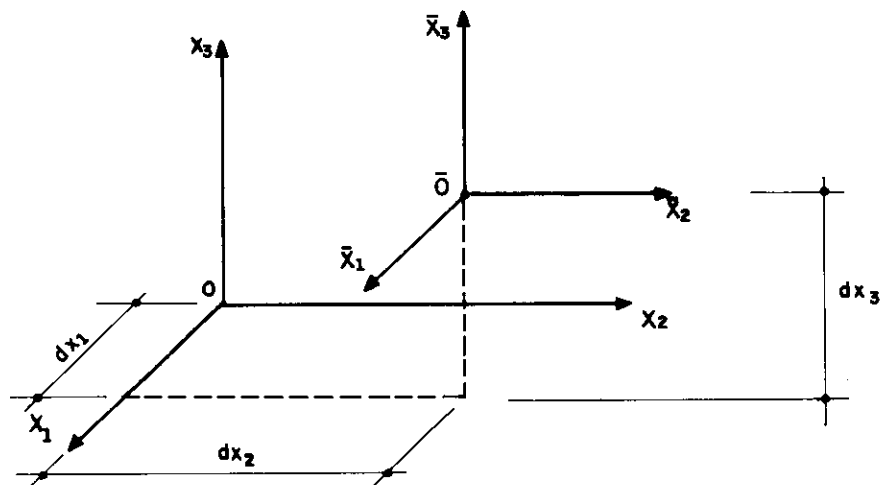


fig. 4.15 Transferência de rigidez

$\tilde{T}$  : é a matriz que realiza a transformação mencionada

A matriz T pode ser subdividida em quatro submatrizes, de acordo com o que se segue :

$$\tilde{T} = \begin{bmatrix} I & 0 \\ \sim & \sim \\ W & I \\ \sim & \sim \end{bmatrix} \quad \dots(4.48)$$

Já a sua inversa pode ser escrita simplesmente como :

$$\tilde{T}^{-1} = \begin{bmatrix} I & 0 \\ \sim & \sim \\ W^T & I \\ \sim & \sim \end{bmatrix} \quad \dots(4.49)$$

Portanto, a transformação inversa escreve-se :

$$\bar{P} = \tilde{T}^{-1} P \quad \dots(4.50)$$

Considerando-se agora a transferência dos deslocamentos e rotações que ocorrem em O para o ponto  $\bar{O}$  tem-se :

$$\bar{U} = \tilde{T}^T U \quad \dots(4.51)$$

onde,

$\bar{U}$  e U : são os deslocamentos e rotações em  $\bar{O}$  e O, respectivamente.

$\tilde{T}^T$  : é a transposta da matriz T apresentada em 4.46.

Para o ponto  $\bar{O}$ , considerando-se  $\bar{S}$  como a matriz de rigidez que relaciona as forças e deslocamentos, pode-se escrever :

$$\begin{matrix} \bar{S} & \bar{U} & = & \bar{P} \\ \sim & \sim & & \sim \end{matrix} \quad \dots(4.52)$$

Através da utilização das equações 4.50 e 4.51, obtém-se :

$$\begin{matrix} \bar{S} & T^T & U & = & T^{-1} & P \\ \sim & \sim & \sim & & \sim & \sim \end{matrix} \quad \dots(4.53)$$

ou ainda,

$$\begin{matrix} T & \bar{S} & T^T & U & = & P \\ \sim & \sim & \sim & \sim & & \sim \end{matrix} \quad \dots(4.54)$$

Chamando-se de matriz  $S$  aquela que relaciona as forças e deslocamentos para o ponto  $O$ , tem-se :

$$\begin{matrix} S & = & T & \bar{S} & T^T \\ \sim & & \sim & \sim & \sim \end{matrix} \quad \dots(4.55)$$

A expressão 4.54 mostra como transladar uma matriz de rigidez, calculada inicialmente num ponto  $\bar{O}$ , para um ponto  $O$ . Aqui, a expressão será utilizada para transladar a matriz de rigidez do elemento, montada em relação ao centróide de cada sapata, para os nós da superestrutura. Somente após essa translação é que ela será adicionada à matriz de rigidez global.

Para se realizar essa translação, deve-se lembrar que a matriz total é composta de  $N^2$  submatrizes, sendo  $N$  o número de sapatas ou de nós da superestrutura, conforme mostra-se no esquema seguinte :

$$\bar{S} = \begin{bmatrix} \bar{S}_{\sim 11} & \bar{S}_{\sim 12} & \dots & \bar{S}_{\sim 1N} \\ \bar{S}_{\sim 21} & \bar{S}_{\sim 22} & \dots & \bar{S}_{\sim 2N} \\ \vdots & \vdots & & \vdots \\ \bar{S}_{\sim N1} & \bar{S}_{\sim N2} & \dots & \bar{S}_{\sim NN} \end{bmatrix}$$

onde,

$\bar{S}_{\sim ij}$  : é a submatriz de ordem seis que relaciona deslocamentos e forças nos centróides das sapatas "i" e "j".

Nesse caso, a transformação da rigidez para os nós da superestrutura pode ser realizada através da expressão 4.54, com a matriz de transformação  $T$  arranjada na forma :

$$T = \begin{bmatrix} T_{\sim 1} & 0 & \dots & 0 \\ 0 & T_{\sim 2} & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & T_{\sim N} \end{bmatrix}$$

onde,

$T_{\sim i}$  : é a matriz da expressão 4.46, montada para a translação da rigidez entre o centróide da sapata "i" e o ponto nodal da superestrutura que lhe corresponde.

Finalmente, resta ressaltar que a transferência da rigidez do elemento resume-se ao seguinte :

$$\bar{S}_{\sim ij} = T_{\sim i} \bar{S}_{\sim ij} T_{\sim j}^T \quad \dots(4.56)$$

onde,

$\tilde{S}_{ij}$  : são as submatrizes já transladadas para os nós da superestrutura.

$\bar{S}_{ij}$  : são as submatrizes ainda em relação aos centróides das sapatas.

$T_j$  : são as matrizes de transformação que relacionam as rigidezes no centróide e no nó da superestrutura para a sapata "j".

#### 4.3.7. Simetrização da Matriz de Rigidez

A matriz obtida para o elemento, através do Método dos Elementos de Contorno (M.E.C.), tem a característica de não apresentar simetria em relação à diagonal principal. Já a matriz da superestrutura, obtida através do Método dos Elementos Finitos (M.E.F.), tem como uma das suas principais características exatamente essa simetria. O problema de como realizar a ligação entre esses dois métodos de análise tem preocupado com frequência cada vez maior os pesquisadores da área de métodos numéricos. Existem algumas maneiras de resolvê-lo, dentre as quais vai-se destacar as seguintes :

a) Desenvolver a formulação do Método dos Elementos de Contorno de modo a obter-se matrizes simétricas, a exemplo do que acontece com o Método dos Elementos Finitos [4.17], [4.18], [4.19].

b) Fazer com que as equações correspondentes à região discretizada através do Método dos Elementos Finitos sejam condensadas nos pontos nodais de interface, adicionando-se

essas contribuições ao sistema não simétrico montado para a região discretizada através do Método dos Elementos de Contorno. A solução desse sistema não simétrico fornece resultados para toda a região de M.E.C., interface inclusive, e os deslocamentos da região de M.E.F. podem ser encontrados por retrosubstituição [4.20].

c) Operar, de forma independente, nos sistemas de equações que dizem respeito ao Método dos Elementos Finitos e Método dos Elementos de Contorno, montando um terceiro conjunto de equações para os pontos nodais da interface. Resolvido esse terceiro sistema, voltar aos dois sistemas originais e completar a solução das equações.

d) Simetrizar a matriz obtida através do Método dos Elementos de Contorno posteriormente, com a utilização de um processo qualquer definido para esse fim. Após esse procedimento, realizar normalmente a adição à matriz simétrica da superestrutura [4.21].

Evidentemente, todos esses processos têm suas vantagens e desvantagens. Para o primeiro, que a princípio admite-se vantajoso para estruturas com grandes regiões de M.E.F., as principais restrições dizem respeito à grande complexidade das teorias e algoritmos até agora desenvolvidos para obter-se a simetrização das matrizes montadas através do M.E.C.. Assim, corre-se o risco de se gastar muito esforço computacional para a obtenção de resultados de eficiência duvidosa.

Quanto às alternativas b) e c), admite-se que possam ser vantajosas para estruturas com grandes regiões de M.E.C., ou pelo menos com regiões de M.E.C. e M.E.F. equivalentes, o que não se adapta com perfeição ao caso específico aqui estudado. Para um programa geral, onde espera-se que a utilização mais frequente seja a de elementos baseados no M.E.F., isso pode tornar-se uma



circunstância desvantajosa. Entretanto, entre os futuros desenvolvimentos previstos para o programa aqui apresentado, pretende-se incluir uma avaliação mais cuidadosa desse tópicó.

O quarto processo apresentado é evidentemente o mais simples e provavelmente o menos preciso. Simetrizar a matriz após a sua montagem pode levar, para os elementos fora da diagonal, à resultados bastantes discrepantes. Mesmo tomando-se como base o procedimento de procurar o erro médio, ou seja utilizar a média dos valores para coeficientes opostos em relação à diagonal principal, em certos casos isso pode não ser recomendável. Entretanto, para o caso aqui analisado, onde os coeficientes da diagonal são bastante preponderantes em relação aos que nela não se posicionam, o procedimento do erro médio parece ser defensável. Levando-se em conta sua grande simplicidade e o fato de não se necessitar de alterações no algoritmo de solução do sistema de equações, foi essa a solução adotada. Portanto, pode-se escrever simplesmente :

$$S_{ij} = \frac{1}{2} ( S_{ij} + S_{ji} ) \quad \dots(4.57)$$

onde,

$S_{ij}$  : são os elementos da matriz de rigidez obtida, para a linha "i" e coluna "j".

Após este procedimento de simetrização resta a montagem da matriz de rigidez obtida na matriz de rigidez global da superestrutura. Essa montagem será efetuada pela rotina montadora geral, já descrita em capítulo anterior, através de um vetor de contribuição. Portanto, um procedimento perfeitamente análogo ao realizado para os outros sete tipos de elementos existentes.

## 4.4. ANÁLISE DE RESULTADOS

### 4.4.1. Introdução<sup>N</sup>

O objetivo deste item é discutir uma série de resultados obtidos com o elemento desenvolvido para a modelagem de sapatas rígidas. Desse modo, espera-se um diagnóstico relativamente claro sobre as principais qualidades e deficiências apresentadas pela formulação proposta. Além disso, algumas informações interessantes a respeito de itens específicos, como por exemplo a importância da consideração da interação solo-estrutura ou o nível de interação entre as sapatas de uma fundação, devem ser obtidas.

Para tanto, optou-se por um estudo por etapas, de modo a cobrir o maior número possível desses pontos de forma conclusiva. O primeiro ponto a ser analisado é a acuidade numérica conseguida e os tempos de processamentos verificados. A importância deste tópico explica-se pelo fato da integração do contorno ser feita numericamente por Gauss. Assim sendo, de acordo com o número de pontos utilizados, pode-se obter maior ou menor precisão nos resultados e evidentemente maior ou menor tempo de processamento.

Logo após essas análises, passa-se a discutir uma série de matrizes de rigidez obtidas para sapatas isoladas. Dessa forma, pretende-se ter uma boa avaliação a respeito de alguns detalhes fundamentais do elemento. Pode-se citar, por exemplo, efeitos devidos à rotação da sapata em relação aos eixos globais, efeitos devidos à translação, imprecisões devidas à simetrização da matriz, influência da consideração da sapata como imersa no terreno, etc.

Após esses resultados para sapatas isoladas, discute-se os resultados obtidos para a matriz de rigidez

de grupos de sapatas. Nesse caso, o ponto de fundamental importância é realmente o afastamento com o qual essas interações passam a ser significativas. Além disso, alguns dos itens mencionados para sapatas isoladas também serão analisados.

Finalmente, como ponto de grande importância para o trabalho, tem-se o estudo da variabilidade dos esforços na superestrutura pela consideração da interação com a fundação. Nesse caso, vai-se procurar mostrar qual é o grau dessa variabilidade pelo cálculo sucessivo da mesma estrutura considerando-se a fundação totalmente rígida ou flexível, com pelo menos dois valores de módulo de elasticidade.

A respeito dos parâmetros que descrevem o comportamento elástico linear do solo, módulo de elasticidade e o coeficiente de Poisson, é fato notório que sua determinação é um assunto muito complexo. Entretanto, o enfoque principal desejado por esta pesquisa continua sendo a estrutura. Desse modo, apenas serão apresentados alguns valores médios para essas características, sem que este ponto tenha o tratamento cuidadoso que mereceria se o enfoque não fosse o mencionado. Assim, apresentam-se nas tabelas 4.1 e 4.2, respectivamente, os valores para  $\nu$  e  $E$  [4.22].

Coeficientes de Poisson Típicos	
Solo	$\nu$
Argila saturada	0.40 a 0.50
Argila parcialmente saturada	0.10 a 0.30
Argila arenosa	0.20 a 0.30
Silte	0.30 a 0.35
Areia compacta	0.20 a 0.40
Areia compacta grossa	0.15
Areia compacta fina	0.25

tabela 4.1

-----  
**Módulos de Elasticidade Típicos**  
 -----

Solo	E (KN/m <sup>2</sup> )
Argila muito mole	300 a 3 000
Argila mole	2 000 a 4 000
Argila média	4 500 a 9 000
Argila dura	7 000 a 20 000
Argila arenosa	30 000 a 42 500
Silte	2 000 a 20 000
Areia siltosa	5 000 a 20 000
Areia fofa	10 000 a 25 000
Areia compacta	50 000 a 100 000
Areia compacta e pedregulhos	80 000 a 200 000

-----  
 tabela 4.2

Convém ressaltar que para todos os exemplos onde analisa-se apenas a matriz de rigidez obtida, os valores adotados foram  $E = 4\ 000\ \text{KN/m}^2$  e  $\nu = 0.35$ . Já nos outros casos esses valores encontram-se explicitados.

#### 4.4.2. Integração Numérica e Tempo de Processamento

Em todo o processo descrito anteriormente para a obtenção da matriz de rigidez de uma sapata existem dois pontos onde são utilizadas integrações numéricas para a obtenção de resultados. O primeiro é na montagem da matriz de coeficientes que relaciona deslocamentos e forças de superfície nos pontos nodais utilizados na discretização, a chamada matriz G do item 4.3.4. O segundo é na integração dos valores das forças de superfície para o cálculo dos parâmetros de rigidez no centróide das sapatas.

A integração das forças de superfície não deve causar preocupação. Isso porque com apenas dois pontos de Gauss por direção, portanto quatro pontos no total, a integração numérica resulta exata, já que a função a ser integrada é de primeiro grau.

Entretanto, não é isso o que acontece com a primeira integração mencionada, a da matriz de coeficientes do sistema que relaciona deslocamentos e forças de superfície no contorno. Nesse caso, a integração de uma função que depende de  $1/r$ , a situação é muito diferente e torna-se necessário um acompanhamento do processo para que se possa chegar ao menor número de pontos que ofereça a precisão desejada. Para chegar-se a esse número conveniente, um exemplo foi resolvido com diferentes conjuntos de pontos de integração. Os valores encontrados estão organizados na tabela 4.3, onde NP é o número de pontos de Gauss para cada direção, TMMG é o tempo de processamento para montagem da matriz G, TT é o tempo total para solução do elemento e  $S_{33}$  é o valor do terceiro elemento da diagonal principal da matriz de rigidez obtida, ou seja, o valor da força necessária para se obter um deslocamento unitário segundo o eixo  $X_3$ .

NP	TMMG (seg)	TT (seg)	$S_{33}$
2	0.11	0.33	8 397
3	0.16	0.44	8 174
4	0.33	0.60	8 096
5	0.50	0.82	8 060
6	0.77	1.04	8 041
8	1.31	1.65	8 021
10	2.09	2.37	8 012

tabela 4.3

Pela observação dos resultados, percebe-se que o valor obtido se a integração fosse exata deveria ser da ordem de 8 000. Assim sendo, para os propósitos deste trabalho, admite-se que a integração numérica realizada com quatro pontos por direção oferece a precisão desejada, sem que o tempo de processamento verificado seja elevado. Portanto, será esse o número de pontos de integração a ser adotado na montagem da referida matriz G.

É importante observar-se que existem procedimentos numéricos que integram equações com singularidade fraca, do tipo  $1/r$ , com mais precisão que o procedimento de Gauss. Entretanto, de acordo com os resultados obtidos, pode-se concluir que a utilização desses outros processos não resultaria em ganho significativo.

Quanto ao tempo de processamento total para a solução do elemento, verifica-se que é excelente. Como forma de completar-se a análise aqui desenvolvida, apresenta-se a tabela 4.4, onde acham-se compilados valores de tempos parciais e totais para elementos compostos por diversos números de sapatas. Nesse caso, NS é o número de sapatas interdependentes presentes no elemento, TMMG é o tempo de montagem da matriz G, TSS é o tempo de solução do sistema de equações lineares, TMMS é o tempo de montagem da matriz de rigidez do elemento, para as NS sapatas, e finalmente TT é o tempo total de cada elemento.

NS	TMMG (seg)	TSS (seg)	TMMS (seg)	TT (seg)
1	0.33	0.05	0.06	0.60
2	1.32	0.66	0.22	2.58
3	2.91	2.09	0.38	6.53
4	5.22	4.88	0.66	12.64
5	8.13	9.55	1.05	21.59
6	11.65	16.31	1.54	33.51

tabela 4.4

Novamente deve-se destacar que os tempos obtidos são excelentes, apesar do já esperado crescimento exponencial em função do número de sapatas presentes no elemento. De qualquer maneira, um tempo de processamento de aproximadamente 1/2 minuto para a obtenção da rigidez de uma fundação onde interrelacionam-se seis sapatas, realmente parece bastante satisfatório.

#### 4.4.3. Sapatas Isoladas

##### 4.4.3.1. Sapata 1m x 1m, centrada, sem rotação

O exemplo aqui apresentado encontra-se na figura 4.16. A matriz de rigidez obtida para o centróide da sapata, sem simetrização, é a seguinte :

6729	0	0	0	-534	0
0	6729	0	534	0	0
0	0	8096	0	0	0
0	-46	0	2620	0	0
46	0	0	0	2620	0
0	0	0	0	0	4151

A primeira observação importante a respeito da matriz apresentada é quanto à predominância da diagonal principal. Essa preponderância indica que para uma sapata isolada, centrada, sem rotação em relação ao sistema global de referência, as repercussões realmente importantes devidas a uma determinada ação vão ser medidas segundo a sua própria coordenada de atuação. Esse dado é muito importante devido ao processo de simetrização da matriz, a ser efetuado posteriormente pela consideração do erro mínimo, ou

elemento médio. Assim sendo, pode-se concluir que, pelo menos no caso específico aqui estudado, essa simetrização não deve ter consequências muito danosas para os resultados a serem obtidos.

Para continuar a análise desenvolvida, apresenta-se a seguir a matriz de rigidez já simetrizada, de acordo com o procedimento mencionado :

6729	0	0	0	-244	0
0	6729	0	244	0	0
0	0	8096	0	0	0
0	244	0	2620	0	0
-244	0	0	0	2620	0
0	0	0	0	0	4151

Como pode-se observar na matriz apresentada, existem apenas quatro elementos não nulos fora da diagonal. O valor da primeira linha com a quinta coluna, indica que ao aplicar-se uma ação segundo o eixo  $X_1$  obtém-se uma rotação em torno de  $X_2$ . Para o elemento da segunda linha e quarta coluna a situação é semelhante, uma força segundo  $X_2$  produzirá uma rotação em torno de  $X_1$ . Já para a valor da quarta linha e segunda coluna conclui-se que um momento em torno de  $X_1$  produz uma translação segundo  $X_2$ , semelhante ao elemento da quinta linha e primeira coluna, que indica que um momento em torno de  $X_2$  tem como consequência também uma translação segundo  $X_1$ .

Para maior clareza a respeito desses interrelacionamentos, pode-se aplicar sobre o nó ao qual essas rigidezes estão associadas forças e momentos segundo os eixos de referência. Adotando-se, apenas por uma questão de melhor visualização dos valores resultantes, forças iguais a  $10^4$  KN e momentos iguais a  $10^4$  KN·m, os resultados obtidos são os apresentados na tabela 4.5. É importante ressaltar que CAR indica o número do carregamento, respectivamente forças segundo  $X_1$ ,  $X_2$  e  $X_3$  e



momentos em torno de  $X_1$ ,  $X_2$  e  $X_3$ . Os valores  $TX_1$ ,  $TX_2$  e  $TX_3$  são as translações segundo os eixos de referência, em metros, e  $RX_1$ ,  $RX_2$  e  $RX_3$ , são as rotações em torno dos eixos de referência, em radianos.

CAR	$TX_1$	$TX_2$	$TX_3$	$RX_1$	$RX_2$	$RX_3$
1	1.488	.0000	.0000	.0000	.1402	.0000
2	.0000	1.488	.0000	-.1402	.0000	.0000
3	.0000	.0000	1.234	.0000	.0000	.0000
4	.0000	-.1402	.0000	3.826	.0000	.0000
5	.1402	.0000	.0000	.0000	3.829	.0000
6	.0000	.0000	.0000	.0000	.0000	2.401

tabela 4.5

Inicialmente pode-se estranhar o fato de uma força horizontal segundo  $X_1$  ou  $X_2$  provocar também uma rotação em torno de  $X_2$  ou  $X_1$ . Entretanto, pode-se comprovar tal fato, pelos menos qualitativamente, através de um modelo em elementos finitos. Para fugir-se dos elementos tridimensionais, normalmente muito custosos em termos de tempo de processamento, adota-se um esquema plano, conforme o que se apresenta na figura 4.17. A rede, composta de 200 elementos membrana ( tipo 4 ), tem ainda dois elementos barra ligando os nós 220 a 221 e 221 a 222. Esses elementos apresentam uma rigidez relativamente alta em relação à dos elementos membrana.

Nesse modelo, aplicando-se uma força horizontal no nó 221, observa-se o deslocamento mostrado na figura 4.18. Por ela pode-se confirmar a tendência ao giro apresentada pela formulação do elemento sapata. Se, por outro lado, for aplicado um momento, o deslocamento será o da figura 4.19. Também pode-se observar a translação obtida, se bem que em termos relativos muito menor que a rotação do caso anterior. Essa relação, aliás pode ser observada na matriz de rigidez da sapata antes da simetrização.

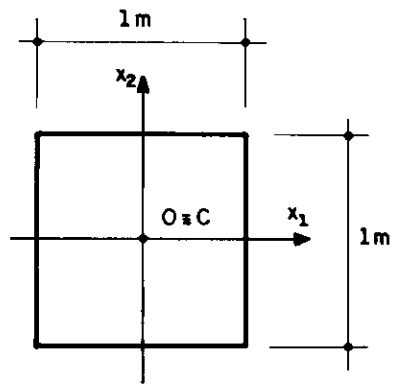


fig 4.16 Sapata 1m x 1m, centrada, sem rotação

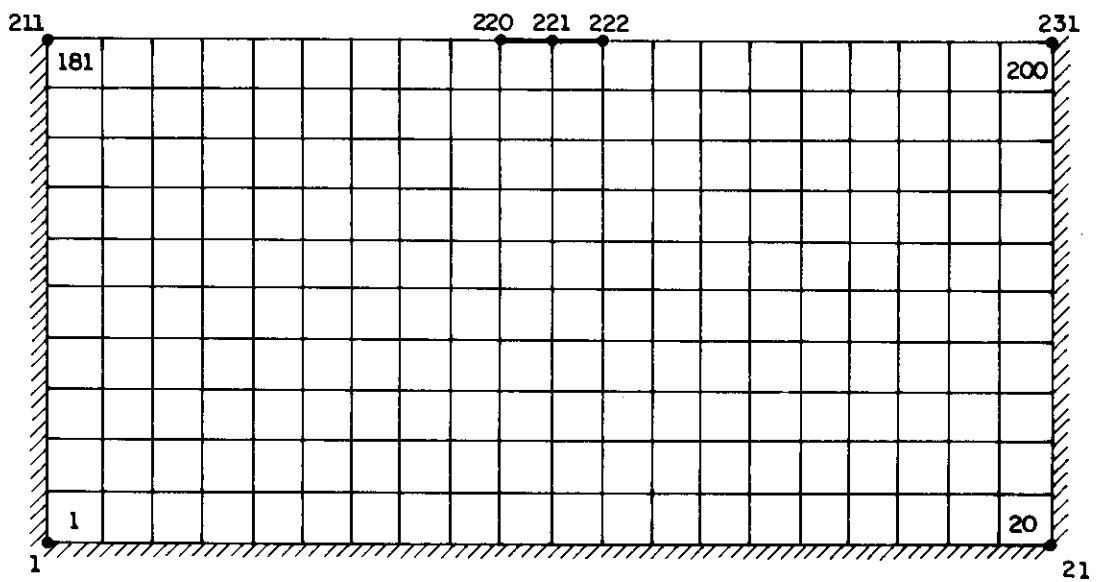


fig. 4.17 Rede de 231 nós, 200 elementos membrana e 2 elementos barra.

De qualquer modo, todos esses efeitos que não dizem respeito ao deslocamento ou rotação na própria coordenada onde a ação se verifica, são muito pequenos em termos relativos, não devendo ter grande importância na variabilidade dos esforços na superestrutura.

#### 4.4.3.2. Sapata 1m x 1m , centrada , com rotação de 45°

O exemplo aqui analisado encontra-se mostrado na figura 4.20. Trata-se da mesma sapata do exemplo anterior, somente que com uma rotação  $\beta = 45^\circ$  em relação ao sistema global de referência. Para tal caso, a matriz de rigidez obtida em relação ao centróide é a que se apresenta a seguir :

6729	0	0	0	-534	0
0	6729	0	534	0	0
0	0	8096	0	0	0
0	-46	0	2620	0	0
46	0	0	0	2620	0
0	0	0	0	0	4151

Como esperado a matriz é idêntica à obtida para o exemplo anterior. Na verdade a sapata quadrada comporta-se da mesma maneira, qualquer que seja o ângulo de rotação que apresente em relação ao sistema de referência global. Isso se verifica pois os momentos de inércia da sua área são invariantes em relação à referência adotada.

Portanto, o presente exemplo serve apenas como verificação de que o algoritmo adotado está produzindo resultados consistentes.

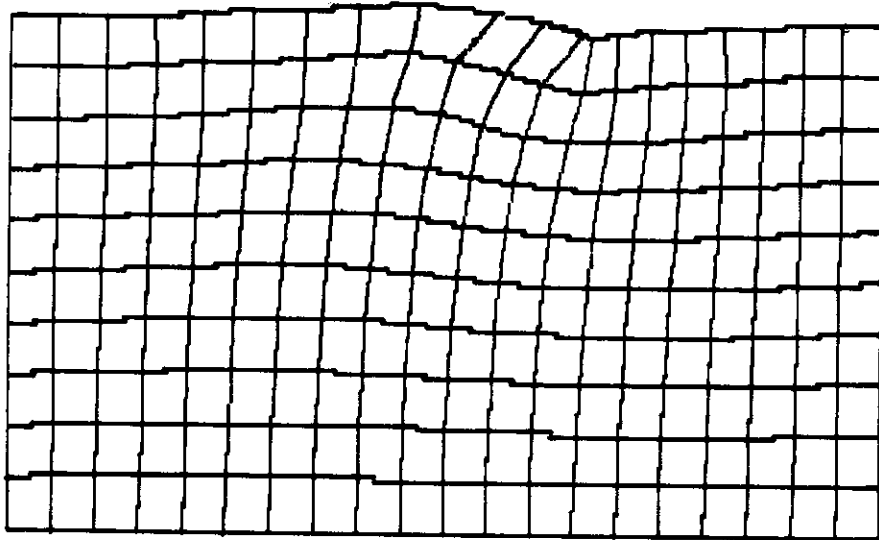


fig. 4.18 Deformada sob força horizontal

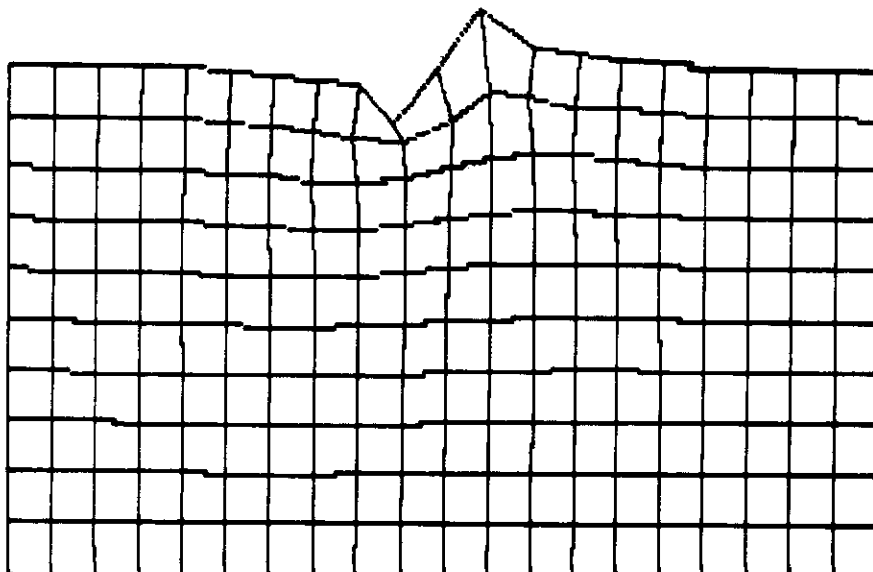


fig. 4.19 Deformada sob ação de um momento

#### 4.4.3.3. Sapata 1m x 1m , excêntrica segundo $X_1$ e $X_2$

O exemplo que se desenvolve neste item é o da figura 4.21. Trata-se da mesma sapata do item 4.4.3.1., apenas com a inclusão da excentricidade em relação aos eixos  $X_1$  e  $X_2$ . Evidentemente a matriz de rigidez em relação ao seu centróide será a mesma lá obtida. A diferença diz respeito à matriz quando referida ao ponto nodal, ou seja, após as transformações de translação mostradas no item 4.3.6. Nesse caso, a matriz de rigidez resulta na seguinte :

6729	0	0	0	-534	3364
0	6729	0	534	0	3364
0	0	8096	4048	-4048	0
0	-46	4048	4644	-2024	-23
46	0	-4048	-2024	4644	23
3364	3364	0	267	-267	7515

O primeiro detalhe que deve ser destacado é a existência de valores fora da diagonal principal de mesma ordem de grandeza dos que nela se encontram. Entretanto, esses valores são produzidos pelas transformações de translação e são perfeitamente simétricos em relação à diagonal. Os valores que não apresentam simetria continuam sendo muito menores que os diagonais, indicando que também neste caso a simetrização da matriz não faz com que se perca grandes informações a respeito do comportamento da sapata.

Interessante aqui é observar os coeficientes produzidos pela excentricidade da sapata. Note-se, por exemplo, que a ação de uma força segundo o eixo  $X_1$  produz não apenas deslocamento na própria direção  $X_1$  e rotação em relação a  $X_2$ , conforme foi visto no primeiro exemplo. Ocorre também uma rotação em torno de  $X_3$ . De modo semelhante, uma força em  $X_2$  produz deslocamento em  $X_2$ , rotação em torno de  $X_1$  e ainda rotação em torno de  $X_3$ .

Quando aplica-se uma força em  $X_3$  tem-se agora rotações em torno de  $X_1$  e  $X_2$ . Já um momento em torno de  $X_1$  produz agora deslocamento em  $X_3$ , rotação em torno de  $X_2$  e até uma pequena rotação em torno de  $X_3$ . Resultados semelhantes são observados pela ação de um momento em torno de  $X_2$ . Finalmente, um momento em torno de  $X_3$  resulta agora em quatro efeitos que inexistiam : deslocamentos em  $X_1$  e  $X_2$  e rotações em torno de  $X_1$  e  $X_2$ . Tudo isso além da rotação em torno do próprio  $X_3$ .

Também aqui pode-se aplicar forças e momentos segundo os eixos de referência. Imaginando-se forças de  $10^4$  KN e momentos de  $10^4$  KN.m , obtém para os deslocamentos e rotações mostrados na tabela 4.6, onde o cabeçalho é o mesmo da anterior.

CAR	$TX_1$	$TX_2$	$TX_3$	$RX_1$	$RX_2$	$RX_3$
1	2.093	.6022	.0694	-.0000	.1389	-1.204
2	.6022	2.093	.0694	-.1389	.0000	-1.204
3	.0694	.0694	3.149	-1.914	1.914	-.0000
4	-.0000	-.1389	-1.914	3.829	.0000	.0000
5	.1389	.0000	1.914	.0000	3.829	.0000
6	-1.204	-1.204	-.0000	.0000	.0000	2.409

tabela 4.6

#### 4.4.3.4. Sapata 1m x 1m , centrada , com distância $C \neq 0$

Neste item, pretende-se analisar a influência da consideração da sapata mergulhada no domínio semi-infinito de uma distância  $C$ . Portanto, pretende-se analisar a influência da consideração da solução fundamental de Mindlin. A sapata que se toma como exemplo é a mostrada na figura 4.22, idêntica à do primeiro exemplo.

É interessante ressaltar-se que o programa seleciona automaticamente a solução fundamental a ser utilizada. Se o

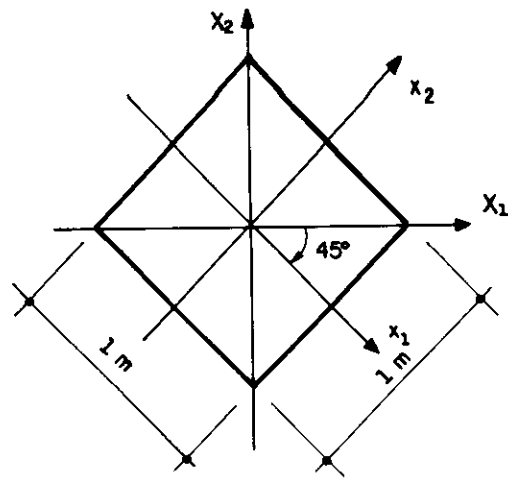


fig 4.20 Sapata 1 m x 1 m, centrada, com rotação de  $45^\circ$

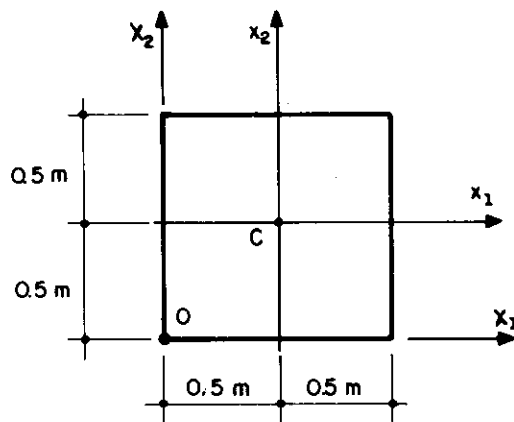


fig 4.21 Sapata 1 m x 1 m, excentrica segundo  $X_1$  e  $X_2$

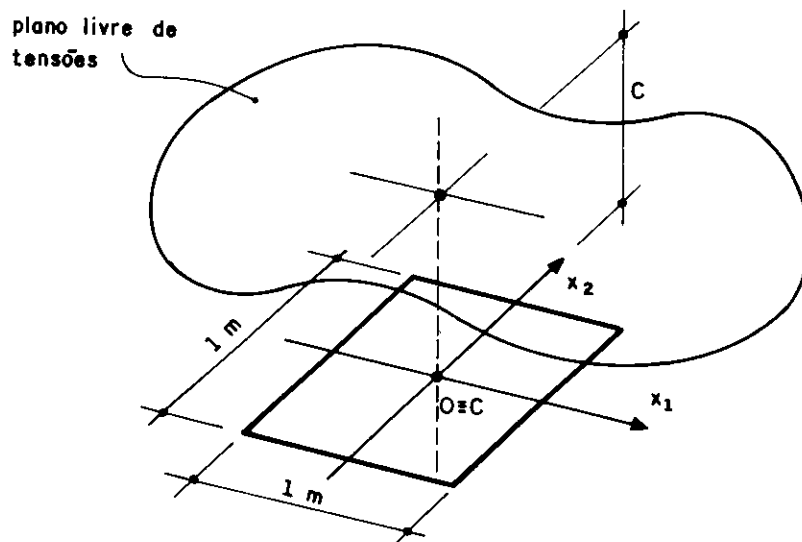


fig 4.22 Sapata 1 m x 1 m com  $C \neq O$

parâmetro  $C$  tem valor zero, é utilizada a solução de Boussinesq-Cerruti, caso contrário usa-se a solução de Mindlin. Assim, inicialmente adotar-se-á um valor muito pequeno para o parâmetro  $C$ , de modo a ter-se uma comparação com os resultados já obtidos no item 4.4.3.1. Fazendo-se  $C = 0.001$  m tem-se a seguinte matriz de rigidez no centroide, sem simetriação :

6741	0	0	0	-533	0
0	6741	0	533	0	0
0	0	8101	0	0	0
0	-38	0	2623	0	0
38	0	0	0	2623	0
0	0	0	0	0	4165

Como era esperado para esse caso os resultados são praticamente idênticos aos obtidos para o primeiro exemplo, já que para  $C = 0$  a solução fundamental de Mindlin e Boussinesq-Cerruti devem apresentar o mesmo valor.

Agora, para se verificar a influência da distância  $C$  nos resultados obtidos, toma-se por exemplo  $C = 2$ m, obtendo-se a seguinte matriz :

12289	0	0	0	85	0
0	12289	0	-85	0	0
0	0	12682	0	0	0
0	-80	0	5599	0	0
80	0	0	0	5599	0
0	0	0	0	0	8425

O que se pode perceber é que os coeficientes da diagonal praticamente dobraram de valor pelo fato da sapata encontrar-se 2m dentro do domínio semi-infinito. Isso significa que uma sapata nessas condições é, pelo menos teoricamente, muito mais rígida que uma outra que estivesse na superfície livre de tensões.



Outra observação muito importante é que os coeficientes fora da diagonal praticamente ficaram simétricos em relação a ela, além de resultarem valores muito pequenos. Isso quer dizer que a simetrização da matriz pelo critério do erro mínimo continua não introduzindo erros importantes.

#### 4.4.3.5 Sapata 0.5m x 1m , centrada , com rotação de 30°

A figura que mostra o presente exemplo é a 4.23. Trata-se de uma sapata retangular com uma rotação de 30° em relação aos eixos de referência globais. Neste caso, ao contrário do exemplo apresentado no item 4.4.3.2., tal rotação provoca alterações na matriz de rigidez em relação ao centróide da sapata, como pode-se observar pela apresentação da referida matriz sem simetrização :

5080	-233	0	64	-256	0
-233	4811	0	330	-64	0
0	0	5993	0	0	0
-83	-102	0	1305	-372	0
6	83	0	-372	875	0
0	0	0	0	0	1739

É interessante observar-se que os valores dos coeficientes que estão presentes na matriz e resultariam nulos caso o ângulo de rotação fosse zero são também muito pequenos em relação aos da diagonal. Além disso, os que apresentam valores um pouco mais significativos são simétricos em relação a ela. Portanto, encerrando-se esta série de exemplos de sapatas isoladas, pode-se concluir que neste caso o processo de simetrização adotado não deve prejudicar de forma sensível os resultados a serem obtidos para as superestruturas.

#### 4.4.4. Sapatas com Interação

##### 4.4.4.1. Duas sapatas 1m x 1m , separadas de 2.5 m

O conjunto de sapatas que se analisa neste item é o apresentado na figura 4.24. Trata-se de duas sapatas de 1m x 1m , separadas por 2m. A matriz de rigidez obtida para este caso é a apresentada a seguir :

7438	0	32	0	-590	0	-2205	0	347	0	203	0
0	7029	0	548	0	55	0	-1410	0	-66	0	260
26	0	8602	0	-69	0	-336	0	-2000	0	-229	0
0	-58	0	2620	0	-2	0	52	0	-35	0	-2
43	0	-71	0	2630	0	40	0	268	0	59	0
0	55	0	2	0	4163	0	-258	0	-25	0	21
-2205	0	-347	0	203	0	7438	0	-32	0	-590	0
0	-1410	0	-66	0	-260	0	7029	0	548	0	-55
336	0	-2000	0	229	0	-26	0	8602	0	69	0
0	52	0	-35	0	2	0	-58	0	2620	0	2
40	0	-268	0	59	0	43	0	71	0	2630	0
0	258	0	25	0	21	0	-55	0	-2	0	4163

Observando-se a matriz, nota-se que, a exemplo de todas as sapatas isoladas que foram analisadas, a simetria em relação à diagonal principal continua aproximadamente mantida. Os valores fora da diagonal e com significância numérica ( linha um e coluna sete, linha dois e coluna oito, linha três e coluna 9, além das posições que lhes são correspondentes na parte inferior da matriz ) são perfeitamente simétricos. Assim, como seria de se esperar, também para sapatas com interação a simetrização posterior da matriz de rigidez não traz maiores imprecisões.

O ponto de grande importância, a ser discutido com o exemplo aqui apresentado, é o nível de interação entre as sapatas. A separação entre os centróides dessas sapatas consideradas é de 2.5 vezes a dimensão dos seus lados. Entretanto, como pode-se notar pelos valores apresentados pela matriz, a interação é relativamente forte, principalmente quando consideram-se ações de forças segundo

os eixos  $X_1$ ,  $X_2$  e  $X_3$ . Nesses casos os coeficientes obtidos para uma sapata, pela aplicação das ações na outra, são da ordem de 25% dos obtidos para a própria sapata onde é aplicada a ação. Já para os momentos, pode-se concluir que a interação não é significativa. Logicamente os comentários são válidos para a referida separação entre centróides.

#### 4.4.4.2. Duas sapatas 1m x 1m , separadas de 5m

A figura onde mostra-se o presente exemplo é a 4.25. Trata-se de um esquema muito semelhante ao do item anterior, com a ressalva de que a separação entre os centróides das sapatas é o dobro, ou seja, 5 vezes a dimensão da sapata. A matriz de rigidez obtida é a seguinte:

6881	0	6	0	-546	0	-1002	0	171	0	89	0
0	6792	0	538	0	6	0	-648	0	-40	0	61
6	0	8208	0	-8	0	-168	0	-934	0	-47	0
0	-48	0	2620	0	-0	0	15	0	-3	0	-0
47	0	-7	0	2621	0	6	0	62	0	7	0
0	6	0	0	0	4151	0	-61	0	-5	0	3
-1002	0	-171	0	89	0	6881	0	-6	0	-546	0
0	-648	0	-40	0	-61	0	6792	0	538	0	-6
168	0	-934	0	47	0	-6	0	8208	0	8	0
0	15	0	-3	0	0	0	-48	0	2620	0	0
6	0	-62	0	7	0	47	0	7	0	2621	0
0	61	0	5	0	3	0	-6	0	-0	0	4151

Para este caso, a interação está limitada a patamares bem mais modestos. Os coeficientes obtidos para as posições já mencionadas no item anterior são aproximadamente 10% dos valores da diagonal. Além disso, os próprios valores da diagonal são muito semelhantes aos obtidos para o exemplo do item 4.4.3.1, sapata isolada.

Com a informação de que em outros exemplos pesquisados, para a separação de 5 vezes a maior dimensão das sapatas, o nível de interação permaneceu sempre nesse mesmo patamar, pode-se concluir que, a rigor, apenas a

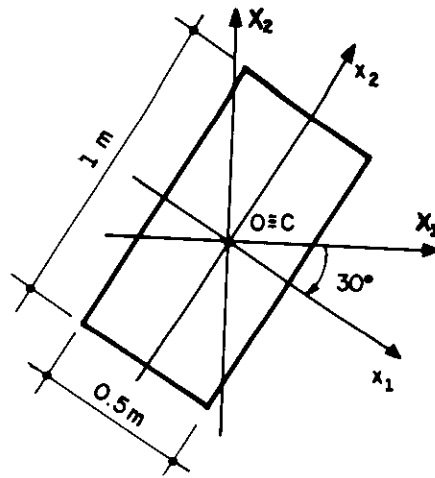


fig 4.23 Sapata 0.5m x 1m, centrada, com rotação de  $30^\circ$

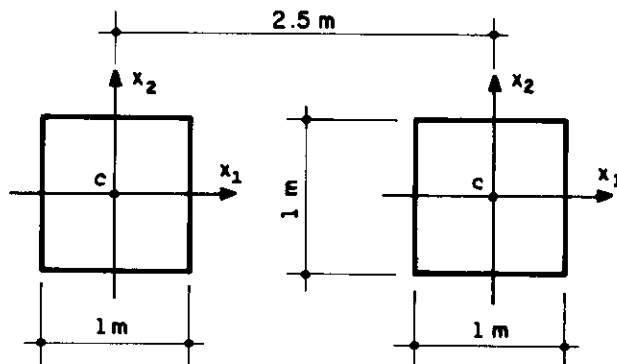


fig 4.24 Duas sapatas de 1m x 1m, separadas de 2.5m

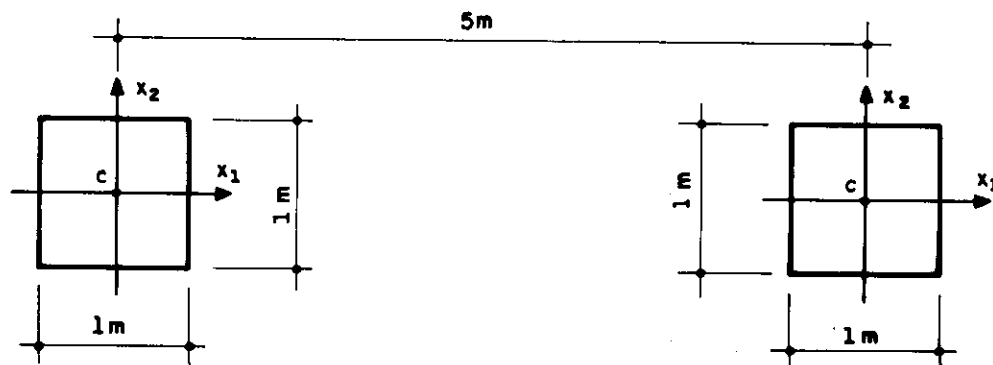


fig 4.25 Duas sapatas 1m x 1m, separadas de 5m

partir dessa distância é que se pode desconsiderar esse efeito. Entretanto, deve-se lembrar que uma formulação que considera o solo como um meio isotrópico, homogêneo e perfeitamente elástico, já apresenta grande discrepância em relação à realidade. Assim sendo, nem sempre é justificável um excesso de preocupação com determinados pontos específicos.

#### 4.4.4.3. Duas sapatas 0.5m x 1m , separadas de 2.5m

O exemplo aqui apresentado vem demonstrar que o parâmetro que deve ser considerado para a decisão de considerar-se ou não a interação entre sapatas é realmente a distância entre os centróides, e não a simples distância entre os extremos das sapatas. Para tanto vai-se tomar o conjunto de duas sapatas de 0.5m x 1m separadas de 2.5m segundo os seus menores lados, conforme a figura 4.26, ou segundo os seus maiores lados, conforme a figura 4.27. Considerando a primeira hipótese a matriz de rigidez obtida é a que se segue :

5489	0	11	0	-230	0	-1201	0	180	0	56	0
0	4762	0	371	0	8	0	-632	0	-34	0	58
11	0	6172	0	-8	0	-188	0	-1022	0	-35	0
0	-155	0	1520	0	-0	0	37	0	-8	0	-0
-45	0	-8	0	661	0	18	0	42	0	3	0
0	7	0	0	0	1740	0	-57	0	-7	0	-9
-1201	0	-180	0	56	0	5489	0	-11	0	-230	0
0	-632	0	-34	0	-58	0	4762	0	371	0	-8
188	0	-1022	0	35	0	-11	0	6172	0	8	0
0	37	0	-8	0	0	0	-155	0	1520	0	0
18	0	-42	0	3	0	-45	0	8	0	661	0
0	57	0	7	0	-9	0	-7	0	-0	0	1740

Já se o esquema adotado for o da separação segundo os maiores lados, portanto com uma separação entre extremos de apenas 1.5m ao invés dos 2m do caso anterior, a matriz resulta :

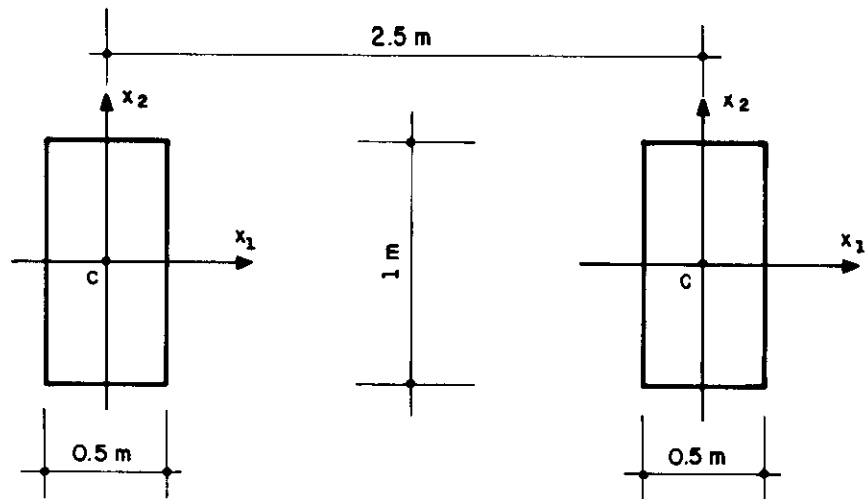


fig 4.26 Sapatas separadas de 2.5m segundo  $x_1$

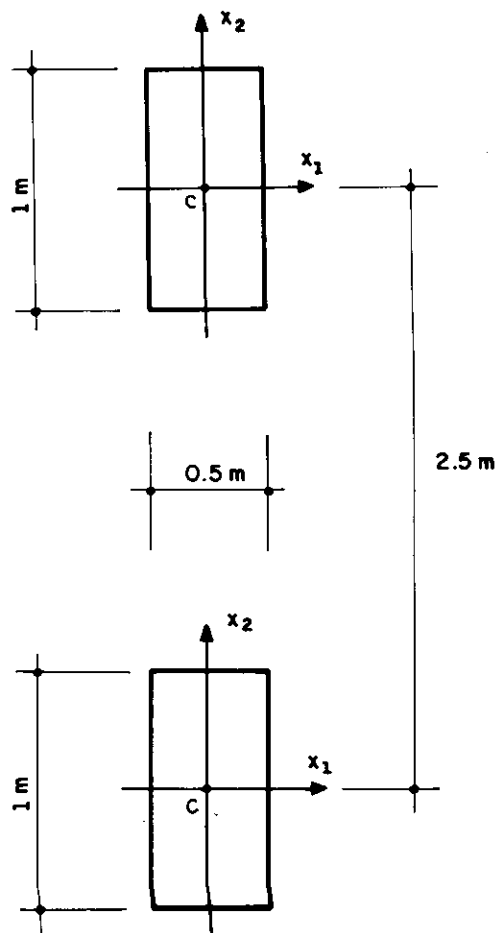


fig 4.27 Sapatas separadas de 2.5m segundo  $x_2$

5348	0	0	0	-222	-17	-827	0	0	0	24	-100
0	4916	10	385	0	0	0-1045	195	-96	0	0	0
0	7	6201	23	0	0	0	-179-1090	100	0	0	0
0	-156	23	1522	0	0	0	8	-125	24	0	0
-42	0	0	0	661	0	-2	0	0	0	-2	-0
-17	0	0	0	0	1741	102	0	0	0	-3	18
-827	0	0	0	24	100	5348	0	0	0	-222	17
0	-1045	-195	-96	0	0	0	4916	-10	385	0	0
0	179-1090	-100	0	0	0	0	-7	6201	-23	0	0
0	8	125	24	0	0	0	-156	-23	1522	0	0
-2	0	0	0	-2	0	-42	0	0	0	661	0
-102	0	0	0	3	18	17	0	0	0	-0	1741

Como pode-se observar, o nível de interação obtido é realmente muito semelhante ao do primeiro esquema, comprovando-se aqui também o que já se tinha notado em uma série de outros exemplos.

#### 4.4.5. Influência das Sapatas na Superestrutura

Este item é de grande importância para o estudo aqui realizado. Nele tenta-se responder, pelo menos parcialmente, à pergunta : qual a importância para os esforços nas estruturas da consideração da flexibilidade das fundações ?. Não é um objetivo fácil de ser alcançado. Entretanto, é fundamental que a tentativa seja feita de maneira séria e profissional e para isso foram adotados alguns procedimentos muito bem definidos, que encontram-se explicitados a seguir.

Inicialmente, adotou-se a utilização de exemplos reais, ou seja, de estruturas que efetivamente tenham sido projetadas e construídas. Procurou-se também manterem-se os principais procedimentos observados nos bons escritórios de projetos, de modo que os resultados obtidos sejam por eles influenciados. Ressalta-se que não se procura uma situação ideal, que por estar distante da prática cotidiana, não permitiria a obtenção das conclusões desejadas. Por fim, menciona-se que a análise dos resultados é feita apenas

sobre os parâmetros principais, de modo a não se dispender esforços com detalhes de menor importância.

#### 4.4.5.1. Edifício Maison Mouette - Ação do Vento

O edifício aqui considerado tem a forma do pavimento tipo apresentada na figura 4.28. Trata-se de uma obra na cidade de S.Paulo que possui 1 subsolo, 1 pavimento térreo e 21 pavimentos tipos.

Para a análise da estrutura sob a ação de cargas horizontais provocadas pelo vento segundo o eixo de simetria, montou-se um modelo de pórtico tridimensional com 314 nós e 460 elementos barra, além de 9 elementos sapata para simular a fundação. Destaca-se que os nós de um pavimento estão obrigados a apresentar o mesmo deslocamento no seu plano, de modo a simular a existência da laje através de um diafragma rígido

É interessante ressaltar que tanto a forma dos pavimentos quanto a própria fundação foram projetados por profissionais das respectivas especialidades. No caso, a taxa de trabalho do solo foi adotada como sendo de  $600 \text{ KN/m}^2$ . Já o módulo de elasticidade e coeficiente de Poisson foram estimados em  $50\ 000 \text{ KN/m}^2$  e  $0.35$ , respectivamente. Para o concreto, a resistência característica à compressão,  $f_{ck}$ , foi estimada em  $18 \text{ MPa}$  e módulo de elasticidade para ações instantâneas foi adotado como sendo de  $2.7 \times 10^7 \text{ KN/m}^2$ .

Convém ressaltar que o edifício analisado neste exemplo configura-se como um caso onde espera-se uma influência relativamente grande da consideração da flexibilidade da fundação. Isso porque sendo a forma do pavimento tipo em laje cogumelo, acabam prevalecendo pilares de grandes dimensões, que tem a tendência de apresentarem significativos momentos fletores na base. Se a fundação não for muito bem projetada de modo a resistir a esses grandes momentos, na verdade eles serão



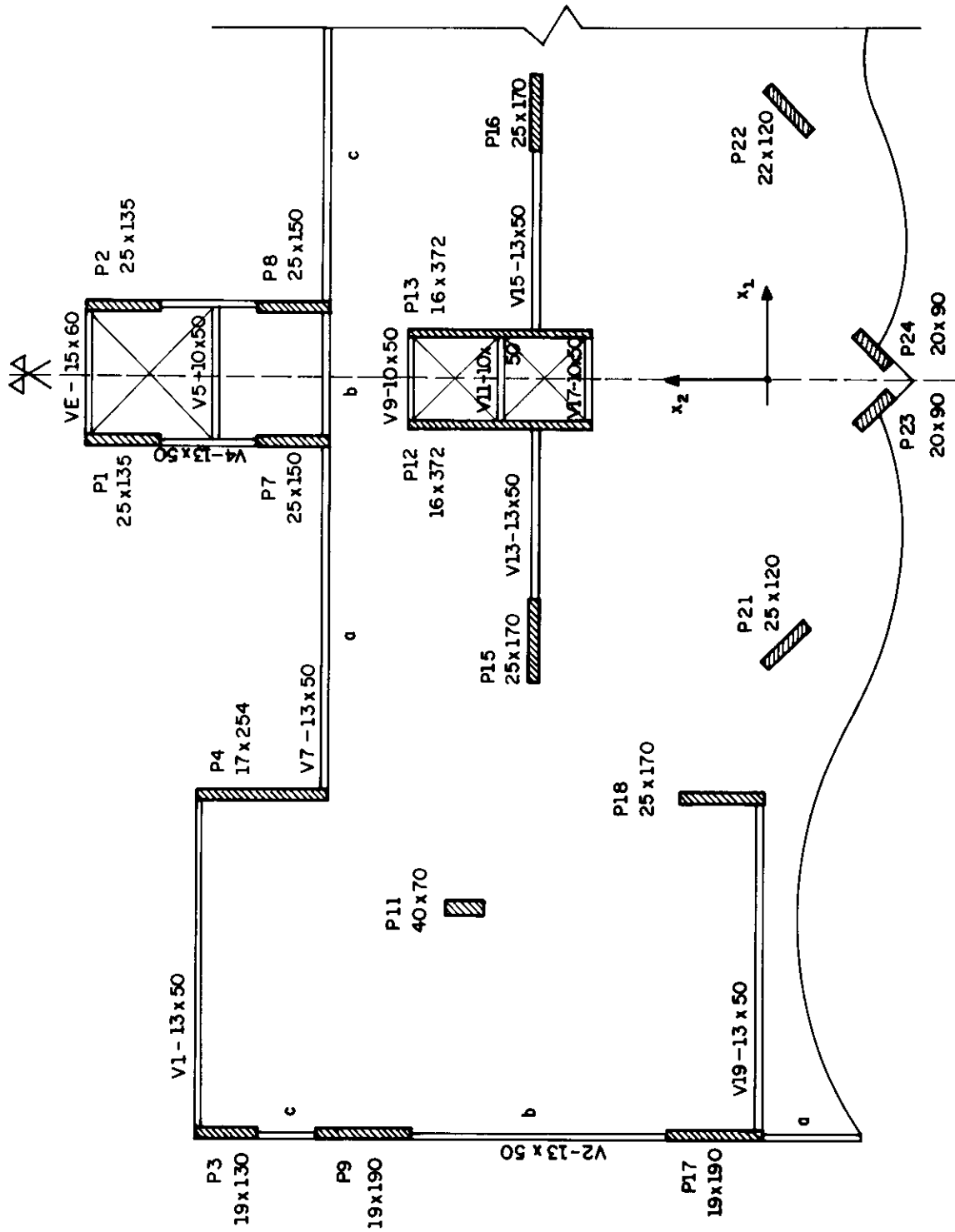


fig 4.28 Forma do pavimento tipo do Ed. Maison Mouette

redistribuídos, alterando-se bastante os esforços obtidos.

Para uma comparação mais efetiva dos resultados, o referido pórtico tridimensional foi calculado para três situações distintas no que diz respeito ao comportamento da fundação. Em primeiro lugar adotou-se uma fundação completamente rígida. Após essa análise, foram então definidas as sapatas e adotado um solo com características  $E = 100\ 000\ \text{KN/m}^2$  e  $\nu = 0.35$ . Por fim, o módulo de elasticidade foi reduzido para  $10\ 000\ \text{KN/m}^2$ .

É interessante ressaltar que esses valores de módulo de elasticidade representam de certo modo os limites possíveis. Entretanto, dificilmente adotar-se-á uma fundação em sapata para solos de pequena capacidade de resistência e conseqüentemente de módulo de elasticidade muito baixo. Assim sendo, o valor  $10\ 000\ \text{KN/m}^2$  é apenas um valor de referência, não se devendo considerar que estruturas reais possam encontrar-se apoiadas sobre sapatas em solos tão pouco rígidos.

Alguns dos resultados obtidos para essas análises estão compilados nas tabelas 4.7 a 4.11, onde os três níveis de rigidez mencionados são chamados de "Fund. Rig", "F.Flex E=100000" e "F.Flex E=10000".

-----			
Normais nos Pilares Junto à Fundação ( KN )			
Pilar	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
-----			
P1	2 278.0	1 667.1	718.8
P3	1 427.4	1 251.0	715.3
P4	7.2	35.6	334.7
P7	2 199.2	1 552.2	245.7
P9	803.5	514.4	184.4
P12	185.3	198.8	130.3
P15	185.3	198.8	130.3
P17	637.3	770.2	1 069.7
P18	72.7	116.8	637.7
-----			

tabela 4.7

Momentos nos Pilares Junto à Fundação ( KN·m )			
Pilar	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
P1	441.5	557.9	428.3
P3	303.8	320.9	213.4
P4	1 976.9	2 417.7	1 871.6
P7	620.6	1 023.7	1 006.1
P9	933.6	1 272.6	1 043.0
P12	5 479.2	3 908.0	1 986.0
P15	19.2	71.1	460.0
P17	925.6	1 694.4	1 939.4
P18	869.1	1.554.2	1 678.8

tabela 4.8

Cortantes nas Vigas do Pav. Térreo ( KN )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V2b	8.21	15.68	62.28
V2c	33.05	4.64	12.83
V4	66.46	37.16	98.06

tabela 4.9

Momentos nas Vigas do Pav. Térreo ( KN·m )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V2b	28.71	54.77	216.18
V2c	44.70	5.51	14.75
V4	116.95	65.36	180.04

tabela 4.10

Deslocamentos dos Pavimentos ( cm )			
Pav.	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
Térreo	0.15	0.43	4.64
1ª Tipo	0.57	1.14	9.08
2ª Tipo	1.09	1.95	13.04
3ª Tipo	1.74	2.88	17.09
4ª Tipo	2.51	3.93	21.23
5ª Tipo	3.36	5.07	25.42
6ª Tipo	4.29	6.30	29.66
7ª Tipo	5.27	7.57	33.92
8ª Tipo	6.30	8.89	38.20
9ª Tipo	7.36	10.25	42.49
10ª Tipo	8.45	11.63	46.77
11ª Tipo	9.56	13.02	51.05
12ª Tipo	10.67	14.42	55.31
13ª Tipo	11.79	15.82	59.55
14ª Tipo	12.91	17.21	63.78
15ª Tipo	14.02	18.60	67.98
16ª Tipo	15.12	20.00	72.16
17ª Tipo	16.21	21.33	76.31
18ª Tipo	17.28	22.67	80.43
19ª Tipo	18.34	24.00	84.55
20ª Tipo	19.39	25.31	88.62
21ª Tipo	20.42	26.60	92.88
Cober.	21.44	27.89	96.73

tabela 4.11

Observando-se os resultados obtidos nota-se que a influência da consideração da flexibilidade da fundação nos esforços da superestrutura é muito grande. Mesmo considerando-se o módulo de elasticidade de 100 000 KN/m<sup>2</sup>, portanto um solo muito rígido, a variação em alguns casos é bastante significativa, indicando que pelo menos na

estrutura analisada a influência da flexibilidade da fundação deveria estar sendo considerada no dimensionamento de suas peças.

Nos pilares, como regra geral, os esforços normais e momentos tendem a uma redistribuição que torne os seus valores menos díspares. Isso pode ser observado com muita clareza nas tabelas 4.7 e 4.8, onde os menores valores tendem a aumentar e os maiores a diminuir. Além disso as mudanças nesses valores são realmente muito significativas, bastando observar-se, por exemplo, o comportamento da normal nos pilares P4 e P7 e do momento nos pilares P12 e P15. No caso da força normal, o P7 sofre uma redução de -89% e o P4 um aumento de 4439%. Quanto aos momentos, o P12 apresenta uma redução de -64% e o P15 um aumento de 2196%.

Quanto ao comportamento das vigas do pavimento Térreo, não se pode por este exemplo tirar uma conclusão mais definida, principalmente devido ao fato de existirem muito poucas peças trabalhando para resistir ao vento na direção considerada.

Para o deslocamento dos pavimentos na direção da ação aplicada, verifica-se que a fundação também influencia de forma marcante. Pode-se afirmar que para um módulo de elasticidade de  $50\ 000\ \text{KN/m}^2$ , precisamente o valor estimado para o solo neste caso, a flecha no topo seria pelo menos o dobro do valor calculado com a fundação rígida, sendo esse acréscimo causado pelas rotações que se verificam na base da estrutura. O que não muda de forma significativa é o aspecto das deformadas, que são apresentadas na figura 4.29. Provavelmente por ausência de um número significativo de vigas na direção atuante do vento, o comportamento da estrutura, que era tipicamente parede com a fundação rígida, não mudou com a introdução da flexibilidade, continuando um comportamento de parede com grandes deslocamentos. A flecha no topo cresceu 30% para  $E = 100\ 000\ \text{KN/m}^2$  e 350% para  $E = 10\ 000\ \text{K/m}^2$ .

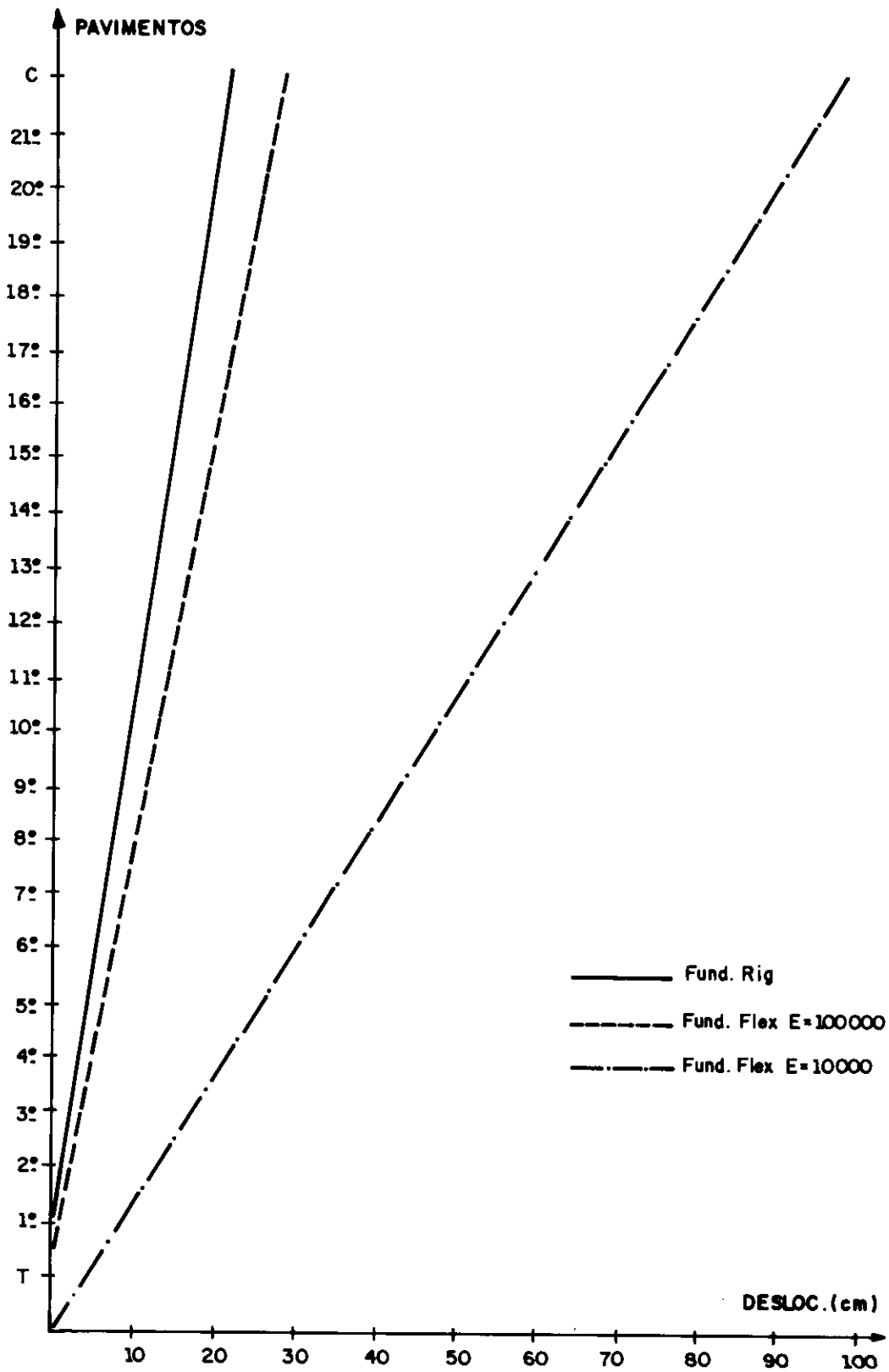


fig 4.29 Deformadas do eixo vertical do Ed Maison Mouette

#### 4.4.5.2. Edifício Arte 5 - Ação do Vento

O edifício aqui considerado tem a forma do pavimento tipo apresentada na figura 4.30. Trata-se de uma obra na cidade de Goiânia que possui 1 subsolo, 1 pavimento térreo e 15 pavimentos tipos.

Para a análise da estrutura sob a ação de cargas horizontais provocadas pelo vento segundo o eixo de simetria, montou-se um modelo de pórtico tridimensional com 274 nós e 464 elementos barra, além de 14 elementos sapata para simular a fundação. Destaca-se que, da mesma maneira que no exemplo anterior, os nós de um pavimento também estão obrigados a apresentar o mesmo deslocamento no seu plano.

Também aqui, tanto a forma dos pavimentos quanto a própria fundação foram projetados por profissionais das respectivas especialidades. A taxa de trabalho para o solo foi adotada como sendo de  $550 \text{ KN/m}^2$ . O módulo de elasticidade e coeficiente de Poisson foram estimados em  $50\ 000 \text{ KN/m}^2$  e  $0.35$ , respectivamente. Para o concreto, a resistência característica à compressão adotada foi de  $21 \text{ MPa}$  e módulo de elasticidade para ações instantâneas  $2.7 \times 10^7 \text{ KN/m}^2$ .

A forma do pavimento tipo do edifício analisado neste item representa o que se pode chamar de uma estrutura bem comportada quanto à ação do vento. Trata-se, diferentemente do caso anterior, de uma estrutura convencional absolutamente típica para uma edificação nessas condições. As normais e momentos obtidos para os pilares neste caso devem ser significativamente menores que as obtidas para o exemplo anterior. Pode-se supor portanto que este não é um exemplo onde a interação com a fundação tenha especial importância.

De modo semelhante ao caso anterior, o pórtico tridimensional também foi calculado para três situações

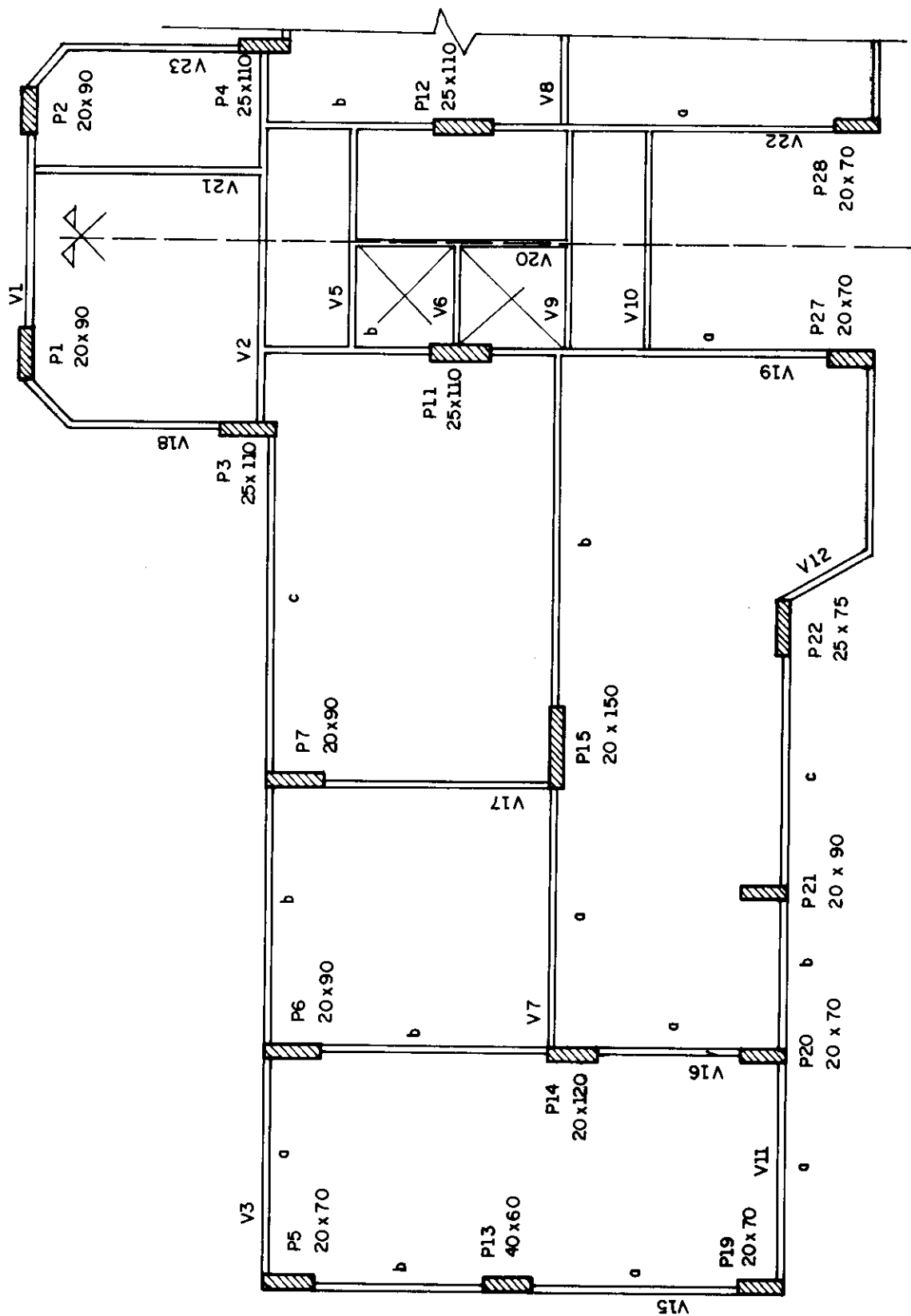


fig 4.30 Forma do pavimento tipo do Ed. Arte 5



distintas de fundação. Inicialmente, adotou-se a fundação completamente rígida. Depois, foram definidas as sapatas e adotados solos com características  $E = 100\ 000\ \text{KN/m}^2$  e  $E = 10\ 000\ \text{KN/m}^2$ . Nos dois casos, manteve-se o valor do coeficiente de Poisson em 0.35.

Os principais resultados obtidos para esta análise encontram-se nas tabelas 4.12 a 4.16, onde os títulos dos cabeçalhos são os mesmos do caso anterior.

Normais nos Pilares Junto à Fundação ( KN )			
Pilar	Fund. Rig.	F. Flex $E=100000$	F. Flex $E=10000$
P3	79.1	94.6	228.8
P5	473.3	437.4	285.6
P6	420.5	451.1	435.0
P7	354.2	357.0	391.3
P11	225.1	234.8	255.6
P13	1.3	7.9	6.3
P14	159.3	89.5	62.3
P15	335.1	303.5	124.8
P19	453.6	435.6	320.2
P20	538.0	462.8	328.3
P21	95.2	150.0	293.1
P22	65.9	85.6	218.3
P27	225.1	234.8	255.6

tabela 4.12

Momentos nos Pilares Junto à Fundação ( KN·m )			
Pilar	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
P3	490.9	358.7	171.2
P5	111.1	131.6	110.7
P6	224.1	304.9	367.0
P7	224.1	293.5	310.1
P11	643.7	455.3	213.3
P13	147.1	204.2	283.9
P14	530.9	409.9	206.9
P15	25.7	386.0	119.2
P19	111.3	120.7	84.8
P20	112.8	141.3	137.7
P21	215.2	243.3	188.3
P22	17.3	28.3	74.4
P27	107.8	138.8	138.1

tabela 4.13

Cortantes nas Vigas do Pav. Térreo ( KN )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V15a	28.60	30.50	32.76
V15b	28.60	30.01	31.03
V16a	40.72	39.15	47.66
V16b	21.73	24.94	33.08
V17	23.10	25.10	32.03
V19a	11.72	13.02	17.76

tabela 4.15

Momentos nas Vigas do Pav. Térreo ( KN.m )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V15a	56.00	59.60	65.46
V15b	55.30	58.01	59.13
V16a	65.42	63.75	76.26
V16b	51.13	58.14	78.68
V17	45.00	49.50	66.83
V19a	37.32	41.92	56.06

tabela 4.16

Deslocamentos dos Pavimentos ( cm )			
Pav.	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
Térreo	0.25	0.38	2.23
1º Tipo	0.77	0.98	4.18
2º Tipo	1.42	1.71	6.24
3º Tipo	2.14	2.50	8.33
4º Tipo	2.88	3.32	10.42
5º Tipo	3.62	4.13	12.49
6º Tipo	4.33	4.90	14.53
7º Tipo	5.00	5.64	16.53
8º Tipo	5.62	6.33	18.47
9º Tipo	6.20	6.97	20.36
10º Tipo	6.71	7.55	22.19
11º Tipo	7.16	8.06	23.95
12º Tipo	7.56	8.53	25.66
13º Tipo	7.91	8.93	27.30
14º Tipo	8.20	9.29	28.91
Cober.	8.46	9.62	30.47

tabela 4.14

Para a estrutura do Ed. Arte 5, a influência da flexibilidade da fundação é sensivelmente menor que a do exemplo anterior. Assim mesmo, existem algumas peças que apresentam grandes variações de esforços, mas variações não tão significativas, até porque os valores obtidos são relativamente pequenos quando comparados com os esforços que devem resultar da consideração de outros carregamentos.

Ao observarem-se com mais detalhes as mencionadas tabelas, verifica-se que para os pilares ainda se mantém, como regra geral, a tendência de uma maior homogeneidade nos esforços através da diminuição dos valores mais elevados e um aumento dos mais baixos. Assim ocorre, por exemplo, com as forças normais dos pilares P20 e P22 e momentos fletores dos pilares P11 e P15. Mas aqui já aparecem algumas excessões a essa tendência, como as forças normais do P6 e P27 e momentos fletores dos pilares P6 e P7. Mesmo assim, as variações extremas que podem ser observadas são sensivelmente menores que as do caso anterior. Por exemplo, quanto às forças normais, a maior diminuição ocorre no P15, com - 63%, e o maior aumento no P22, com 131%. Já quanto aos momentos fletores, a maior diminuição verifica-se no P11, com -67%, e o maior aumento, com valores significativos, no P6 com 64%.

Quanto aos esforços atuantes nas vigas, verifica-se um aumento desses valores, seja na cortante ou no momento, para todas as peças que trabalham no suporte da ação do vento. Entretanto essas variações não se apresentam tão grandes quanto nos pilares. Em relação à cortante a maior variação ocorre na viga V16b, com 64%. Considerando-se o momento, tem-se na mesma peça uma variação de 54%. Pode-se tentar explicar essas variações como uma mudança de comportamento da associação paredes e pórticos que compõe a estrutura de travamento aos esforços horizontais. Com a flexibilização da fundação, diminuem os momentos nos pilares junto à base, passando a estrutura a trabalhar preferencialmente como pórtico. Com isso, os esforços nas

vigas são aumentados. Essa suposição pode ser de certo modo comprovada pela observação da figura 4.31, onde mostram-se as deformadas da estrutura. Nota-se que para  $E = 10\ 000\ \text{KN/m}^2$ , a forma da elástica junto à base já não é típica de uma associação com presença de paredes.

Percebe-se que também para os deslocamentos dos pavimentos, a influência da fundação é menor neste exemplo do que para o apresentado no item anterior. Com relação à flecha no topo da estrutura, tem-se para este caso um acréscimo de somente 13%, para o módulo de elasticidade de  $100\ 000\ \text{KN/m}^2$ , e 260%, para o módulo de  $10\ 000\ \text{KN/m}^2$ . Entretanto, aqui ocorre uma alteração na forma da deformada, quando se considera a flexibilidade da fundação com o módulo de elasticidade mais baixo. A forma típica de uma associação de paredes e pórticos cede lugar a uma elástica típica de pórtico apenas.

#### 4.4.5.3. Edifício Arte 5 - Cargas Verticais

Os dois exemplos anteriores tiveram por objetivo mostrar o comportamento de estruturas submetidas a cargas horizontais interagindo com a fundação. Entretanto, os edifícios são ainda submetidos a pelo menos um outro carregamento muito importante. Tratam-se evidentemente das ações verticais, que dizem respeito ao peso próprio e demais sobrecargas atuantes.

Para tentar verificar-se qual a influência da flexibilidade da fundação com relação a esse carregamento, tomou-se o mesmo edifício do exemplo anterior. Sua estrutura, cuja forma apresenta-se na figura 4.30, continua sendo discretizada pelo pórtico lá mencionado, somente que agora é solicitado pelo conjunto das cargas verticais.

Para tanto os nós do pórtico tridimensional foram carregados com as reações nos pilares obtidas para o cálculo do pavimento feito de maneira independente. Esse

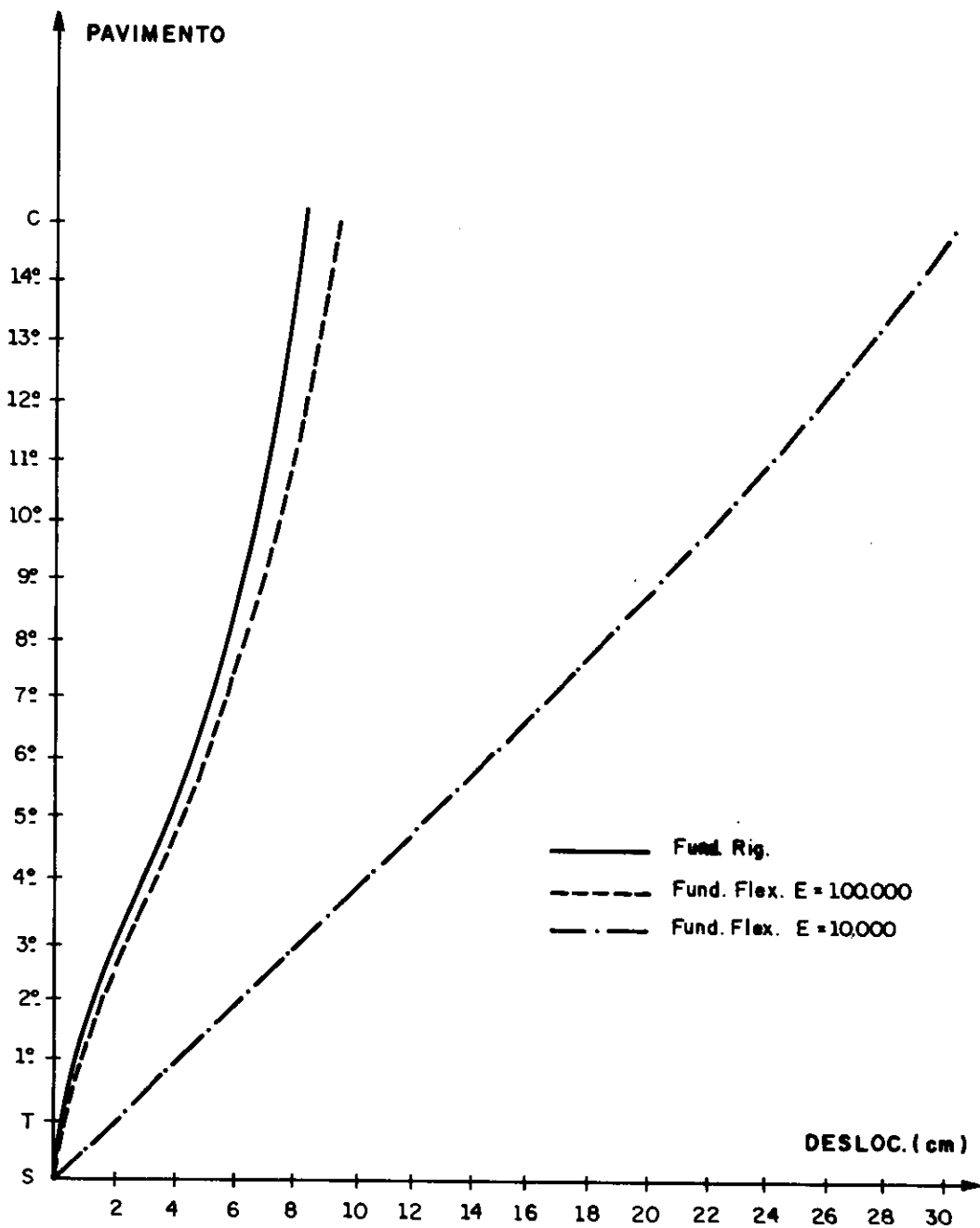


fig 4.31 Deformadas do eixo vertical do Ed. Arte 5

procedimento, que absolutamente não faz parte do roteiro normal de cálculo dos escritórios de projeto, deve permitir que se estime o erro que se comete nos esforços dos pilares e vigas pelo cálculo dos pavimentos como se fossem independentes entre si e independentes da fundação.

De modo semelhante ao casos anteriores, também aqui faz-se o cálculo da estrutura considerando-se a fundação rígida, a fundação flexível com  $E = 100\ 000\ \text{KN/m}^2$  e finalmente a fundação flexível com  $E = 10\ 000\ \text{KN/m}^2$ . Os principais resultados obtidos estão nas tabelas 4.15 a 4.18., apresentadas a seguir.

Normais nos Pilares Junto à Fundação ( KN )			
Pilar	Fund. Rig.	F. Flex $E=100000$	F. Flex $E=10000$
P1	1 059.2	1 078.3	1 083.2
P3	2 246.2	2 211.4	2 050.7
P5	630.5	693.2	916.4
P6	1 298.7	1 410.6	1 827.3
P7	1 850.3	2 051.1	2 518.2
P11	3 306.2	3 226.3	2 833.2
P13	1 673.3	1 609.9	1 606.6
P14	2 386.3	2 212.5	1 844.5
P15	4 087.2	3 794.3	2 799.6
P19	880.2	942.9	1 147.7
P20	1 088.1	1 193.0	1 366.6
P21	1 377.2	1 444.3	1 596.8
P22	1 493.2	1 448.3	1 336.9
P27	959.4	1 039.3	1 432.4

tabela 4.15

Cortantes nas Vigas do Pav. Térreo ( KN )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V3a	0.93	1.68	8.92
V3b	0.93	2.26	17.64
V3c	0.48	2.45	13.75
V7a	0.94	6.72	33.13
V11a	0.66	2.88	18.39
V11b	0.39	1.34	5.17
V11c	0.19	3.90	13.12
V15a	0.46	3.20	2.66
V15b	1.53	6.03	13.90
V16a	2.31	18.38	58.60
V16b	0.99	8.60	32.20
V17	1.49	17.27	62.47
V19a	0.59	6.05	32.82

tabela 4.16

Momentos nas Vigas do Pav. Térreo ( KN·m )			
Viga	Fund. Rig.	F. Flex E=100000	F. Flex E=10000
V3a	1.86	3.57	20.15
V3b	2.24	5.21	39.64
V3c	1.47	7.52	43.16
V7a	2.50	17.98	89.01
V11a	1.32	5.66	36.75
V11b	0.70	2.06	10.28
V11c	0.39	8.47	28.74
V15a	0.93	6.39	5.42
V15b	2.98	11.82	27.17
V16a	3.80	29.41	96.91
V16b	2.35	20.40	76.41
V17	3.25	38.19	121.53
V19a	1.90	19.56	104.50

tabela 4.17



Deslocamentos nos Pilares Junto à Fundação ( cm )			
Pilar	Fund. Ríg.	F. Flex E=100000	F. Flex E=10000
P1	0.00	0.31	3.06
P3	0.00	0.55	5.10
P5	0.00	0.20	2.65
P6	0.00	0.23	3.02
P7	0.00	0.31	3.85
P11	0.00	0.71	6.23
P13	0.00	0.29	2.91
P14	0.00	0.49	4.05
P15	0.00	0.75	5.56
P19	0.00	0.23	2.85
P20	0.00	0.30	3.40
P21	0.00	0.32	3.51
P22	0.00	0.42	3.87
P27	0.00	0.26	3.56

tabela 4.18

A observação dos resultados apresentados nas tabelas anteriores levam à conclusão que, mesmo tratando-se de ações verticais, a influência da flexibilidade da fundação é bastante sensível. Não representa esse fato um resultado esperado, e muito menos desejável. Para as ações horizontais a consideração dessa flexibilidade exige apenas alguns recursos adicionais de programação, mas não uma alteração completa dos procedimentos de trabalho, sendo que o mesmo não se dá para as ações verticais. Neste caso, a consideração dessa flexibilidade exigiria uma modificação bastante profunda dos métodos tradicionais de análise, pois o cálculo dos pavimentos de forma independente é um processo de análise utilizado de maneira absolutamente generalizada.

Para as forças normais dos pilares, apresentadas na tabela 4.15, as modificações mais significativas nos valores ocorrem para os pilares P27, com variação de 49%, e pilar P15, com -31%. Entretanto esses valores dizem respeito à variação da fundação rígida à calculada com  $E = 10\ 000\ \text{KN/m}^2$ . Caso considere-se apenas a variação entre a rígida e a com  $E = 100\ 000\ \text{KN/m}^2$  esses valores serão de, respectivamente, 8% e -7%. Portanto, para os valores de módulos de elasticidade usuais, a diferença das normais obtidas em relação à fundação rígida pode variar de 20% a 30%, o que não é de todo desprezível.

Quanto aos esforços atuantes nas vigas do pavimento térreo, pode-se perceber que os valores obtidos com a fundação rígida, cortantes de 2 KN e momentos de 3 KN.m, são muito pequenos se comparados com aqueles usuais para uma estrutura desse tipo. Entretanto, quando considera-se a flexibilidade, mesmo com o valor do módulo mais alto, os esforços passam a ser significativos. A força cortante na viga V16a resulta 18.4 KN e momento fletor na viga V17 38.2 KN x m. Com o módulo de  $10\ 000\ \text{KN/m}^2$ , esses valores crescem ainda mais, chegando respectivamente aos valores 58.6 KN e 121.5 KN x m.

Para finalizar, comentam-se os resultados mostrados na tabela 4.18, que são os deslocamentos verticais das bases dos pilares. Para o módulo de elasticidade  $100\ 000\ \text{KN/m}^2$ , eles são relativamente pequenos e todos abaixo de 1 cm. Entretanto, quando considera-se o menor valor do módulo de elasticidade, seus valores crescem muito, variando de 2.65 cm para o P5 até 6.23 cm para o P11. Essas diferenças é que levam os esforços nas vigas a apresentarem os valores que foram constatados.

#### 4.5. CONCLUSÕES

O objetivo deste item é estabelecer algumas conclusões sobre o desenvolvimento e a utilização do elemento sapata rígida.

O primeiro ponto a ser analisado é a facilidade de utilização. Pensa-se que a respeito desse particular os objetivos do trabalho foram perfeitamente alcançados. A utilização do elemento é simples e rápida, encaixando-se com perfeição dentro do esquema inicialmente traçado.

Quanto aos resultados obtidos, eles parecem ser satisfatórios. Entretanto, deve-se reconhecer que não é simples estabelecer comparações. A utilização de outros métodos numéricos fica muito restrita, limitando-se quase que exclusivamente aos aspectos qualitativos. Por mais que se tentasse, não foi possível modelar-se com fidelidade os domínios semi-infinitos através de métodos alternativos, como por exemplo os elementos finitos.

Nem mesmo a comparação com soluções analíticas foi possível de maneira direta. Essas tratam geralmente da aplicação de carregamento uniforme ou linear sobre os domínios considerados [4.23],[4.24]. Como no presente caso as sapatas são rígidas, acontece que o carregamento resulta numa forma diferente da linear, tornando difícil a comparação. Entretanto, é possível fazer algumas considerações a esse respeito.

Tomando-se uma área retangular sobre um domínio semi-infinito, onde aplica-se uma carga uniforme, pode-se encontrar, através das referências citadas, os valores dos deslocamentos nos cantos e no centróide da área. Após esse procedimento, define-se uma sapata, de mesmas dimensões da área considerada, e submetida a um carregamento igual à resultante da carga distribuída lá aplicada. Verifica-se que o deslocamento obtido para a sapata é apenas um pouco superior ao encontrado para os cantos da área, sendo bem inferior ao obtido para o centróide. Assim, pode-se

imaginar que a rigidez obtida para a sapata esteja sendo ligeiramente superestimada.

Esse efeito de certo modo era esperado, pela aproximação adotada para o elemento. É relativamente simples explicá-lo para o próprio caso reportado no parágrafo anterior. A figura 3.31 mostra, através da linha cheia, como devem ser as forças de superfície na sapata, para uma vista lateral. Pelo processo adotado para a montagem da matriz de rigidez, considerando-se pontos nodais apenas nos cantos da sapata, as forças integradas podem ser representadas pela linha tracejada, devendo-se ter um aumento da rigidez estimada em relação ao seu valor real.

Para eliminar-se essa deficiência, seria recomendável a adoção de um maior número de nós para a discretização da sapata. Se, por exemplo, fossem adotados oito nós, conforme a figura 3.32, a integração representaria com maior fidelidade as forças de superfície resultantes. Desse modo, a rigidez estimada deveria ser mais próxima da real. Entretanto, isso ocasionaria um tempo de processamento sensivelmente superior aos obtidos para a aproximação aqui desenvolvida, sendo necessário um estudo cuidadoso que pese com muito critério os benefícios e custos envolvidos. Essa análise é um dos pontos interessantes para a continuação deste trabalho.

Outra conclusão geral de importância diz respeito aos efeitos da consideração da flexibilidade da fundação nos esforços obtidos para as superestruturas de edifícios. Através dos exemplos desenvolvidos, ficou relativamente claro que esses efeitos são realmente importantes, principalmente para edifícios com pilares de grandes dimensões e submetidos a cargas horizontais. De qualquer modo, os exemplos analisados até agora indicam certas tendências gerais que podem estar presentes com maior ou menor intensidade, dependendo do caso considerado.

Entre essas tendências, pode-se destacar, para os

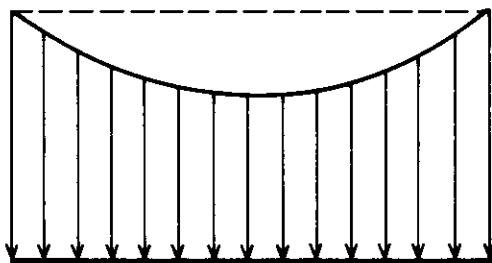


fig 4.32 Vista lateral das forças de superfície numa sapata

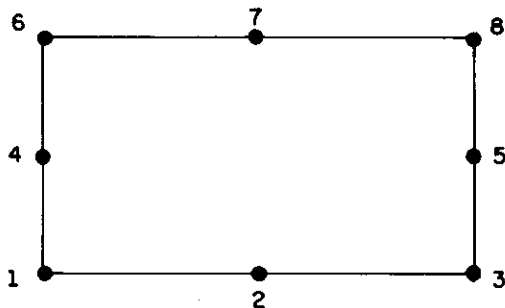


fig 4.33 Discretização da sapata por 8 pontos nodais

casos de solicitação horizontal, a homogeneização dos esforços nos pilares, o aumento dos esforços das vigas e o fato da estrutura ter sua característica de pórtico acentuada, reduzindo-se a contribuição das paredes. Para finalizar, deve-se destacar que mesmo para o carregamento vertical, essa interação também apresenta níveis significativos, constituindo-se essa conclusão, de certo modo, numa surpresa.

São essas as principais conclusões que se pode tirar do assunto desenvolvido, devendo-se, entretanto, ressaltar que nem todas as simulações realizadas e exemplos resolvidos foram apresentados. Desse conjunto, para que a explanação não se tornasse excessivamente longa, foram escolhidos apenas os casos mais representativos.

#### 4.6 REFERÊNCIAS BIBLIOGRÁFICAS

[4.1] Borisenko, A.I.; Tarapov, I.E.; "Vector and Tensors Analysis"; Dover Publications, Inc; New York; 1968.

[4.2] Nakaguma, R.K.; "Three Dimensional Elastostatics Using the Boundary Element Method"; Ph.D. Thesis, University of Southampton; 1979.

[4.3] Venturini, W.S.; "Um Estudo Sobre o Método dos Elementos de Contorno e Suas Aplicações em Problemas de Engenharia"; Tese de Livre-Docência; EESC-USP; 1988.

[4.4] Cruse, T.A.; "Numerical Solutions in Three Dimensional Elastostatics"; Int. J. Solids Structures, vol 5; 1969.

[4.5] Love, A.E.H.; "Treatise on Mathematical Theory of Plasticity"; Dover; 1944.

[4.6] Mindlin, R.D.; "Force at a Point in the Interior of a Semi-Infinite Solid"; Journal of Physics, vol 7; 1936.

[4.7] Boussinesq, J.; "Applications des Potenciels à L'étude de L'équilibre et du Moviment des Solides Elastique"; Gualtier-Villars; Paris; 1885.

[4.8] Brebbia, C.A.; "Weighted Residual Classification of Aproximate Methods"; Appl. Mathematic Modelling, vol 2; 1978.

[4.9] Somigliana, C.; "Sopra l'Equilibrio di un Corpo Elastico Isotropo"; Il Nuovo Cimento; 1886.

[4.10] Lachat, J.C.; "A Further Development of the Boundary Integral Technique for Elastostatics"; Ph. D. Thesis; University of Southampton; 1975.

[4.11] Brebbia,C.A.; Dominguez,J.; "Boundary Elements - An Introductory Course"; Computational Mechanics Publications, 1988.

[4.12] Hartmann,F.; "Computing the C-Matrix on Non Smooth Boundary Points"; In : Brebbia, C.A.,ed. New Developments in Boundary Element Methods CML Publ.; 1980.

[4.13] Zienkiewicz,O.C.; "El Método de los Elementos Finitos"; Ed. Reverté; Barcelona; 1980.

[4.14] Stroud,A.H.; Secrest,D.; "Gaussian Quadrature Formulas", Prentice-Hall, New York, 1966.

[4.15] Ergatoudis,J.G.; "Isoparametric Finite Elements in Two or Three Dimensional Stress Analysis"; Ph. D. Thesis; University of Swansea; 1968.

[4.16] Meek,J.L.; "Matrix Structural Analysis"; McGraw-Hill Kogakusha, Ltd; Tokio; 1971.

[4.17] De Figueiredo,T.G.B.; Brebbia,C.A.; "A Hybrid Displacement Variational Formulation Of B.E.M."; In : Boundary Elements X, Ed. C.A. Brebbia, vol 1; Computational Mechanics Publication, 1988.

[4.18] Dumont, N.A.; "The Hybrid Boundary Element Methods in Elastostatics : Overview of the Theory and Examples", In : Boundary Elements X, Ed. C.A. Brebbia, vol 1; Computational Mechanics Publication, 1988.

[4.19] Maier,G.; Novati,G.; Perego,V.; "Plastic Analysis by Boundary Elements"; In : Finite Elements and Boundary Element Techniques from Mathematical and Engineering Point of View, Ed. E.Stein and W.L. Wendland; International Centre for Mechanical Sciences; 1988.



[4.20] Meek, J.L.; Bandyopadhyay, B.; "BECOUP - A Program for Coupled Boundary and Finite Element Analysis in 3-D Elasto-Statics"; In : Boundary Element X, Ed. C.A. Brebbia, vol 3; Computational Mechanics Publications; 1988.

[4.21] Brebbia, C.A.; Georgiou, P.; "Combination of Bondary and Finite Elements in Elastostatics"; App. Math. Modelling, vol 3; 1979.

[4.22] Vilar, O.M.; Bueno, B.S.; "Mecânica dos Solos - Vol II"; Seção de Publicações da EESC-USP; São Carlos; 1985.

[4.23] Poulos, H.G.; Davis, E.H.; "Elastic Solutions for Soil and Rock Mechanics"; John Wiley & Sons, Inc; 1974.

[4.24] De Sá, P.A.C.O.; Telles, J.C.F.; "Análise de Problemas de Elasticidade Linear Tridimensional Pelo Método dos Elementos de Contorno Utilizando as Soluções Fundamentais de Kelvi e Mindlin"; VII Congresso Latino-Americano de Métodos Computacionais Para Engenharia; São Carlos, 1986.