

How  
621.8  
P436e  
TES/A

212.0

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

ELABORAÇÃO DE ROTINAS GRÁFICAS PARA DAR SUPORTE  
A UM PROGRAMA COMPUTACIONAL DE ANÁLISE ESTÁTICA  
E DINÂMICA DE MÁQUINAS ROTATIVAS

Dissertação apresentada à Universidade Federal de  
Uberlândia por José Carlos Pereira para obtenção do título  
de Mestre em Engenharia Mecânica aprovada em 09/04/1990  
pela seguinte banca examinadora **881800**

Valder Steffen Jr. (Orientador) <sup>1992</sup>

Francisco Paulo Lépore Neto (UFU)

Carlos Roberto Ribeiro (UFU)

Mário Francisco Mucheroni (EESC - USP)

DIRBI/UFU 621.81 P436e /TES/FU  
02182/90



1000017153

## AGRADECIMENTOS

Agradeço a todas as pessoas que, de alguma forma contribuíram na realização deste trabalho.

À minha família,  
e a todos meus amigos.

ELABORAÇÃO DE ROTINAS GRÁFICAS PARA DAR SUPORTE  
A UM PROGRAMA COMPUTACIONAL DE ANÁLISE ESTÁTICA  
E DINÂMICA DE MÁQUINAS ROTATIVAS

ÍNDICE

. LISTA DE FIGURAS	vii
. LISTA DE SÍMBOLOS	ix
1. INTRODUÇÃO	1
2. MÉTODO DOS ELEMENTOS FINITOS EM DINÂMICA DE ROTAÇÃO	6
2.1. Energia cinética	7
2.1.1. Elemento de disco	7
2.1.2. Elemento de barra	9
2.2. Energia de deformação	11
2.3. Força peso	14
2.4. Forças nos mancais	16
2.5. Equações do rotor	17
3. GRAPHICAL KERNEL SYSTEM (GKS)	19
3.1. Introdução ao GKS	19
3.2. Primitivas gráficas de saída	20
3.2.1. Polyline	21
3.2.2. Polymarker	22
3.2.3. Text	23
3.2.4. Fill Area	25
3.2.5. GDP	27
3.3. Sistemas de coordenadas	29
3.4. Segmentação	31
3.4.1. Construção de segmentos	33

3.4.2. Atributos de segmentos	33
3.5. Dispositivos lógicos de entrada	35
3.5.1. Classe de dispositivos lógicos de entrada	37
3.5.2. Modo de operação dos DLE's	36
3.6. Estações de trabalho	39
4. ESTRUTURA COMPUTACIONAL DOS PROGRAMAS GRÁFICOS	42
4.1. Programa Rotor	42
4.1.1. Estrutura do arquivo de dados de entrada	42
4.1.2. Estrutura do programa Rotor	45
4.1.3. Estrutura do arquivo de dados de saída	46
4.2. Construção das rotinas gráficas para o programa Rotor	47
4.2.1. Estrutura do programa de pré-processamento do programa Rotor	48
4.2.2. Estrutura do programa de pós-processamento do programa Rotor	60
5. APLICAÇÕES	79
6. CONCLUSÃO	88
7. REFERÊNCIAS BIBLIOGRÁFICAS	90
8. ANEXOS	92
8.1. Anexo A - Matrizes massa e Coriolis de um disco	92
8.2. Anexo B - Matrizes massa e Coriolis de um elemento de barra	92
8.3. Anexo C - Matrizes rigidez elástica (com efeito de cisalhamento) e rigidez geométrica de um elemento de barra	95
8.4. Anexo D - Orientação para instalação e uso do programa de pré-processamento do Programa Rotor, do Programa Rotor e do programa de pós-processamento do Programa Rotor	98

PEREIRA.J.C, Elaboração de rotinas gráficas para dar suporte a um programa computacional de análise estática e dinâmica de máquinas rotativas, Uberlândia, 1990,

## RESUMO

Este trabalho tem por objetivo o desenvolvimento de rotinas computacionais gráficas para dar suporte a um programa de análise estática e dinâmica de rotores que permitam, antes do processamento dos dados, que se tenha uma visualização da configuração da máquina rotativa em estudo, e também dos resultados obtidos após o processamento. Quanto à configuração do rotor, o que se pretende, é se dispor no terminal de vídeo, de uma representação do modelo elementos finitos da máquina rotativa em estudo, a malha adotada, numeração dos nós, elementos utilizados, geometria e materiais constitutivos, além das condições de contorno (mancais). Quanto aos resultados, são obtidos gráficos das frequências naturais para diferentes velocidades de rotação (diagrama de Campbell), resposta do sistema a um desbalanceamento ou a uma excitação assíncrona e representação dos modos próprios com animação. Para a elaboração das rotinas gráficas, escolheu-se um "software" gráfico padronizado internacionalmente, qual seja, o GKS (Graphical Kernel System).

PEREIRA, J.C., Graphical routines to support a computer software for rotating machinery analysis, Uberlândia, 1990,

#### ABSTRACT

The objective of this research work is to develop computer graphic routines to support an existing computer program for the statical and dynamical analysis of rotors. These graphics routines permit, to obtain a representation on the display of the rotating machine under study before processing the finite element data and to present the results graphically after calculations. This way it is possible to have a representation of the finite element model of the rotating machine, the position of the nodes and their numbers, the types of elements used, their geometry and materials proprieties and the boundary conditions (journals). For the results, plots of natural frequencies for different speeds of rotation ( Campbell diagram ), response due to unbalance or to non-synchronous excitation representation of the mode shapes are shown. The graphical routines are based on an international graphical software which is the Graphical Kernel System - GKS.

## LISTA DE FIGURAS

Figura	Significado	Página
2.1.1	Sistema de eixos de referência	7
2.1.2	Elemento de barra finito	9
2.2	Sistema de eixos de referência do rotor	12
2.3	Forças devido a aceleração gravitacional	14
2.4	Esquema de um mancal	16
3.2.1	Representação de uma "Polyline"	22
3.2.2	Representação de uma "Polymarker"	23
3.2.3	Representação de uma "Text"	25
3.2.4.	Representação de uma "Fill Área"	27
3.2.5	Representação de uma "GDP"	28
3.3	Transformação de coordenadas	31
5.1.1	Modelo elementos finitos para o exemplo de aplicação didática	79
5.1.2	Características de um elemento de barra	81
5.1.3	Características de um disco rígido	82
5.1.4	Características de um mancal	82
5.1.5	Diagrama de Campbell	83
5.1.6	Resposta do sistema a um desbalanceamento	84
5.1.7	Representação de uma imagem referente a animação do 1 modo a 2000 rpm	84
5.2.1	Modelo elementos finitos para o exemplo de aplicação industrial	85
5.2.2	Diagrama de Campbell	86
5.2.3	Resposta do sistema a um desbalanceamento colocado sobre o disco no nó 2	86



5.2.4

Representação de uma imagem referente a  
animação do 1 modo a 3000 rpm

87

## LISTA DE SÍMBOLOS

Símbolo	Significado	Unidade
$A_{xx}$	amortecimento segundo o eixo X	N/(m/s)
$A_{zz}$	amortecimento segundo o eixo Z	N/(m/s)
$A_{xz}, A_{zx}$	termos cruzados de amortecimento	N/(m/s)
C	matriz giroscópica	N/(m/s <sup>2</sup> )
E	módulo de elasticidade	N/m <sup>2</sup>
$F_{u1}, F_{w1}$	forças de ligação na árvore em X e Z respectivamente	N
$F_{u2}, F_{w2}$	forças de ligação na fundação em X e Z respectivamente	N
$F_1$	força peso	N
$F_2, F_3$	forças devido ao desbalanceamento	N
$F_4, F_5$	forças devido ao efeito assíncrono	N
G	módulo de Coulomb	N/m <sup>2</sup>
$I_{dx}, I_{dy}, I_{dz}$	inércias de massa do disco	Kg.m <sup>2</sup>
I	inércia média da seção transversal em relação a um diâmetro	m <sup>4</sup>
$I_1, I_2$	inércia da seção transversal nos nós 1 e 2 do elemento	m <sup>4</sup>
$I_p$	inércia polar de seção transversal	m <sup>4</sup>
$K_{xx}$	rigidez segundo o eixo X	N/m
$K_{zz}$	rigidez segundo o eixo Z	N/m
$K_{xz}, K_{zx}$	termos cruzados de rigidez	N/m
K	matriz rigidez global do sistema	N/m
L	comprimento de elemento de barra	m
M	matriz massa global do sistema	Kg

$M_d$	massa do disco	Kg
$S$	área média da seção transversal do elemento de barra	$m^2$
$S_1, S_2$	área da seção transversal nos nós 1 e 2 do elemento	$m^2$
$a$	coeficiente	*
$g$	gravidade	$m/s^2$
$u, w$	coordenadas em X e Z referentes ao centro de massa de um elemento de barra ou disco	m
$\Omega$	velocidade angular do rotor	rd/s
$\Omega_x, \Omega_y, \Omega_z$	componentes do vetor instantâneo de rotação $\vec{\Omega}$	rd/s
$\beta$	coeficiente próximo de 1 que permite a obtenção da seção equivalente de cisalhamento	*
$\epsilon$	deformação longitudinal total	*
$\epsilon_l$	deformação longitudinal de um ponto x,y da seção	*
$\epsilon_{nl}$	parte não linear da deformação longitudinal	*
$\phi$	ângulo em torno de y	rd
$\psi$	ângulo em torno de Z	rd
$\theta$	ângulo em torno de x1	rd
$\rho$	massa volumétrica do elemento de barra	$Kg/m^3$
$\tau$	tensão normal na direção y	$N/m^2$

---

\* termos adimensionais

ELABORAÇÃO DE ROTINAS GRÁFICAS PARA DAR SUPORTE  
A UM PROGRAMA COMPUTACIONAL DE ANÁLISE ESTÁTICA  
E DINÂMICA DE MÁQUINAS ROTATIVAS

1. INTRODUÇÃO

A solução de problemas de dinâmica de rotação depende da formulação de um modelo matemático-computacional adequado e de um elenco de rotinas computacionais capazes de resolver de forma eficiente as várias operações matriciais necessárias para a análise estática e dinâmica do sistema.

Os dois principais métodos de análise de sistemas rotativos empregados atualmente são o método das matrizes de transferência e o método dos elementos finitos. Ambos adotam uma representação contínua dos parâmetros dos elementos, constituindo a principal razão para sua relativamente alta precisão, comparada com a modelagem convencional por parâmetros concentrados [1].

O método das matrizes de transferência permite a obtenção de resultados teóricos que apresentam de boa concordância com testes experimentais. Este método tem sido usado em análise modal para cálculo das frequências naturais amortecidas e das funções próprias, e tem a vantagem de requerer computadores com pequena capacidade de memória, embora seja de aplicação não adequada para rotores complexos.

Em muitos trabalhos sobre dinâmica de rotores utilizou-se este método analítico, tal como em [2], onde procurou-se modelar matematicamente um rotor considerado como sistema contínuo, de forma a obter-se a formulação geral com respeito

aos auto-valores e funções próprias. Daí então, para uma dada configuração física, são calculadas as diversas frequências próprias, velocidades críticas e funções próprias.

Por outro lado, o método dos elementos finitos é uma técnica de análise numérica para obtenção aproximada de soluções para uma variedade de problemas em engenharia, que fornece grande precisão na modelagem de sistemas rotor-mancais, mesmo para pequeno número de elementos, sendo também menos sensível a instabilidades numéricas do que o método precedente [3].

Mesmo para problemas complexos, pode-se escrever as equações que os governam e as condições de contorno, assim como obter-se soluções de boa precisão ao comparar-se com resultados experimentais [3]. Com a disponibilidade, velocidade de processamento e capacidade de memória dos computadores digitais, pode-se conservar as complexidades do problema e ainda obter solução numérica aproximada de boa qualidade.

O método dos elementos finitos leva essencialmente a um conjunto de equações dinâmicas independentes de qualquer pré-especificação da excitação de entrada. Detalhes adicionais, tais como mancais flexíveis por exemplo, podem ser incorporados facilmente na formulação por elementos finitos. Similarmente, deformações cisalhantes e inércia de rotação das seções transversais tem também sido incluídas, em geral, na modelagem por elementos finitos, e usadas em problemas de dinâmica de rotores, já que podem ter um efeito considerável, particularmente nos modos superiores. Deformações devidas ao cisalhamento podem ser consideradas utilizando-se de elementos do tipo viga simples de Timoshenko [3], [4] e [5].

Alguns estudos mais antigos na área de dinâmica de rotores incluem somente a energia elástica e a energia cinética translacional, enquanto que os efeitos de inércia rotacional, momentos giroscópicos, força axial, torque axial, deformações cisalhantes são desprezados. Nos modelos mais recentemente adotados, incluem-se tais efeitos como em [6], onde considera-se o efeito de inércia rotacional, momentos giroscópicos, força axial e mancais com rigidez linear e amortecimento viscoso.

Na referência [7] foi utilizado o método dos elementos finitos para analisar sistemas rotor-mancais de grande porte, baseando-se em uma representação da massa e da elasticidade como parâmetros distribuídos, enquanto que os mancais são do tipo filme fluido. O método modal foi usado para reduzir o número de graus de liberdade do sistema, permitindo assim seu tratamento computacional. O método foi aplicado para a obtenção da resposta em regime permanente devido a um desbalanceamento e para a análise de estalidade.

Quando o rotor está em rotação, suas frequências próprias não são as mesmas do caso não-giroscópico, dependendo pois da velocidade de rotação. A rotação produz sobre o rotor forças de excitação adicionais que podem se tornar muito perigosas quando a velocidade de rotação coincide com a frequência natural do rotor (velocidade crítica). Dentro deste estudo, os métodos de cálculo permitem determinar as frequências e modos dos rotores em função da velocidade de rotação e a resposta a excitações síncronas ou assíncronas.

No programa Rotor [8], estabeleceu-se as equações do movimento do rotor utilizando-se das equações de Lagrange e de

uma formulação pelo método dos elementos finitos, onde os elementos constitutivos do rotor são discos rígidos, elementos de barra de Timoshenko e mancais.

O programa Rotor, desenvolvido no Laboratório de Mecânica de Estruturas do INSA de Lyon (França), escrito em linguagem Fortran, dispõe de uma versão compatível para microcomputadores do tipo IBM-PC, e tem grande aceitação comercial.

O programa em questão, testado para as mais diferentes máquinas rotativas, utiliza arquivo de dados de entrada e de saída de forma convencional, o que dificulta o trabalho de conferência dos dados do modelo elementos finitos antes do processamento e, a análise dos resultados pelo fato do programa não fornecer uma saída gráfica dos resultados, como por exemplo, os valores de frequência natural em função da velocidade de rotação (Diagrama de Campbell) com seus modos correspondentes e a resposta à um desbalanceamento ou à uma excitação assíncrona.

Já que ao programa de análise máquinas rotativas foi acoplado um programa de otimização [9], [10], com o objetivo de se proceder a otimização dinâmica do sistema partindo de uma configuração inicial proposta, a visualização do processo de otimização é muito interessante e, simplifica a tomada de decisão pelo fato de ser possível acompanhar a evolução do processo em busca do ótimo ao longo das várias iterações.

Neste sentido, o objetivo deste trabalho é dar suporte ao programa Rotor, desenvolvendo rotinas gráficas de modo a reproduzir o modelo elementos finitos em estudo, destacando os elementos que o compõem, com opções para que possa ser feita a conferência dos valores no arquivo de entrada, e após o

processamento, gerar os gráficos dos resultados, inclusive com animação dos modos próprios.

Devido ao fato de o programa Rotor ser comercializado, tem-se a necessidade de um "software" gráfico que seja independente de periféricos e que não comprometa a portabilidade dos programas quando da migração para outras instalações, e que possa interfacear as rotinas gráficas com o programa aplicação. Tais exigências levaram à escolha do "software" GKS (GRAPHICAL KERNEL SYSTEM), padrão gráfico internacional, que dispõe de um conjunto de procedimentos e funções que permitem o acesso e o controle dos dispositivos gráficos de entrada e saída. Tais funções dão mais versatilidade ao GKS, permitindo uma melhor comunicação do operador com o sistema gráfico através da seleção dos dispositivos de entrada em função da natureza da informação a ser tratada, e de uma representação visual das imagens sintetizadas não necessariamente ao final do processo de criação, podendo assumir instâncias intermediárias necessárias ao desenrolar do próprio processo.



## 2. MÉTODOS DOS ELEMENTOS FINITOS EM DINÂMICA DE ROTAÇÃO

Um típico sistema rotor-mancais consiste de um eixo suportado por dois ou mais mancais em que são montados um certo número de discos ou inércias. O eixo raramente é uniforme, mas para a análise proposta pode ser facilmente representado por uma série de pontos nodais conectados por segmentos de eixos de seção uniforme. Para facilitar a modelização, pontos nodais são geralmente introduzidos em cada localização onde o diâmetro do eixo é alterado, nos pontos onde há massa concentrada, em cada localização dos mancais, e em ambas as extremidades.

Assim qualquer sistema rotor-mancais pode ser representado pela combinação destes três tipos de "elementos finitos":

- elemento de barra com simetria axial e propriedades de elasticidade e inércia distribuídas uniformemente.
- elemento de disco rígido simétrico.
- elemento de mancal, com coeficientes de rigidez linear e de amortecimento viscoso.

Se o movimento do rotor na direção axial é considerado como sendo desprezível e o sistema é torsionalmente rígido, o número total de graus de liberdade de um elemento de barra é reduzido para oito.

Nesta investigação, um sistema rotor-mancais flexíveis é representado pelo modelo elementos finitos, levando em consideração momentos giroscópicos, inércia rotacional, deformação devido ao cisalhamento, e rigidez e amortecimento nos mancais.

As equações de movimento de discos e dos elementos de barra são obtidas através da equação de Lagrange:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} - Q_i = 0 \quad (1)$$

## 2.1. Energia cinética [11]:

### 2.1.1. elemento de disco.

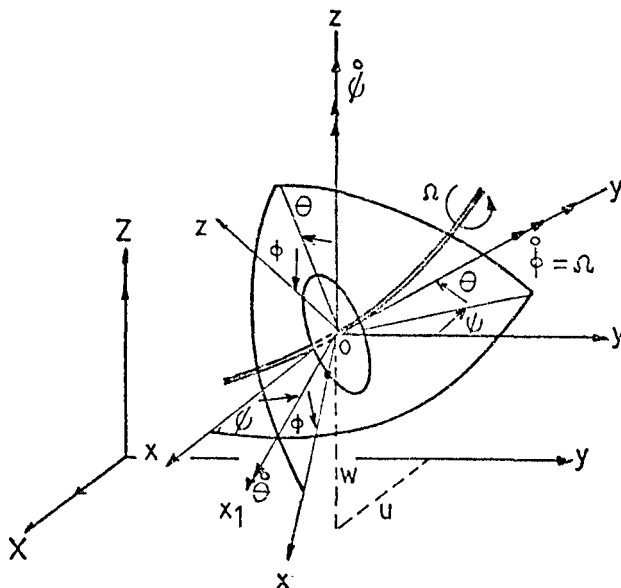


fig. 2.1.1 - Sistema de eixos de referência

O disco está fixo no plano x-z do sistema de eixos rotativos x y z e o vetor velocidade angular instantânea de x y z em relação a X Y Z, expresso por:

$$\vec{\Omega} = \dot{\psi} \vec{Z} + \dot{\theta} \vec{x}_1 + \dot{\phi} \vec{y} \quad (2)$$

No sistema fixo ao disco, tem-se:

$$\begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = \begin{bmatrix} -\dot{\psi} \cos \theta \sin \phi + \dot{\theta} \cos \phi \\ \dot{\phi} + \dot{\psi} \sin \theta \\ \dot{\psi} \cos \theta \cos \phi + \dot{\theta} \sin \phi \end{bmatrix} \quad (3)$$

A energia cinética do disco é então:

$$T_d = \frac{1}{2} M_d \left( \dot{u}^2 + \dot{w}^2 \right) + \frac{1}{2} \left( I_{dx} \Omega_x^2 + I_{dy} \Omega_y^2 + I_{dz} \Omega_z^2 \right) \quad (4)$$

Levando-se em consideração que  $\theta$  e  $\psi$  são pequenos e  $\phi = \Omega t$ , tem-se:

$$\cos \theta = 1, \cos \psi = 1, \sin \theta = \theta, \sin \psi = \psi, \sin^2 \theta = 0, \sin^2 \psi = 0$$

Então a equação (4) se torna:

$$T_d = \frac{1}{2} M_d \left( \dot{u}^2 + \dot{w}^2 \right) + \frac{1}{2} I_{dx} \left( \dot{\theta}^2 + \dot{\psi}^2 \right) + I_{dy} \Omega \dot{\psi} \theta + \frac{1}{2} I_{dy} \Omega^2 \quad (5)$$

A partir da aplicação da equação (5) na equação (1), obtém-se as matrizes de inércia e giroscópica do disco. A matriz de inércia é obtida através dos dois primeiros termos, a matriz anti-simétrica de Coriolis (giroscópica) é obtida através do terceiro. Estas matrizes são apresentadas no anexo A.

## 2.1.2. elemento de barra.

Aplicando a equação (5) em um elemento de barra de seção transversal constante e comprimento L tem-se:

$$\begin{aligned}
 T_{eb} = & \frac{1}{2} \rho S \int_0^L \left( \dot{u}^2 + \dot{w}^2 \right) dy + \\
 & \frac{1}{2} \rho I \int_0^L \left( \dot{\psi}^2 + \dot{\theta}^2 \right) dy + \rho I_p \Omega \int_0^L \dot{\psi} \dot{\theta} dy + \\
 & \frac{1}{2} \rho \Omega^2 I_p L
 \end{aligned} \tag{6}$$

Seja o vetor deslocamento nodal da seguinte forma (de acordo com a fig. 2.1.2):

$$\begin{aligned}
 \delta u = & | u_1, \psi_1, u_2, \psi_2 |^t \\
 \delta w = & | w_1, \theta_1, w_2, \theta_2 |^t
 \end{aligned} \tag{7}$$

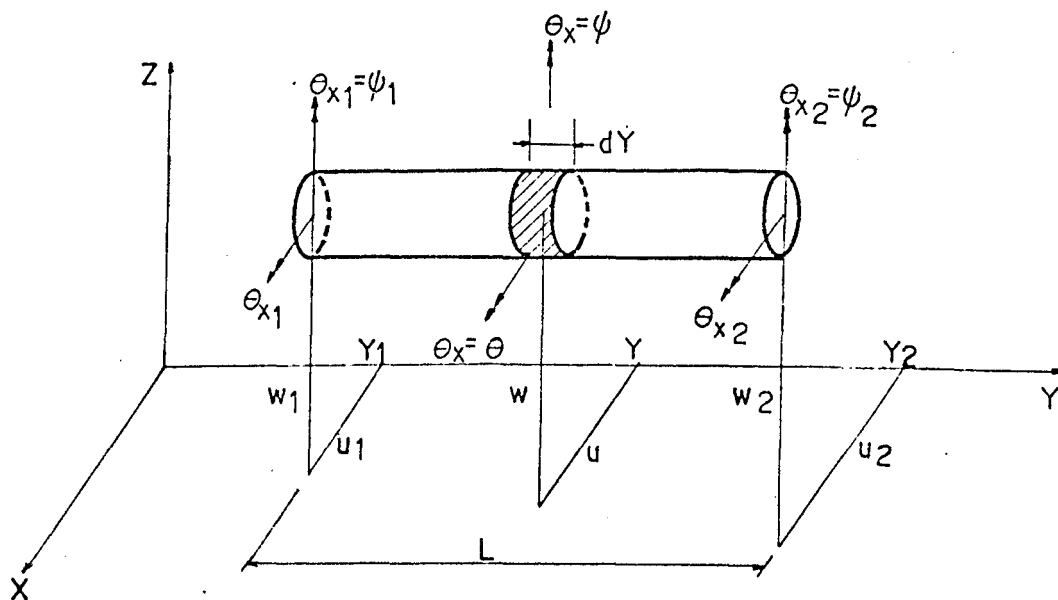


fig. 2.1.2 - elemento finito de barra

As funções deslocamento são:

$$u = N_1(y) \delta u \quad ; \quad w = N_2(y) \delta w \quad (8)$$

onde:

$$N_1(y) = \left[ 1 - \frac{3y^2}{L^2} + \frac{2y^3}{L^3} ; -y + \frac{2y^2}{L} - \frac{y^3}{L^2} ; \frac{3y^2}{L^2} - \frac{2y^3}{L^3} ; \right. \\ \left. - \frac{y^2}{L} + \frac{y^3}{L^2} \right]$$

$$N_2(y) = \left[ 1 - \frac{3y^2}{L^2} + \frac{2y^3}{L^3} ; y - \frac{2y^2}{L} + \frac{y^3}{L^2} ; \frac{3y^2}{L^2} - \frac{2y^3}{L^3} ; \right. \\ \left. - \frac{y^2}{L} + \frac{y^3}{L^2} \right] \quad (9)$$

Substituindo-se as equações (7), (8) e (9) na equação (6),

tem-se:

$$T_{eb} = \frac{1}{2} \rho S \int_0^L \left( \delta \dot{u}^t N_1^t N_1 \delta \dot{u} + \delta \dot{w}^t N_2^t N_2 \delta \dot{w} \right) dy + \\ \frac{1}{2} \rho I \int_0^L \left( \delta \dot{u}^t \frac{dN_1^t}{dy} \frac{dN_1}{dy} \delta \dot{u} + \delta \dot{w}^t \frac{dN_2^t}{dy} \frac{dN_2}{dy} \delta \dot{w} \right) dy + \\ \rho I_p \Omega \int_0^L \delta \dot{u}^t \frac{dN_1^t}{dy} \frac{dN_2}{dy} \delta \dot{w} dy + \frac{1}{2} \rho \Omega^2 I_p L \quad (10)$$

Aplicando a equação (1) na equação (10), a matriz de massa é obtida através dos dois primeiros termos; o terceiro termo fornece a matriz de Coriolis, e o quarto, por ser constante, não tem influência. Estas matrizes estão no anexo B.

## 2.2. Energia de deformação [11]:

A expressão geral para a energia de deformação é:

$$U = \frac{1}{2} \int_{\tau} \epsilon^t \sigma \, d\tau \quad (11)$$

onde:

$$\sigma = E \epsilon \quad (12)$$

Seja  $u^*$  e  $w^*$  componentes do deslocamento de um ponto qualquer da seção transversal no eixo de referência, atribuído a uma seção transversal não deformada de um elemento de barra, (fig. 2.2). Considerando somente a deformação longitudinal, a deformação total pode ser expressa por:

$$\epsilon_l = -x \frac{\partial^2 u^*}{\partial y^2} - z \frac{\partial^2 w^*}{\partial y^2} \quad (13)$$

$$\epsilon_{nl} = \frac{1}{2} \left( \frac{\partial w^*}{\partial y} \right)^2 + \frac{1}{2} \left( \frac{\partial u^*}{\partial y} \right)^2 \quad (14)$$

$$\epsilon = \epsilon_l + \epsilon_{nl} \quad (15)$$

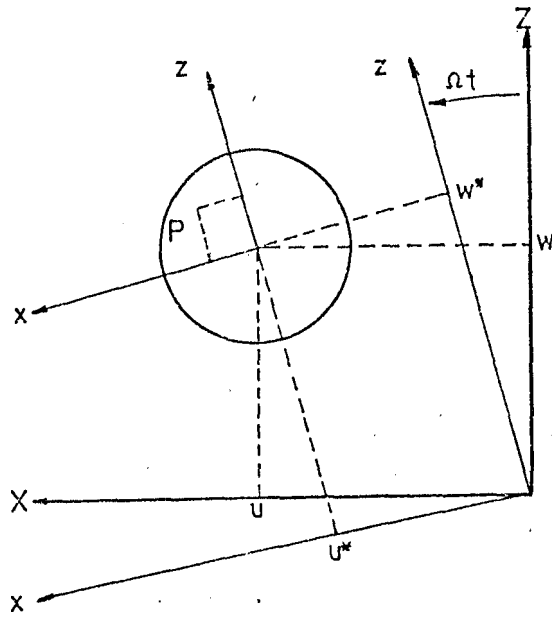


fig. 2.2 - Sistema de eixos de referência do rotor

Desprezando os efeitos não lineares, a equação da energia de deformação é dada por:

$$U_1 = \frac{1}{2} E \int_{\tau} \left[ -z \frac{\partial^2 w^*}{\partial y^2} - x \frac{\partial^2 u^*}{\partial y^2} \right]^2 d\tau \quad (16)$$

Desde que o rotor seja submetido a uma força axial  $F_0$  constante, a energia de deformação correspondente pode somar-se a  $U_1$  da seguinte forma:

$$U_2 = \int_{\tau} \frac{F_0}{S} \left( \epsilon_1 + \epsilon_{n1} \right) d\tau \quad (17)$$

Por causa da simetria, a integral de  $\epsilon_1$  sobre a seção do elemento é nula, logo:

$$U_2 = \frac{F_0}{2} \int_{\tau} \frac{1}{S} \left[ \left( \frac{\partial w^*}{\partial y} \right)^2 + \left( \frac{\partial u^*}{\partial y} \right)^2 \right] d\tau \quad (18)$$

Os deslocamentos  $u^*$  e  $w^*$  são tais que:

$$u^* = -w \operatorname{sen} \Omega t + u \operatorname{cos} \Omega t \quad (19)$$

$$w^* = w \operatorname{cos} \Omega t + u \operatorname{sen} \Omega t$$

Expressando  $U_1$  e  $U_2$  em função de  $u$  e  $w$ , tem-se:

$$U = U_1 + U_2$$

$$U = \frac{1}{2} E \int_0^L I \left[ \left( \frac{\partial^2 w}{\partial y^2} \right)^2 + \left( \frac{\partial^2 u}{\partial y^2} \right)^2 \right] dy + \frac{F_0}{2} \int_0^L \left[ \left( \frac{\partial w}{\partial y} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 \right] dy \quad (20)$$

Substituindo as equações (7), (8) e (9) na equação (20), obtém-se:

$$U = \frac{1}{2} E I \int_0^L \left[ \delta u^t \frac{d^2 N_1^t}{dy^2} \frac{d^2 N_1}{dy^2} \delta u + \delta w^t \frac{d^2 N_2^t}{dy^2} \frac{d^2 N_2}{dy^2} \delta w \right] dy + \frac{1}{2} F_0 \int_0^L \left[ \delta u^t \frac{dN_1^t}{dy} \frac{dN_1}{y} \delta u + \delta w^t \frac{dN_2^t}{dy} \frac{dN_2}{dy} \delta w \right] dy \quad (21)$$



Aplicando a equação (21) na equação (1), as matrizes de rigidez elástica e de rigidez geométrica são obtidas através do primeiro e segundo termo respectivamente. Estas matrizes são apresentadas no anexo C.

### 2.3. Força peso [6]:

Considerando as forças devido a uma aceleração  $g$  sobre o plano XZ, cujos cosenos diretores são CX e CZ, temos:

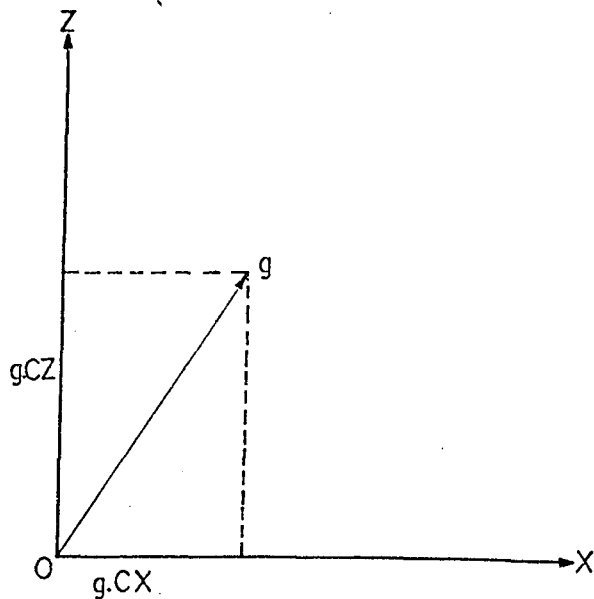


fig. 2.3 - Forças devido a aceleração gravitacional

A força gravitacional na direção Z para um volume elementar é:

$$\delta W = \int_{\tau} f \delta w d\tau \quad (22)$$

onde:

$$f = \rho g CZ \quad (23)$$

Supondo que a seção varie linearmente ao longo do elemento, obtém-se as forças nodais equivalentes na direção Z:

$$\begin{aligned} F_{w1} &= \frac{\rho g L}{20} ( 7 S_1 + 3 S_2 ) \cdot CZ \\ F_{\theta 1} &= \frac{\rho g L^2}{60} ( 3 S_1 + 2 S_2 ) \cdot CZ \\ F_{w2} &= \frac{\rho g L}{20} ( 3 S_1 + 7 S_2 ) \cdot CZ \\ F_{\theta 2} &= - \frac{\rho g L^2}{60} ( 2 S_1 + 3 S_2 ) \cdot CZ \end{aligned} \quad (24)$$

As forças segundo Ox são :

$$\begin{aligned} F_{u1} &= \frac{\rho g L}{20} ( 7 S_1 + 3 S_2 ) \cdot CX \\ F_{\psi 1} &= - \frac{\rho g L^2}{60} ( 3 S_1 + 2 S_2 ) \cdot CX \\ F_{u2} &= \frac{\rho g L}{20} ( 3 S_1 + 7 S_2 ) \cdot CX \\ F_{\psi 2} &= \frac{\rho g L^2}{60} ( 2 S_1 + 3 S_2 ) \cdot CX \end{aligned} \quad (25)$$

## 2.4. Forças nos mancais [8]:

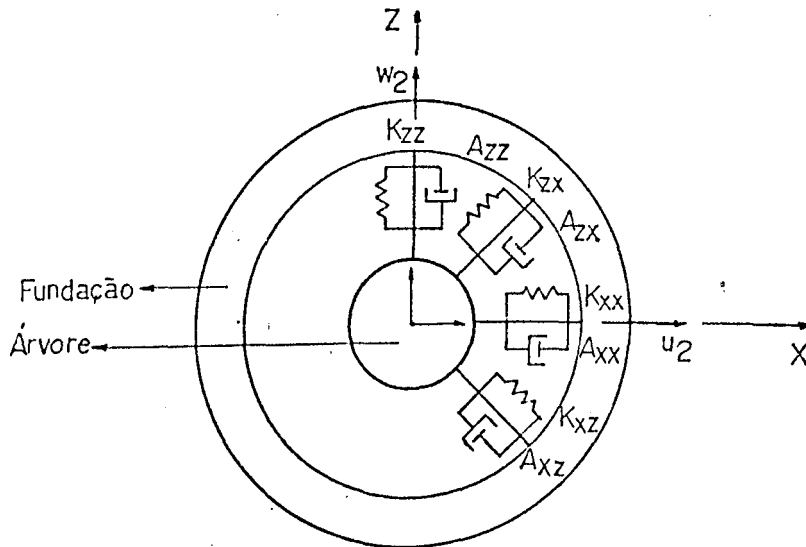


fig. 2.4 - Esquema de um mancal

As forças que surgem nos mancais, para os termos da matriz de rigidez e de amortecimento, supostos constantes, são:

$$\begin{vmatrix} Fu1 \\ Fw1 \\ Fu2 \\ Fw2 \end{vmatrix} = - \begin{vmatrix} K_{xx} & K_{xz} & -K_{xx} & -K_{xz} \\ K_{zx} & K_{zz} & -K_{zx} & -K_{zz} \\ -K_{xx} & -K_{xz} & K_{xx} & K_{xz} \\ -K_{zx} & -K_{zz} & K_{zx} & K_{zz} \end{vmatrix} \begin{vmatrix} u1 \\ w1 \\ u2 \\ w2 \end{vmatrix} - \begin{vmatrix} A_{xx} & A_{xz} & -A_{xx} & -A_{xz} \\ A_{zx} & A_{zz} & -A_{zx} & -A_{zz} \\ -A_{xx} & -A_{xz} & A_{xx} & A_{xz} \\ -A_{zx} & -A_{zz} & A_{zx} & A_{zz} \end{vmatrix} \begin{vmatrix} \dot{u}1 \\ \dot{w}1 \\ \dot{u}2 \\ \dot{w}2 \end{vmatrix} \quad (26)$$

## 2.5. Equações do rotor:

A montagem das matrizes elementares dos elementos constituintes do rotor conduz a uma equação diferencial matricial do tipo:

$$M \ddot{X} + C \dot{X} + K X = F_1 + F_2 \sin \Omega t + F_3 \cos \Omega t + F_4 \sin a \Omega t + F_5 \cos a \Omega t \quad (27)$$

A solução da equação (27) é obtida pelo método modal que se utiliza dos modos do sistema não-giroscópico para reduzir o número de graus de sistema completo. São então calculados os modos do sistema não-giroscópico associado:

$$M \ddot{X} + K X = 0 \quad (28)$$

Os  $n$  primeiros auto-vetores ( $n \ll N$ ) são usados para a mudança de variável da forma:

$$X = \Phi p \quad (29)$$

Usando a equação (29), a equação (27) se torna:

$$\begin{aligned} \Phi^t M \Phi \ddot{p} + \Phi^t C \Phi \dot{p} + \Phi^t K \Phi p = \Phi^t F_1 + \\ \Phi^t \left( F_2 \sin \Omega t + F_3 \cos \Omega t \right) + \\ \Phi^t \left( F_4 \sin a \Omega t + F_5 \cos a \Omega t \right) \end{aligned} \quad (30)$$

As frequências naturais em função da velocidade de rotação são obtidas resolvendo o seguinte sistema de equações homogênea:

$$\Phi^t M \Phi \ddot{p} + \Phi^t C \Phi \dot{p} + \Phi^t K \Phi p = 0 \quad (31)$$

A resposta de regime permanente devido à força  $F_1$ , constante, é obtida pela solução da equação:

$$\Phi^t K \Phi p = \Phi^t F_1 \quad (32)$$

A resposta de regime permanente devido às forças  $F_2$  e  $F_3$  ou  $F_4$  e  $F_5$  é encontrada, procurando-se soluções da forma:

$$p = P_1 \cos \Omega t + P_2 \sin \Omega t \quad (33)$$

ou

$$p = P_1 \cos a \Omega t + P_2 \sin a \Omega t, \quad (34)$$

respectivamente.

### 3. GRAPHICAL KERNEL SYSTEM (GKS) [12],[13],[14]

#### 3.1. Introdução ao GKS

O GKS é um "software" gráfico cuja especificação foi aprovada como padrão internacional pela ISO (International Standard Organization) em 1984 e por diversas outras entidades nacionais de padronização (como a DIN, ANSI, BSI). É o resultado de um trabalho de conjunto em nível internacional, que envolveu mais de 100 especialistas desde 1976.

O pacote busca atender as diversas tarefas inerentes a um sistema gráfico poderoso:

- a) geração e representação de figuras;
- b) canalização das figuras criadas no sistema de coordenadas do programa de aplicação nos diferentes dispositivos de exibição gráfica, mapeando-as na superfície de exibição desses dispositivos;
- c) controle funcional dos dispositivos de entrada e saída conectados ao sistema;
- d) manipulação dos recursos de entrada do sistema;
- e) estruturação de figuras em partes manipuláveis independentes;
- f) armazenagem e transmissão de figuras.

Como todo "software" gráfico, o GKS posiciona-se entre o programa de aplicação e os diversos periféricos gráficos normalmente acessados por intermédio do sistema operacional. No sentido de tornar os programas de aplicação independentes das características específicas e dos códigos de controle dos periféricos de uma dada instalação, a especificação do GKS

introduziu uma série de modelos e conceitos. Através desses modelos, ficam definidos, para o programa de aplicação, periféricos virtuais que se comportam de maneira uniforme. Entretanto, quando o programa de aplicação necessitar de informações específicas sobre os periféricos (tais como se um dispositivo de saída é vetorial ou matricial, se é policromático ou não, etc.) pode-se fazer uso de uma série de funções de consulta através das quais o GKS responde sobre os valores de parâmetros da instalação e do estado do sistema.

### 3.2. Primitivas gráficas de saída

As imagens produzidas pelo programa, e apresentadas em dispositivos gráficos de exibição, são constituídas de segmentos de retas nos dispositivos vetoriais e de "pixels" (contração de "picture elements") nos dispositivos matriciais. Esses elementos básicos são primitivas gráficas do dispositivo. É de se esperar, assim, que os pacotes gráficos forneçam um conjunto mais amplo e elaborado de elementos gráficos básicos, tais como caracteres e símbolos, utilizáveis na composição de imagens. Esses elementos gráficos básicos, que são as primitivas gráficas de saída do pacote, devem ser passíveis de representação em qualquer dispositivo de saída gráfica disponível no sistema, independentemente das peculiaridades destas.

Cada primitiva gráfica de saída é caracterizada por sua geometria e por sua aparência na superfície de exibição do dispositivo de saída. Seu aspecto é controlado por um conjunto de atributos, ou seja, por propriedades que determinam a

aparência da primitiva no dispositivo. Exemplos de atributos de primitiva são a cor de um segmento de reta, o tamanho de um caractere e as coordenadas dos vértices de um polígono.

O GKS dispõe de cinco primitivas de saída: uma para traçado de linhas, uma para marcar pontos, uma para texto, uma orientada para dispositivos matriciais, e uma de uso geral.

### 3.2.1. "Polyline"

A primitiva "polyline" gera uma poligonal unindo uma sequência dada de pontos do plano por segmentos de reta. A poligonal é traçada em resposta à invocação da função "polyline" cujos parâmetros são os atributos geométricos da primitiva, ou seja a sequência de pontos.

São quatro os atributos de "polyline": tipo de linha, fator de escala da espessura, índice de cor e índice da "polyline". O tipo de linha especifica a textura do traçado de cada segmento de reta. Quatro tipos são prédefinidos e obrigatórios:

1. traço contínuo
2. traçado
3. pontilhado
4. traço-ponto

Outros tipos tem textura dependente da implementação.

O fator de escala de espessura controla a espessura do traçado de cada segmento. A espessura nominal do segmento (que depende do dispositivo que exibirá a "polyline" é multiplicada pelo fator de escala (um real) na obtenção da espessura a ser usada no traçado. O índice de cor (um inteiro) controla a cor com que a "polyline" será representada na superfície de exibição dos dispositivos de saída.



Como exemplo tem-se:

x(1) = 2; y(1) = 2;

x(2) = 4; y(2) = 2;

x(3) = 4; y(3) = 4;

x(4) = 2; y(4) = 4;

x(5) = 2; y(5) = 2;

call gpl(5,x,y)

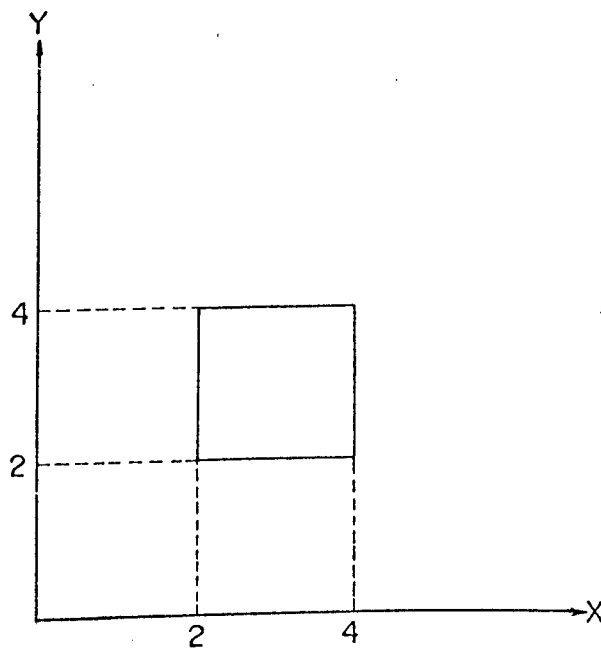


fig. 3.2.1 - Representação de uma "Polyline"

### 3.2.2. "Polymarker"

Esta primitiva tem por finalidade marcar pontos na superfície de exibição dos dispositivos de saída gráfica. As marcas resultam do traçado de figuras centradas em cada posição de uma sequência de pontos fornecida na invocação desta função.

O tipo de marca é o atributo que determina a figura utilizada no desenho da marca. Alguns valores são predefinidos e obrigatórios em qualquer implementação. Os tipos com valores

superiores a 5 estão associados a figuras dependentes da implementação do GKS.

Outros dois atributos da "polymarker" são o fator de escala de tamanho e índice de cor.

Como exemplo tem-se:

```
x(1) = 2; y(1) = 2;
```

```
x(2) = 4; y(2) = 2;
```

```
x(3) = 4; y(3) = 4;
```

```
x(4) = 2; y(4) = 4;
```

```
call gpm(4,x,y)
```

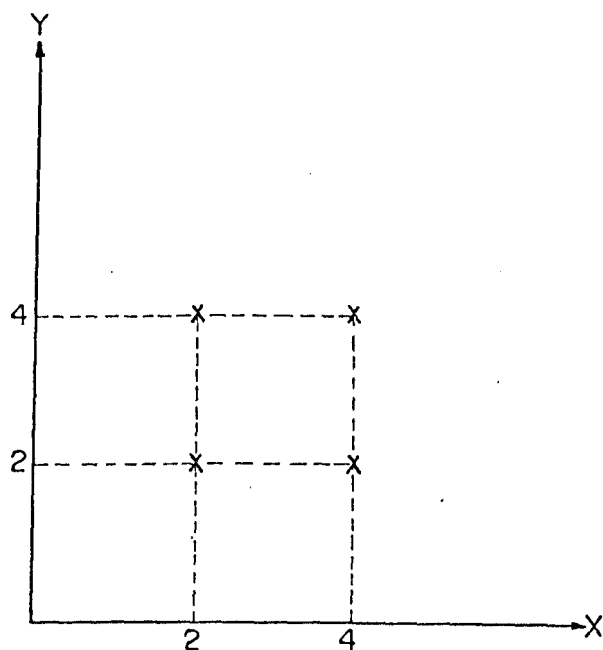


fig. 3.2.2. - Representação de uma "Polymarker"

### 3.2.3. "Text"

Informações de texto são frequentemente adotadas como complemento a informações gráficas. Rotulações e titulação em mapas e descrições do significado dos eixos de um gráfico são alguns exemplos da aplicação de textos em combinação com

gráficos. O GKS fornece uma primitiva para atender a esta necessidade.

A primitiva "text" é uma das mais complexas do GKS. Um total de nove atributos são utilizados na construção da figura representando o texto solicitado.

Os atributos da primitiva "text" são:

a) precisão: caracteriza a qualidade com que os atributos geométricos do texto são respeitados. Assume um entre os três valores: "string", "char", "stroke". A precisão "string" é a mais rudimentar e a precisão "stroke" a mais elaborada.

b) fonte: seleciona a fonte dos caracteres do texto. Em sistemas simples é de se esperar a disponibilidade de um único fonte atendendo a codificação da tabela ASCII. Em sistemas mais sofisticados, os fontes romanos, itálicos, cursivos, caracteres gregos, etc. podem ser incluídos.

c) altura: altura dos caracteres.

d) vetor de orientação: caracteriza a direção vertical de cada caractere do texto.

e) direção: caracteriza a direção com que cada caractere se posiciona em relação aos precedentes.

f) alinhamento: estabelece a localização precisa do ponto de partida do texto dentro da figura.

g) fator de expansão: define a relação largura/altura da quadrícula contendo cada caractere.

h) espaçamento: um real que especifica a distância entre dois caracteres subsequentes da cadeia.

i) índice de cor: controla a cor com que serão traçados os caracteres.

Como exemplo tem-se:

```
call gtxs(1.0,2.0,3,'GKS')
```

onde:

1.0 - posição inicial no eixo x

2.0 - posição inicial no eixo y

3 - número de caracteres

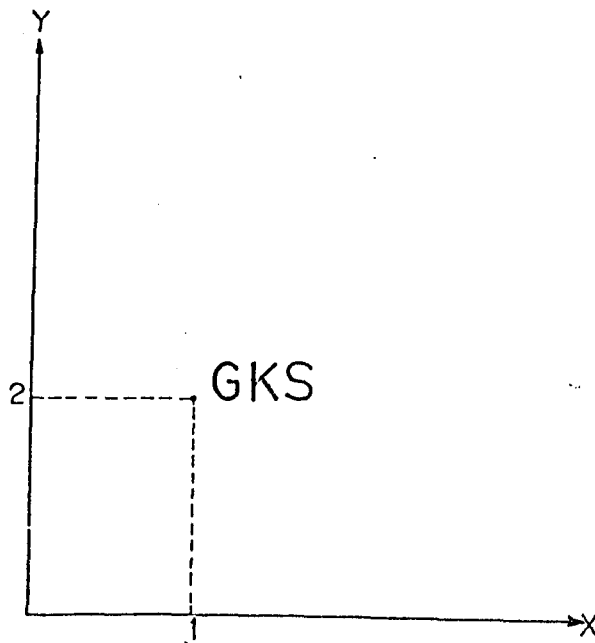


fig. 3.2.3 - Representação de um "Text"

#### 3.2.4. "Fill Área"

Esta é a primitiva definida pela caracterização de áreas poligonais da imagem. As áreas são especificadas por uma fronteira expressa por uma poligonal fechada. Desta forma, o último ponto não precisa ser fornecido (como na "polyline"). Bastam n pontos para especificar-se uma área cuja fronteira possua n vértices.

Há diversas alternativas de preenchimento de áreas geradas pelo GKS. O parâmetro estilo de preenchimento de área controla

a seleção dessas alternativas que são:

"hollow": é traçado apenas a poligonal de fronteira da área com o índice de cor de preenchimento.

"solid": o interior é preenchido uniformemente com o índice de cor do preenchimento.

"hatch": o interior é preenchido com uma hachura com o índice de cor de preenchimento. O tipo de hachura a ser usado é determinado pelo índice de estilo de preenchimento. Os tipos disponíveis são dependentes da implementação.

"pattern": o interior da área é preenchido com um padrão regular com características específicas, em uma entrada da tabela de padrões, apontada pelo índice de estilo de preenchimento.

A única condição obrigatória da especificação do GKS é de que dispositivos vetoriais representem "fill área" pelo menos no estilo "hollow", em resposta a qualquer estilo de preenchimento. Dispositivos matriciais devem, entretanto, atender a todas as especificações de estilo de preenchimento.

Como exemplo tem-se:

```
x(1) = 2; y(1) = 2;
```

```
x(2) = 4; y(2) = 2;
```

```
x(3) = 4; y(3) = 4;
```

```
x(4) = 2; y(4) = 4;
```

```
call gfa(4,x,y)
```

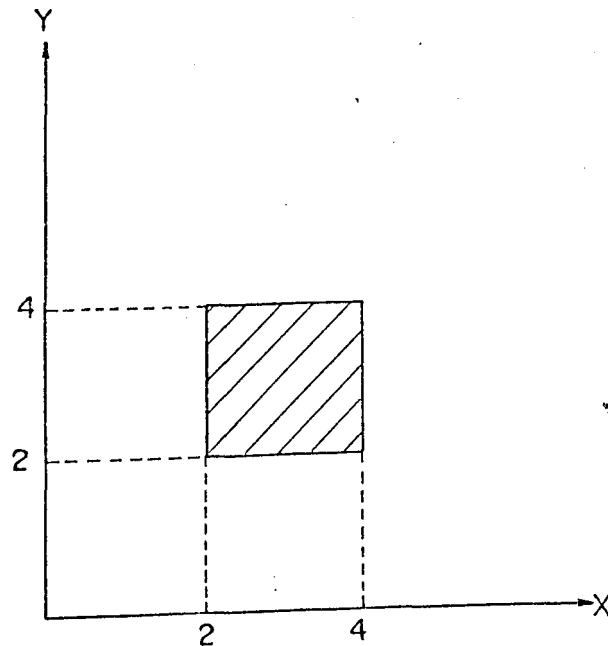


fig. 3.2.4 - Representação de uma "Fill Área"

### 3.2.5. GDP

A primitiva generalizada de desenho ("Generalized Drawing Primitive"), está prevista no GKS como um recurso padronizado para dar acesso a primitivas especiais aceitáveis por dispositivos de saída específicos e possivelmente não previstas entre as demais. Exemplos de aplicações potenciais desta primitiva estão entre o traçado de cônicas, "splines" e pintura de regiões.

A portabilidade do programa aplicação que faz uso de GDP's evidentemente fica limitada a instalações onde a mesma primitiva esteja definida. A função GDP do GKS admite quatro parâmetros: dois especificando uma sequência de pontos, um identificando qual primitiva é solicitada, e o quarto contendo informações necessárias ao traçado da primitiva.

Como exemplo tem-se:

```
call ggdp(3,x,y,2,Idr,datrec)
```

onde:

3. número de pontos

x(1). abcissa do centro do arco

x(2). abcissa do ponto inicial do arco

x(3). abcissa do ponto final do arco

y(1). ordenada do centro do arco

y(2). ordenada do ponto inicial do arco

y(3). ordenada do ponto final do arco

2. identificador de arco

Idr. não usado nesta implementação.

datrec. não usado nesta implementação

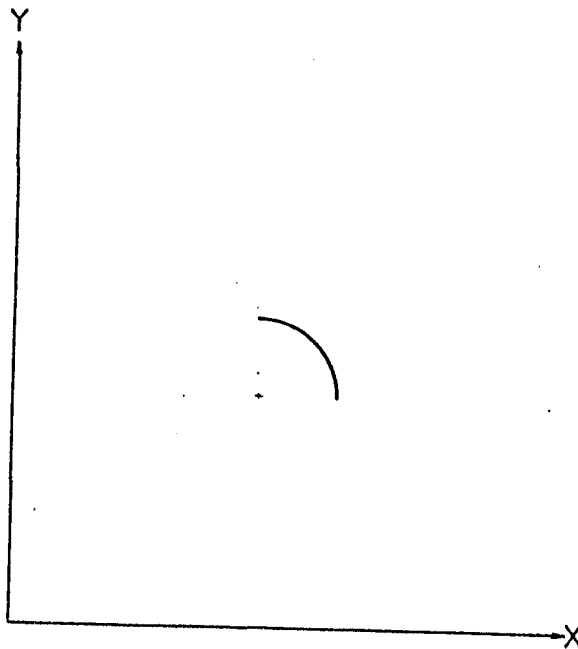


fig. 3.2.5 - Representação de uma "GDP"

### 3.3. Sistemas de coordenadas

As primitivas gráficas descritas anteriormente são abstrações de entidades bidimensionais. Para a descrição e representação precisa dos atributos geométricos dessas entidades, é necessário fixar-se um sistema de coordenadas. Diferentes aplicações podem requisitar diferentes sistemas de coordenadas por serem naturais na representação do fenômeno em estudo.

Em todo sistema gráfico, o sistema de coordenadas em uso precisa estar bem definido. Alguns sistemas gráficos oferecem sistemas alternativos de coordenadas. Entretanto, a maioria das aplicações requer ou satisfaz-se com um sistema cartesiano de coordenadas com escala linear, o que justifica a escolha deste na maioria dos sistemas gráficos, e no GKS em particular. Se uma dada aplicação manipular dados em outros sistemas de coordenadas, deverá prover ela própria as transformações de seu sistema de coordenadas para um sistema cartesiano. Para diferenciar esses dois sistemas de coordenadas, denominam-se de sistemas de coordenadas do usuário, o sistema de coordenadas preferido pelo programa aplicação e, de sistemas de coordenadas universais, o utilizado para descrever atributos geométricos de primitiva do GKS.

O GKS define uma superfície de exibição virtual representada pelo quadrado  $[0,1] \times [0,1]$ , ou seja, pelos pontos de coordenadas  $(x,y)$  tais que  $0 \leq x \leq 1$  e  $0 \leq y \leq 1$ . Coordenadas neste quadrado são denominadas coordenadas normalizadas.

A área do plano formada por esse quadrado engloba as



figuras passíveis de serem exibidas. O quadrado é uma tela virtual onde as diversas figuras constituídas de primitivas gráficas compõem uma figura virtual. Para que as primitivas gráficas de saída com atributos geométricos expressos em coordenadas virtuais possam ser representadas na tela virtual, estabelece-se uma transformação linear entre os dois espaços. Uma maneira conveniente de caracterizar-se uma transformação com essa finalidade, é definir-se uma área retangular no sistema de coordenadas universais, e especificar-se um segundo retângulo na tela virtual onde a primeira área é mapeada. Estes retângulos são tradicionalmente denominados "window" e "viewport", respectivamente, e são tomados como lados paralelos aos eixos coordenados.

Esta transformação linear é denominada de transformação de normalização, e uma vez definida através da especificação da "window" e da "viewport", as primitivas gráficas de saída expressas em coordenadas universais, podem ser mapeadas no sistema de coordenadas do dispositivo normalizado no GKS.

Como exemplo de transformação de coordenadas tem-se:

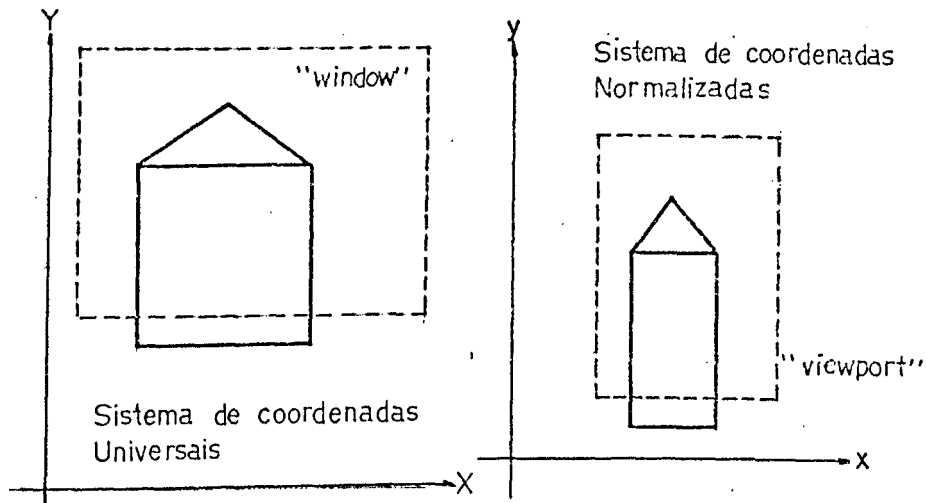


fig. 3.3 - Transformação de coordenadas

#### 3.4. Segmentação

Os recursos do GKS discutidos até aqui permitem a construção de figuras através de primitivas e o seu posicionamento na superfície de exibição pelo uso das transformações de normalização. Uma vez construídas, as figuras não podem ser modificadas, eliminadas ou reposicionadas, a menos que toda imagem seja apagada e reconstruída pelo programa aplicação. Esta restrição torna o sistema gráfico insatisfatório para aplicações interativas onde o operador da estação gráfica deseja produzir alterações e eliminações em partes selecionadas do desenho.

Quando se considera a hipótese de incluir-se mecanismos de alterações dinâmicas das imagens em um sistema gráfico, surgem duas questões básicas a serem respondidas:

a) que partes da imagem serão submetidas a uma dada alteração ?

b) Qual elenco de alterações dinâmicas básicas estarão disponíveis no sistema gráfico ?

A primeira questão pode ser respondida pela estruturação da imagem em partes identificadas por um número que será usado na referência da parte a ser alterada. Estas partes são constituídas por um conjunto de primitivas de saída formando uma unidade logicamente definida, e que se submeta a alterações solicitadas. Desta forma, a estruturação da imagem em unidades lógicas fica a cargo do programa de aplicação refletindo numa organização adequada ao objetivo. A esses conjuntos de primitivas que formam unidades referenciáveis em alterações denominam-se segmentos.

A escolha do elenco de alterações dinâmicas a que podem ser submetidos os segmentos, exige certo critério. Um elenco muito extenso contendo funções muito elaboradas, como por exemplo as necessárias para efeitos de animação, poderia tornar o sistema gráfico excessivamente grande e/ou exigir recursos de "hardware" mais sofisticados. Um elenco pobre teria pouca utilidade. Um repertório de funções de alterações de segmentos do GKS não é muito grande, mas atende aos requisitos de grande número de aplicações. Além das operações primitivas de criação e eliminação de segmentos, o GKS oferece a possibilidade de alterar-se os atributos dos segmentos. Estes atributos dos segmentos controlam a visibilidade, a localização, o realce, a prioridade e a detectabilidade das primitivas no segmento.

### 3.4.1. Construção de segmentos

Os segmentos do GKS são criados por uma função com argumento  $n$ , um inteiro que identifica o segmento que está sendo criado. As primitivas de saída invocadas enquanto  $n$  estiver aberto são inseridas neste segmento, além de serem exibidas. A especificação do segmento aberto fica concluída com a invocação de uma função que fecha o segmento. A partir de então as primitivas subsequentes não são incluídas naquele segmento e não há mais segmento aberto.

Uma vez criados, segmentos podem ser definitivamente eliminados por meio de uma função que provoca o desaparecimento das primitivas que o compõe como se nunca tivessem sido invocadas.

Como exemplo de construção de um segmento temos:

```
call gcrsg(10) (cria o segmento)
call gspkid(101) (define um número para identificar a
polilyne abaixo)
call gpl(3,x1,y1) (traça a polilyne)
call gspkid(102) (define um número para identificar a
polilyne abaixo)
call gpl(4,x2,y2) (traça a polilyne)
call gclsg (fecha o segmento)
```

### 3.4.2. Atributos de segmentos

#### a) Transformação de segmentos

Além das transformações de normalização correntes, as entidades geométricas dos segmentos estão sujeitas a uma segunda transformação linear, característica e única para todo o segmento, denominada transformação de segmento. Esta é uma

transformação linear geral (em contraste com a transformação de normalização que compõe uma transformação e uma escala) permitindo a rotação e reflexão das primitivas.

b) Visibilidade

Este atributo controla se as primitivas do segmento são exibidas ou não. Ao ser criado o segmento, este atributo possui valor "visible" e o segmento normalmente durante sua criação ou após, o valor deste atributo pode ser alterado para "invisible", o que faz com que as primitivas desapareçam temporariamente até que o atributo seja novamente alterado para "visible".

c) Realce

Um segmento pode ser exibido com um destaque especial por intermédio deste atributo. Quando o atributo de realce passa do valor normal ao valor "highlighted", o segmento é exibido com um atributo visual que o destaca dos não realçados: um aumento de intensidade, uma cor especial, um efeito de cintilação, etc. A forma do realce é dependente da implementação do GKS.

d) Prioridade de segmentos

Quando uma imagem apresenta várias áreas cheias criadas por primitivas "fill área" surge uma questão: o que será exibido na intersecção destas áreas? O GKS define um mecanismo através do qual é possível exercer-se um controle parcial sobre qual primitiva será exibida nas áreas de intersecção. A prioridade de um segmento é um número real no intervalo  $[0,1]$ . Um segmento é exibido sempre na frente dos segmentos de prioridade inferior. Os resultados são indefinidos para segmentos de mesma prioridade ou primitivas dentro de um mesmo segmento.

### e) Detectabilidade

Este atributo só faz sentido em conjunção com dispositivos lógicos de entrada do tipo apontador ("pick device"). Este dispositivo permite que o operador de uma estação, pela manipulação de um dispositivo físico conveniente (teclado, mesa digitalizadora, etc.), selecione um segmento ou parte dele. Feita a seleção, o programa de aplicação receberá do GKS informações identificando o segmento ou a parte selecionada.

O atributo de detectabilidade permite controlar se um segmento pode ou não ser selecionado pelo operador do apontador. Se este atributo possui valor "detectable", o operador tem a possibilidade de selecioná-lo. Se o valor do atributo for "undetectable" (valor "default" de criação) o dispositivo de apontador fica "cego" a este segmento e o operador não consegue selecioná-lo.

Como exemplo de atributo de segmento tem-se:

```
call gcrsg(20) (cria o segmento)
call gtxs(2.0,3.0,4,'text') (traça o texto)
call gclsg (fecha o segmento)
call gsvs(20,0) (torna o texto invisível)
```

### 3.5. Dispositivos lógicos de entrada

A variedade de dispositivos gráficos de entrada, possibilita que uma informação gráfica de certo tipo possa ser suprida de diversas formas. Assim, por exemplo, a localização de um ponto no plano pode ser introduzida, dando-se suas coordenadas pelo teclado ou, alternativamente, fazendo-a corresponder à posição, no momento em que se aperta o botão de

um cursor cujo movimento sobre a tela de um terminal é controlado por um "mouse" ou "joystick". Além disso, a localização de um ponto pode ser fornecida via mesa digitalizadora. Desse modo, dispositivos físicos de entrada precisam ser associados a dispositivos lógicos de entrada (DLE) para que se possa fazer uso de toda essa diversidade de equipamento de uma forma independente de configuração. Um programa fará então referência a um dispositivo lógico deste tipo, sem ter que se preocupar com o meio físico por intermédio do qual essa entrada é adquirida.

No GKS, os dados de entrada para um programa de aplicação foram classificados em seis tipos distintos, estabelecendo-se, para cada um deles, uma classe de DLE correspondente. Essas classes são:

a) Posicionador ("locator"), que fornece ao programa uma posição (x,y) do plano.

b) Cadeia de posições ("stroke"), que fornece uma sequência de posições (x,y) do plano.

c) Seletor ("choice"), que dá um valor inteiro não negativo pertencente a uma dada faixa e que serve para selecionar uma opção dentre as de um conjunto finito.

d) Quantificador ("valuator"), que passa para o programa um valor real.

e) Apontador ("pick"), que identifica um segmento que está sendo exibido ou parte dele.

f) Cadeia de caracteres ("string"), que fornece uma cadeia de caracteres.

Essa classificação, portanto, diz respeito apenas ao tipo de dado que o DLE entrega ao programa não havendo, em

princípio, restrições quanto ao tipo de dispositivo físico que pode ser associado a um DLE de uma dada classe qualquer.

### 3.5.1. Classe de dispositivos lógicos de entrada

#### a) Localizador

Um dispositivo localizador passa para o programa aplicativo a posição de um ponto em coordenadas universais. Normalmente, entretanto, o que os dispositivos físicos localizadores captam é uma posição na tela ou na mesa digitalizadora, devendo portanto processar essa informação trilhando todo o caminho inverso do processo de visualização, até obter o ponto indicado em coordenadas universais.

Os dispositivos físicos chamados localizadores são: "mouse", "joystick", principalmente.

#### b) Cadeia de posições

Um DLE da classe de cadeia de posições permite passar ao programa de uma só vez toda uma sequência de pontos evitando que se tenha de fazer isso através da utilização reiterada de um dispositivo localizador.

Os dispositivos físicos deste tipo são: "mouse", "joystick", e mesa digitalizadora, principalmente.

#### c) Apontador

Um DLE da classe apontador retorna para o programa o número de um segmento ou parte dele, atribuindo-lhe um identificador de apontador durante a criação do segmento, contendo uma primitiva apontada pelo dispositivo físico.

Os dispositivos físicos chamados apontadores são caneta de luz, "joystick", etc.



## d) Quantificador

Um DLE quantificador entrega ao programa um valor real.

Os quantificadores podem ser associados a teclado ou potenciômetro.

## e) Seletor

A realização de um DLE seletor num dispositivo físico pode variar consideravelmente. Pode-se, por exemplo, escolher pelo teclado uma ação a ser tomada de um conjunto de alternativas que está sendo mostrado na tela. Também pode-se associar cada opção a uma área específica da tela, onde possivelmente está escrito um texto qualquer que identifique a opção; escolhe-se uma alternativa simplesmente movendo o cursor para dentro da área a ela associada. Outra alternativa comum é efetuar a seleção apertando-se um determinado botão de um painel, ou uma tecla de função.

Os dispositivos físicos associados a este DLE são chaves seletoras ou teclas de funções ou um dispositivo físico localizador qualquer usado para selecionar uma opção de um menu exibido na tela.

## f) Cadeia de caracteres

Um DLE deste tipo entrega ao programa uma cadeia de caracteres.

Está associado a este tipo de dispositivo um teclado.

## 3.5.2. Modo de operação de DLEs.[15]

Os três modos de operação dos dispositivos lógicos de entrada especificam quem (o operador ou o programa aplicação) tem a iniciativa. O modo "sample" é adquirido diretamente pelo programa aplicação, o modo "request" é produzido pelo operador

em resposta direta ao programa aplicação, e o modo "event" é gerado assincronamente pelo operador.

Estes três modos de operação trabalham da seguinte forma:

a) "Request": O programa aplicação e o processo de entrada trabalham alternativamente. Primeiro, o programa aplicação requer uma entrada e então espera por uma resposta. O processo de entrada é acessado pelo "request", entrega um dado de entrada ao programa aplicação, e retorna ao estado de espera. Um ou outro processo está ativo mas não ambos.

b) "Sample": O programa aplicação e o processo de entrada são ativados juntos, mas o programa aplicação é a parte dominante. O processo de entrada trabalha em segundo plano, fornecendo o último dado de entrada para o dispositivo, que pode ou não ser usado pelo programa aplicação. Por outro lado, o programa aplicação continua executando, tomando e usando os dados de entrada corrente de um dispositivo quando ele o requerer.

c) "Event": O programa aplicação e o processo de entrada também são ativados juntos, mas a parte dominante é o processo de entrada. Ele entrega os dados de entrada ao programa aplicação e espera ele atuar, dependendo do dado recebido. O operador controla quando o dado de entrada está disponível, e efetivamente direciona a iteração.

### 3.6. Estações de trabalho

A maior parte dos recursos e mecanismos do GKS descritos até aqui configuram um sistema gráfico onde especificidades de periféricos gráficos não são considerados: definem o modelo de

GKS independentes de dispositivos. Entretanto, quando as abstrações representadas pelas primitivas gráficas de saída precisam concretizar-se em uma imagem na tela de um terminal ou na mesa de um traçador, vários fatores específicos à natureza de cada dispositivo físico devem ser levados em conta.

O GKS concentra no conceito de estação de trabalho as particularidades dos equipamentos periféricos disponíveis na instalação. O conceito não só unifica a possível diversidade de características dos periféricos de exibição, como abriga outros recursos desejáveis em um sistema gráfico, tais como: dispositivos de entrada, periféricos para armazenagem e para a transmissão de figuras. O conceito de estação de trabalho é um dos pontos mais fortes introduzidos pelo GKS que permite a um programa de aplicação operar diversas estações simultaneamente de forma homogênea.

Há 6 categorias de estações de trabalho funcionalmente diferenciadas:

a) Estações de saída ("OUTPUT - output workstation"): são estações com capacidades de exibição de primitivas gráficas e sem recursos de entrada; um exemplo típico dessa categoria é um traçador ("plotter").

b) Estações de entrada ("INPUT - input workstation"): são estações com capacidades de entrada apenas, como por exemplo, uma mesa digitalizadora.

c) Estações interativas ("OUTIN - output-input workstation") são estações com capacidade de exibição gráfica além de incluírem recursos de entrada; a maioria dos terminais de vídeo está nesta categoria pois, além da tela, incluem um teclado e por vezes canetas de luz, "mouses" e mesa digitalizadora.

d) Armazenador independente de segmentos ("WISS" - "workstation independente segment storage"): uma estação virtual especial, sem recursos de entrada ou exibição, mas com capacidade ampliada de armazenagem temporária de segmentos, oferecendo facilidades de cópia de seus segmentos para outras estações.

e) Metarquivo de saída ("MO - metafile output"): estações virtuais especiais, sem recursos de entrada ou exibição cuja função é a armazenagem a longo prazo de figuras ou de sua transmissão a outras instalações.

f) Metarquivo de entrada ("MI - metafile input"): estações virtuais especiais, sem recursos de entrada ou exibição gráficas, cuja função é a recuperação de figuras ou entidades gráficas previamente armazenadas ou transmitidas por outras instalações por meio de metarquivos de saída.

## 4. ESTRUTURA COMPUTACIONAL DOS PROGRAMAS GRÁFICOS

### 4.1. Programa Rotor

O programa Rotor oferece várias opções ao usuário de acordo com as necessidades de análise do projeto da máquina rotativa em questão. Tais opções podem variar desde um simples cálculo estático com forças externas somente, até um cálculo completo envolvendo a dinâmica do sistema giroscópico e a resposta à excitações síncronas ou assíncronas. Estas possibilidades são organizadas em um menu com 7 opções. Feita a opção, procede-se a montagem do arquivo de entrada do programa onde são carregados os dados do problema: posições dos nós, características geométricas do rotor, parâmetros físicos dos vários elementos considerados (barras, discos, mancais e outros) informações sobre os tipos de cálculo a serem efetuados.

#### 4.1.1. Estrutura do arquivo de dados de entrada

O arquivo de dados do programa Rotor é construído de forma estruturada, isto é, os dados que se relacionam são organizados em blocos. Tais blocos são separados por um cartão, onde são colocados os inteiros 99999.

Os dados são colocados no arquivo de entrada de forma lógica: os cartões tipo 1 são reservados para a descrição do problema que está sendo analisado; o cartão tipo 2 é usado para os dados mais gerais do arquivo, como o índice que se relaciona com a opção de análise a ser efetuada pelo programa, número de nós usado no modelo elementos finitos do rotor, número de

frequências naturais desejadas, controle de impressão dos dados de entrada, etc.

Os cartões que se seguem dizem respeito somente às propriedades físicas da máquina rotativa, tais como características dos nós, e elementos e materiais que o compõem. A seguir, se necessário, são colocados amortecimentos modais, peso próprio e forças externas.

Se no índice escolhido anteriormente constam o cálculo da resposta à uma excitação ou dinâmica em rotação, a introdução de alguns cartões adicionais são necessários, informando: a faixa de velocidades de rotação em que se deseja obter os resultados, assim como o valor de um passo para obtenção dos valores intermediários; as características físicas da excitação, como número do nó onde esta é aplicada e ângulo de posição, massa do desbalanceamento e sua excentricidade para uma excitação síncrona, ou intensidade da força para uma excitação assíncrona. No caso de se desejar a resposta do sistema, é preciso entrar com os números dos nós onde se deseja a resposta.

Segue abaixo a sequência de cartões do arquivo de entrada:  
cartão tipo 1: título (2 cartões)  
cartão tipo 2: dados gerais (1 cartão)

INDIC: 0 - Cálculo estático - somente forças externas.

1 - Cálculo estático - peso próprio além das forças externas.

2 - Dinâmica do sistema não-giroscópico -  
determinação das frequências e dos auto-vetores que serão utilizados como base modal para os cálculos seguintes. São calculadas as matrizes modais.

3 - Resposta a um desbalanceamento ou a uma força assíncrona.

4 - Dinâmica do sistema giroscópico (unicamente).

5 - Dinâmica do sistema não-giroscópico e dinâmica do sistema giroscópico (2 + 4).

6 - Dinâmica do sistema não-giroscópico e resposta a um desbalanceamento ou força assíncrona (2 + 3).

cartão tipo 3: coordenadas dos nós (tantos quantos necessário)

cartão tipo 4: condições de contorno (tantos quantos necessário)

cartão tipo 5: tipo e número de elementos (1 cartão para cada tipo de elemento)

cartão tipo 6: descrição dos elementos (tantos quantos necessário)

cartão tipo 7: características dos materiais (tantos quantos necessário)

cartão tipo 8: propriedades geométricas (tantos quantos necessário)

cartão tipo 9: descrição da forma de entrada de dados dos elementos adicionais (1 cartão por elemento)

cartão tipo 10: leitura dos dados da matriz linha por linha.  
( 4 ou 8 cartões)

cartão tipo 11: leitura dos valores indicando número da linha, número da coluna e o valor. (4 ou 8 cartões)

cartão tipo 12: número de amortecimentos modais. (1 cartão)

cartão tipo 13: valores dos amortecimentos modais. (tantos quantos necessário)

cartão tipo 14: peso próprio. (1 cartão)

cartão tipo 15: forças externas. (tantos quantos necessário)

cartão tipo 16: faixa de velocidade para cálculo das frequências e modos, resposta a um desbalanceamento ou força assíncrona, e frequências de excitação para o cálculo da resposta a uma excitação assíncrona (tantos quantos necessário)

cartão tipo 17: descrição da massa de desbalanceamento (tantos quantos necessário)

cartão tipo 18: descrição da força assíncrona (tantos quantos necessário)

cartão tipo 19: número dos nós onde se deseja a resposta (tantos quantos necessário)

A referência [8] contém todos os detalhes relativos aos cartões acima (formatos, ordem de entrada, etc.)

#### 4.1.2. Estrutura do Programa Rotor

O início do programa principal consta da leitura dos dados gerais e das características da máquina rotativa onde são montadas as matrizes elementares de rigidez, massa, coriolis e amortecimento.

Se a opção para o cálculo estático indica  $INDIC = 0$ , considera-se somente as forças e momentos externos. Para  $INDIC = 1$ , são calculadas as forças nodais devidas ao peso do rotor, e eventualmente são consideradas as forças e momentos externos.

Com as opções acima, somente as matrizes rigidez elementares são montadas, de modo a se obter a matriz rigidez total do sistema. Assim, para problemas estáticos a solução é obtida resolvendo um sistema linear pelo método de Gauss, onde é encontrada a deformação estática.



Para os cálculos seguintes são calculadas as frequências e vetores próprios do sistema não-giroscópico que serão utilizados como base modal. Para isso são montadas as matrizes massa e rigidez elementares. A base modal é constituída com as  $n$  menores frequências próprias e  $n$  vetores próprios correspondentes do sistema simétrico:  $K \phi = M \phi \omega^2$  comportando originalmente  $N$  equações ( $N \gg n$ ).

Para a dinâmica em rotação, a busca das frequências e modos sobre o sistema completo conduz à determinação dos auto-valores e auto-vetores de :  $M \ddot{p} + D \dot{p} + K p = 0$ , sendo  $D = A + \Omega C$ , onde  $A$  e  $C$  são as matrizes de amortecimento e giroscópica do sistema completo, respectivamente.

Para a resposta a um desbalanceamento ou força assíncrona são calculadas as componentes em  $x$  e  $z$  das forças de excitação nos nós em que estas atuam. Estas serão as forças que comporão a equação não homogênea do sistema. A resolução deste sistema é feita usando a base modal do sistema não-giroscópico anteriormente calculada, e procura-se soluções para o sistema não homogêneo. Para a resposta a um desbalanceamento, a força de excitação atuante no nó é dada em função da massa desbalanceada, de sua excêntricidade, e da rotação do rotor. Para uma força assíncrona a força de excitação não é mais função da velocidade de rotação do rotor, supondo-se, porém, sua amplitude como sendo constante.

#### 4.1.3. Estrutura do arquivo de saída

Para a resolução de um problema de análise dinâmica, seja para um sistema giroscópico ou não-giroscópico, é necessário a determinação dos vetores próprios e o cálculo das matrizes

modais. Executando-se o programa Rotor com os INDICs = 2, 5, 6 ou 7, é criado um arquivo de saída onde é escrita a base modal da máquina rotativa em estudo. Assim, caso seja feito posteriormente um novo cálculo visando a obtenção da resposta do sistema à uma excitação, ou o estudo de sua dinâmica em rotação, INDIC = 3 e 4 respectivamente, a base modal é simplesmente lida do arquivo antes citado. Isto garante significativa redução no tempo de processamento. Caso não se tenha feito nenhum cálculo anterior da máquina rotativa e deseja-se utilizar os INDICs = 3 ou 4, basta apenas executar o programa anteriormente com o INDIC = 2 onde será calculado a base modal que será utilizada posteriormente.

#### 4.2. Construção das rotinas gráficas para o programa Rotor

As rotinas gráficas são construídas em dois programas periféricos ao programa Rotor. O primeiro programa, isto é, o programa de pré-processamento do programa Rotor auxilia na verificação dos dados do arquivo de entrada através da construção no terminal de vídeo da malha da máquina rotativa em estudo, composta dos elementos de barra, discos rígidos e mancais. Este programa consta de um menu que dá ao usuário a opção de construção do arquivo de dados de entrada na forma interativa/conversacional, ou a utilização de um arquivo já construído. Em ambos os casos através de chaves, pode-se verificar as características dos vários elementos incluídos no arquivo de entrada.

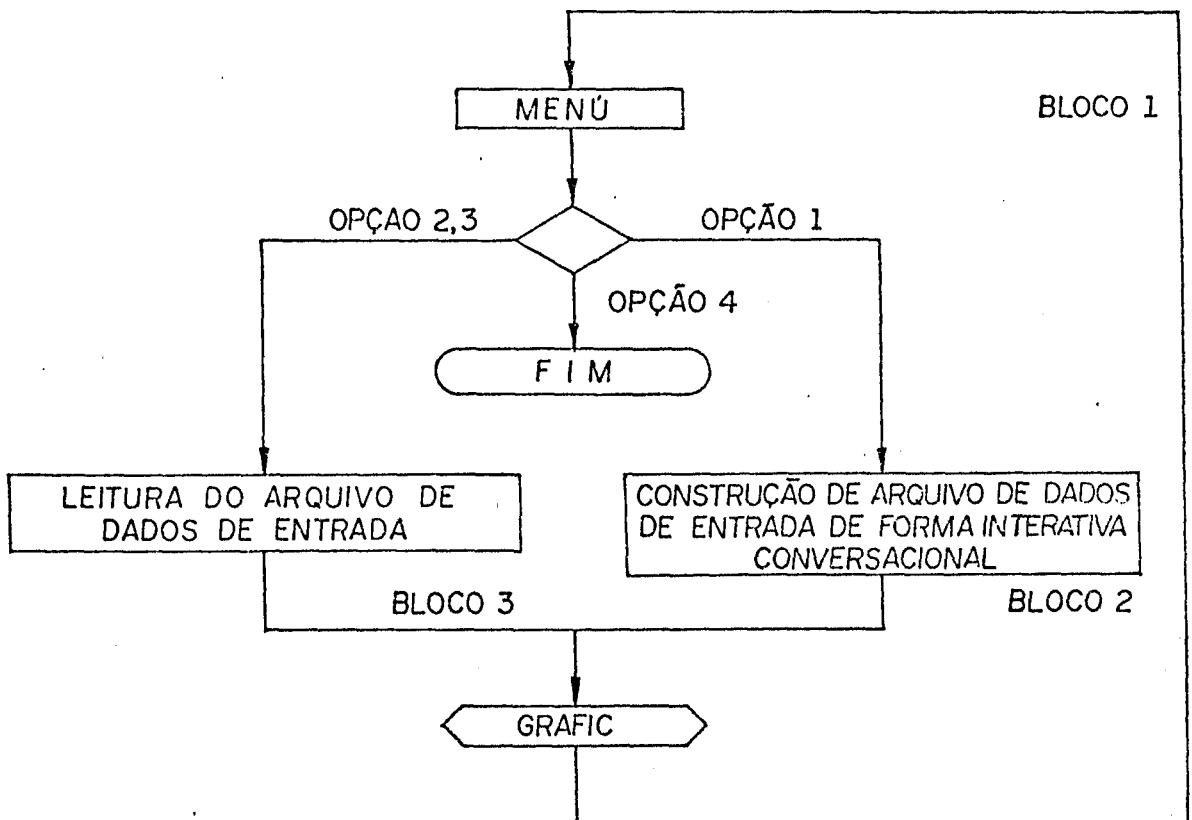
O segundo programa, ou o programa de pós-processamento do programa Rotor, trabalha com um outro arquivo de saída,

especialmente construído pelo programa Rotor, onde constam todas informações necessárias para a construção dos gráficos dos resultados.

Este programa, após a leitura do arquivo de saída, oferece também em um menu, as opções de construção do diagrama de Campbell, resposta do sistema a um desbalanceamento ou excitação assíncrona, e representação dos modos, desde que o índice escolhido no arquivo de entrada de dados contenha a opção que leve ao cálculo destes resultados.

4.2.1. Estrutura do programa de pré-processamento do programa Rotor

1 - ORGANOGRAMA DO PROGRAMA PRINCIPAL



## II - Estrutura do programa principal

### BLOCO 1

Este bloco contém o menu que expõe ao usuário as opções que o programa oferece, que são:

- 1 - Construir novo arquivo
- 2 - Visualização do rotor no vídeo
- 3 - Visualização do rotor na impressora
- 4 - Encerrar programa

A opção 1 é utilizada para a construção e visualização no terminal de vídeo de um arquivo novo; já as opções 2 e 3 são para a visualização no terminal de vídeo ou na impressora de um arquivo construído anteriormente pela opção 1. A visualização de um rotor na impressora, após a construção de um novo arquivo, somente é feita retornando-se ao menu principal e escolhendo-se a opção 3.

### BLOCO 2

A construção de um arquivo de forma interativa/conversacional é feita seguindo-se o manual do programa Rotor, e uma explicação mais detalhada será feita apenas sobre os cartões que contém informações a serem processadas graficamente:

#### Cartão tipo 3

É lido o número, coordenada do nó e um código de geração automática que utiliza o valor 1 para gerar elementos de mesmo comprimento. Assim, utilizando-se deste código, é montado um vetor com as coordenadas dos nós.

#### Cartão tipo 4

É lido o número do nó e as condições de contorno a que está submetido o nó. No vetor MANC são armazenados os valores 0, 1 ou 2 para mancais flexíveis, mancais do tipo apoio simples ou engastamento, respectivamente.

#### Cartão tipo 5

Especifica os tipos de elementos constitutivos da máquina rotativa como: elemento de barra, elemento de disco rígido, elemento de mancal, além de elementos de rigidez, massa, coriolis e amortecimento adicionais, que são armazenados.

#### Cartão tipo 6

Descreve os elementos do tipo indicado anteriormente pelo cartão tipo 5.

Para elementos de barra, são montados vetores com o número do material e o número da propriedade geométrica de cada elemento. Para elementos de disco rígido, são montados vetores com posição do disco, número da propriedade geométrica e número do material. E para os mancais, são montados vetores com as posições, e número da propriedade geométrica para rigidez e amortecimento. Para os elementos adicionais são armazenados apenas os números dos nós a que estão ligados.

#### Cartão tipo 7

Armazena as características dos materiais constituintes da máquina rotativa em questão.

#### Cartão tipo 8

Armazena as propriedades geométricas dos elementos de barra, discos rígidos e mancais.

## Cartão tipo 9

Define a característica da matriz do elemento adicional, se é simétrica ou não simétrica, maneira de dar entrada nos valores e número de cartões necessários para interpretá-la.

## Cartão tipo 10

Entra com os valores da matriz de um elemento adicional, linha por linha.

## Cartão tipo 11

Entra com os valores da matriz de um elemento adicional, especificando a linha, a coluna, e o valor.

## Cartão tipo 15

Armazena as forças e momentos externos.

## Cartão tipo 17

Armazena as características da massa de desbalanceamento.

## Cartão tipo 18

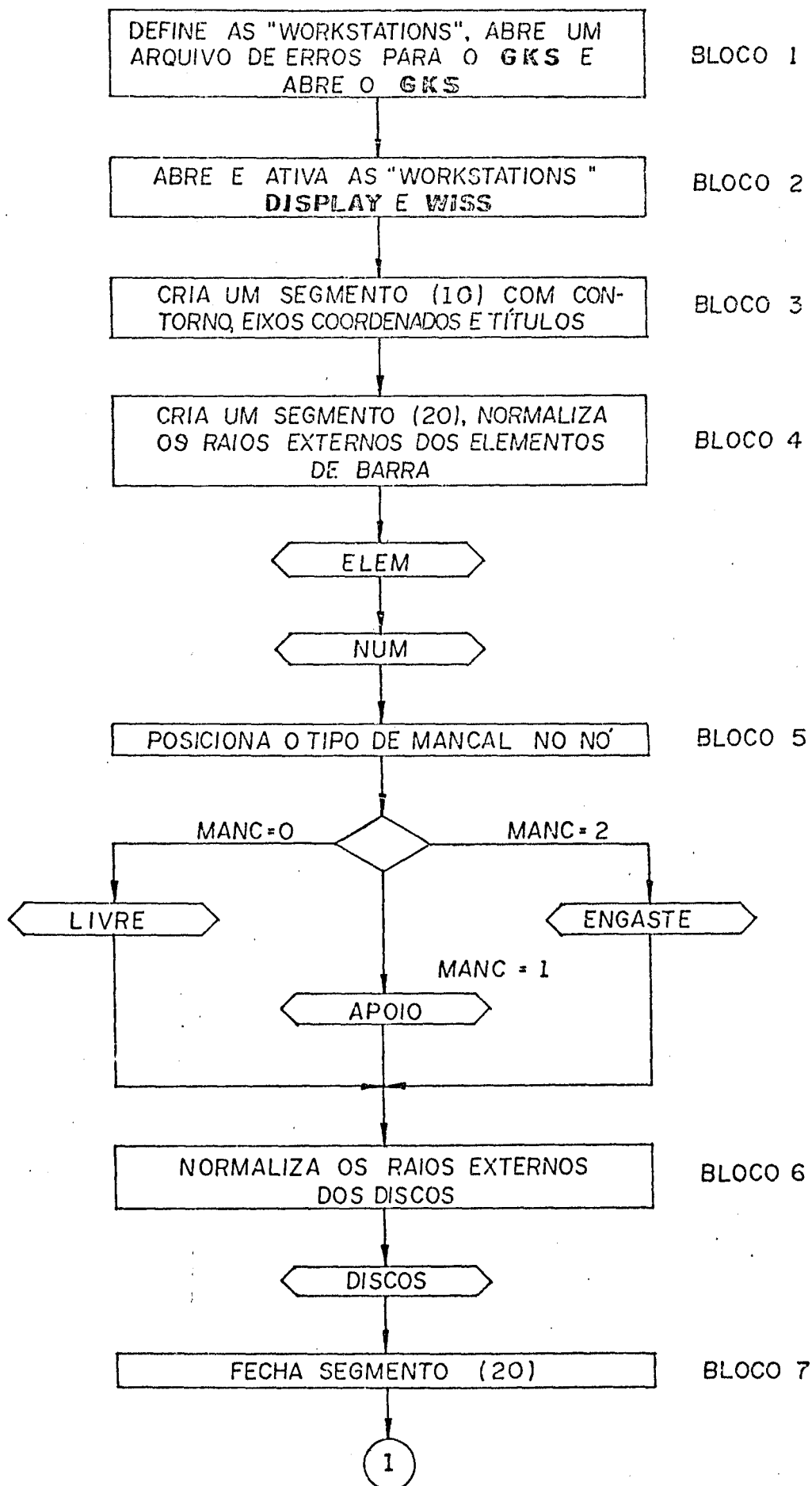
Armazena as características da força assíncrona.

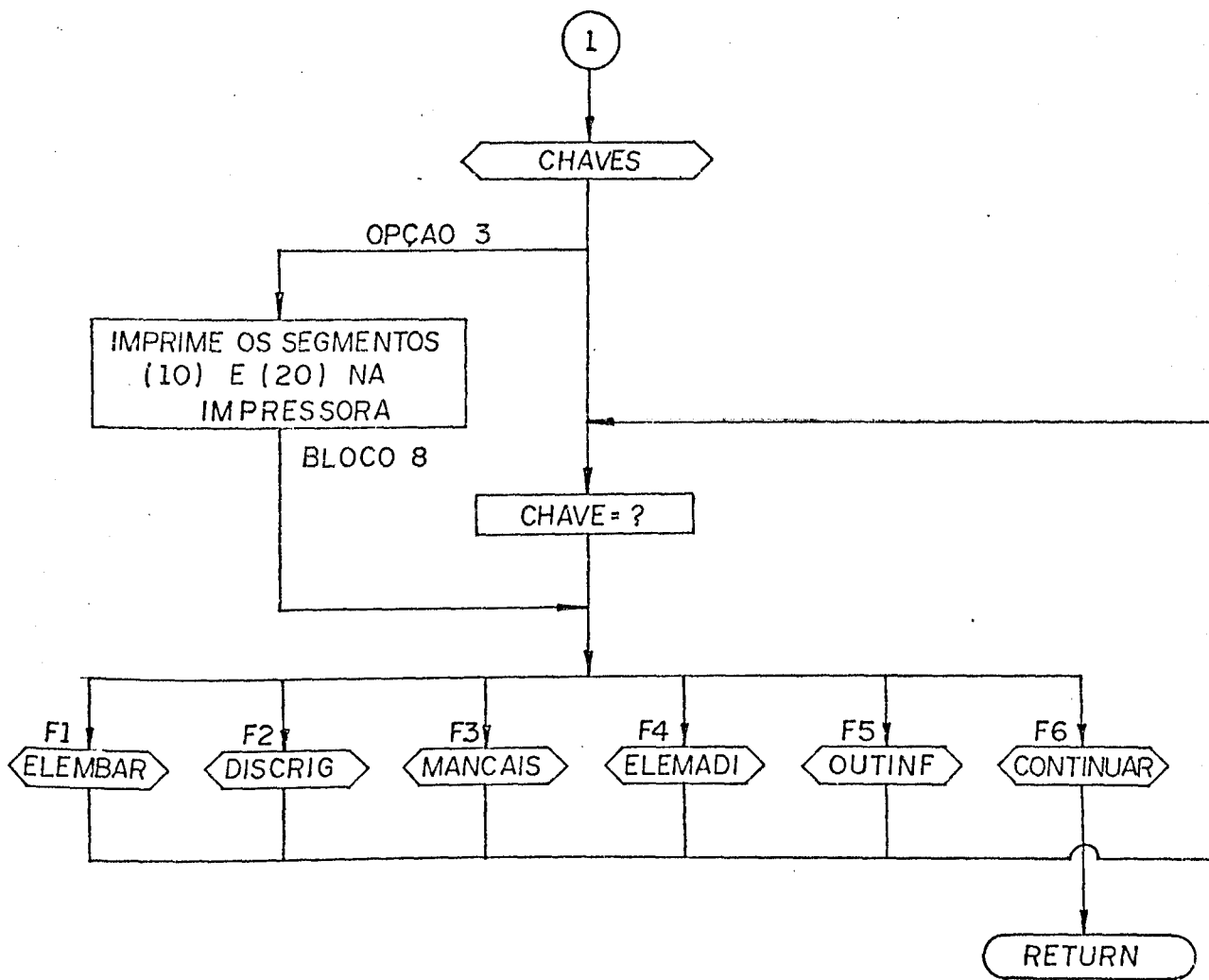
## Cartão tipo 19

Armazena os números dos nós onde se deseja a resposta.

## BLOCO 3

Da mesma forma que no BLOCO 2, assim que os dados são lidos do arquivo de entrada, estes são armazenados em vetores específicos.

III - ORGANOGRAMA DO SUB-PROGRAMA **GRAFIC**





## IV - Estrutura do sub-programa GRAFIC

## BLOCO 1

Antes de iniciar um programa utilizando o GKS é necessário saber-se antecipadamente quais os tipos de estações de trabalho ("workstations") deseja trabalhar, como por exemplo, "DISPLAY", "PRINTER", "WISS", etc., informando para o GKS, em uma subrotina os nomes, números e tipos de "workstations" que que serão utilizadas no programa.

Um manuseio errado em qualquer subrotina do GKS, como um valor de argumento não existente no nível de implementação do GKS utilizado, tentar trabalhar com uma "workstation" não inicializada ou não aberta, e assim como centenas de outros erros possíveis de ocorrer, são apontados em um arquivo aberto para utilização somente do GKS. O número da unidade lógica do arquivo de erros é passado ao GKS na forma de argumento para uma subrotina que o abre.

## BLOCO 2

Desde que trabalha-se com segmentos e deseja-se copiar estes para outro tipo de "workstation" é necessário além de uma "workstation" de entrada e saída como "DISPLAY", de uma outra em que são armazenados os segmentos e tenha potencial de envia-los para outras, como, por exemplo, uma do tipo "WISS". Desta forma usa-se uma subrotina para abrir a "workstation", inicializando o dispositivo, e atribuindo a esta um número que a identificará para uso em outras subrotinas. Além disso, é necessário ativar as "workstations" para, por exemplo exibir as saídas ou armazenar segmentos.

## BLOCO 3

Cria o segmento (10) para armazenar todas as informações que serão comentadas a seguir.

A definição da área de exibição na tela do vídeo é feita pelo traçado de quatro linhas pela "polyline", que fazem o contorno desta área.

Um sistema de eixos cartesianos colocado no canto superior esquerdo da tela, orientará o usuário na verificação de dados de entrada como momentos de inércia de massa do disco em relação os eixos X e Y. Da mesma que no arquivo de dados de entrada, o texto que serve de auxílio na identificação do arquivo, é impresso na parte superior da tela.

## BLOCO 4

Cria o segmento (20) para armazenar todas as informações que serão comentadas a seguir.

Faz uma transformação de coordenadas dos raios externos dos elementos de barra passando de coordenadas universais para coordenadas normalizadas do dispositivo.

## Sub-programa ELEM

Constrói os elementos de barra traçando uma "polyline" passando pelos seus quatro vértices.

## Sub-programa NUM

Terminada a construção de todos os elementos de barra do rotor, seus nós são numerados logo acima destes e, para evitar uma possível superposição no caso de haver um número de nós elevado, é estipulado um valor limite igual a quinze (15), a

partir do qual são colocados apenas os números ímpares dos nós.

#### BLOCO 5

A leitura da variável MANC, já citada anteriormente, define o tipo de mancal que será posicionado no nó.

##### Sub-programa LIVRE

Esta subrotina traça através da "polyline", um "segmento de mola", que tem a função de mostrar ao usuário que trata-se de um mancal flexível.

##### Sub-programa APOIO

Esta subrotina traça através da "polyline" um pequeno triângulo sob o nó, que tem a função de mostrar que trata-se de um apoio simples.

##### Sub-programa ENGASTE

Esta subrotina traça através da "polyline" três segmentos de reta formando um "colchete", um acima e outro abaixo do nó, caracterizando um engastamento.

#### BLOCO 6

Define o tamanho dos discos rígidos em coordenadas normalizadas do dispositivo através de uma combinação de quatro raios externos com três espessuras.

##### Sub-programa DISCO

Constrói os discos rígidos traçando uma "polyline" passando pelos seus quatro vértices.

## BLOCO 7

Fecha o segmento (20) contendo todas as informações logo após a sua criação.

## Sub-programa CHAVES

Em toda parte inferior da tela é construído um painel com todas as opções que o programa oferece para a verificação dos dados de entrada, e quais as teclas utilizadas para cada uma destas, que são:

- F1 - Elementos de barra
- F2 - Disco rígido
- F3 - Mancais
- F4 - Elementos adicionais
- F5 - Outras informações
- F6 - Continuar

## BLOCO 8

Se a escolha no menu for a visualização do rotor na impressora, a "workstation" "DISPLAY" aberta anteriormente é fechada, dando lugar à "workstation" "PRINTER" para onde os segmentos (10) e (20) são enviados. Após a impressão destes segmentos neste dispositivo, ele é fechado, dando lugar novamente ao "DISPLAY", para onde os segmentos são retornados.

## ESCOLHA DE UMA CHAVE ENTRE AS OPÇÕES:

No caso de uma chave diferente daquelas especificadas como opções for teclada, uma mensagem será exibida na tela informando do erro. Para os demais casos, tem-se:

### F1 - Sub-programa ELEMBAR

Cria-se um segmento que é exibido na tela, constando da pergunta sobre o número do elemento de barra desejado. Logo que escolhido, o segmento é deletado da tela e um novo segmento é criado, exibindo suas características geométricas e as características do material que o constitui. Os valores exibidos são: número do elemento, área da seção transversal e momentos de inércia ou raio externo e interno, módulo de Young, massa volumétrica e coeficiente de Poisson e, além destes, a chave F1 para continuar, que ao ser teclada faz com que este segmento seja deletado, e volta-se ao ambiente da tela principal, podendo-se fazer uma nova escolha através de qualquer chave.

### F2 - Sub-programa DISCRIG

Cria-se um segmento que é exibido na tela constando da pergunta sobre o número do disco desejado. Logo que escolhido, o segmento é deletado da tela e um novo segmento é criado, exibindo suas características geométricas e as características do material que o constitui. Os valores exibidos são: número do disco, massa, momentos de inércia de massa em relação aos eixos X e Y, ou raio externo, raio interno e espessura, módulo de Young, massa volumétrica e coeficiente de Poisson e, além destes, a chave F1 para continuar.

### F3 - Sub-programa MANCAIS

Cria-se um segmento que é exibido na tela constando da pergunta sobre o número do mancal desejado. Logo que escolhido, o segmento é deletado da tela e um novo segmento é criado

exibindo as características de rigidez e de amortecimento do mancal, além da chave F1 para continuar.

#### F4 - Sub-programa ELEMADI

Na ausência de qualquer elemento adicional, é criado um segmento com uma mensagem informando tal inexistência, juntamente com a chave F1 para continuar. Mas, na presença de qualquer um dos elementos adicionais, é criado um segmento para cada um, onde são exibidos as rigidez, massas, termos de coriolis e amortecimentos sem voltar ao ambiente da tela principal.

No segmento contendo a informação de um dado elemento são exibidos o tipo de elemento adicional, em qual ou quais nós está posicionado, e o seu valor, além da chave F1 para continuar a exibição de outros elementos adicionais ou, se não houver mais nenhum, voltar ao ambiente da tela principal.

#### F5 - Sub-programa OUTINF

No caso de não haver forças externas, massa de desbalanceamento ou força assíncrona, cria-se um segmento com uma mensagem informando que não há mais informações juntamente com a chave F1 para voltar ao ambiente da tela principal. Mas, na presença de uma dessas informações, um segmento é criado para cada uma delas e exibido um a um sem voltar ao ambiente da tela principal. As informações contidas no segmento são: para a força externa, número do nó onde está aplicada suas projeções ao longo de X e Z e a chave F1 para continuar; para massa de desbalanceamento, número do nó onde se acha o desbalanceamento, valor da massa excêntrica, distância desta ao

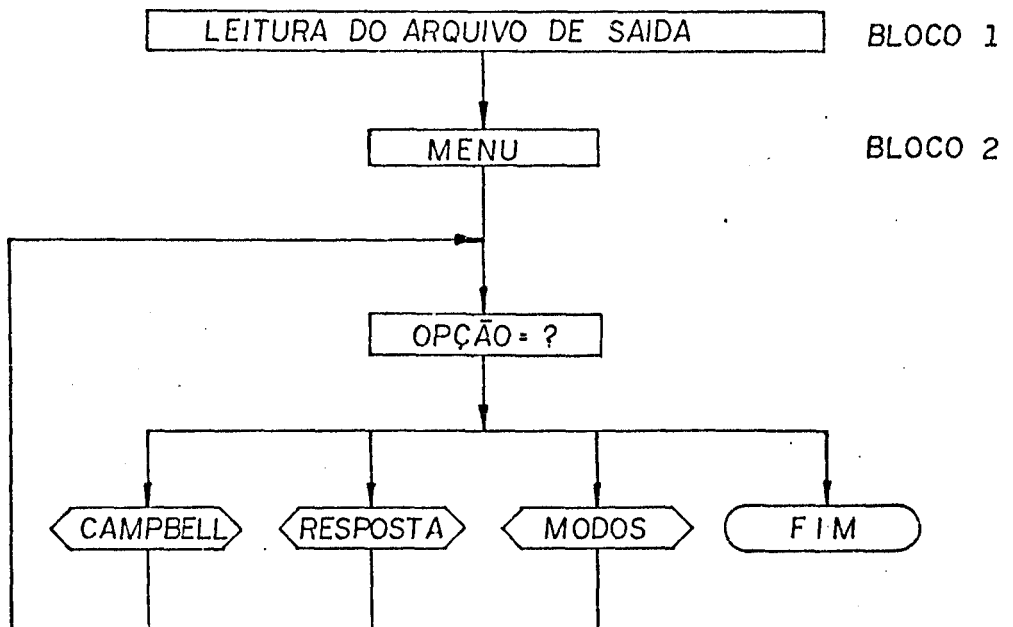
eixo e o ângulo de posição, e a chave F1 para continuar; para a força assíncrona, número do nó onde está aplicada, amplitude da força e o ângulo de posição, e a chave F1 para continuar.

#### F6 - Continuar

Esta chave faz com que os segmentos (10) e (20) sejam deletados da tela e as "workstations" "DISPLAY" e "WISS" desativadas e fechadas. Também são fechados o GKS e o seu arquivo de erros, retornando assim ao menu para a escolha de um novo arquivo, ou para encerrar o programa.

4.2.2. Estrutura do programa de pós-processamento do programa Rotor.

#### I - ORGANOGRAMA DO PROGRAMA PRINCIPAL



## II - Estrutura do programa principal

### BLOCO 1

Como dito anteriormente, quando da execução do programa Rotor, é criado um arquivo de saída, onde são gravados os valores correspondentes aos cartões tipos 2 a 8 e os valores correspondentes aos resultados do diagrama de Campbell, resposta do sistema, e modos próprios.

A razão para recolocar os cartões tipo 2 a tipo 8 neste arquivo é para que seja montado novamente o rotor e, através dos modos próprios, representar os modos com animação. Uma única mudança ocorre na cartão tipo 2, onde são gravados apenas os valores INDIC, NOEU, COD, NFREQ e KMOD, referentes a: escolha do tipo de cálculo desejado, número de nós, código de leitura das propriedades geométricas, número de frequências desejadas código para impressão dos modos e matrizes modais respectivamente.

### BLOCO 2

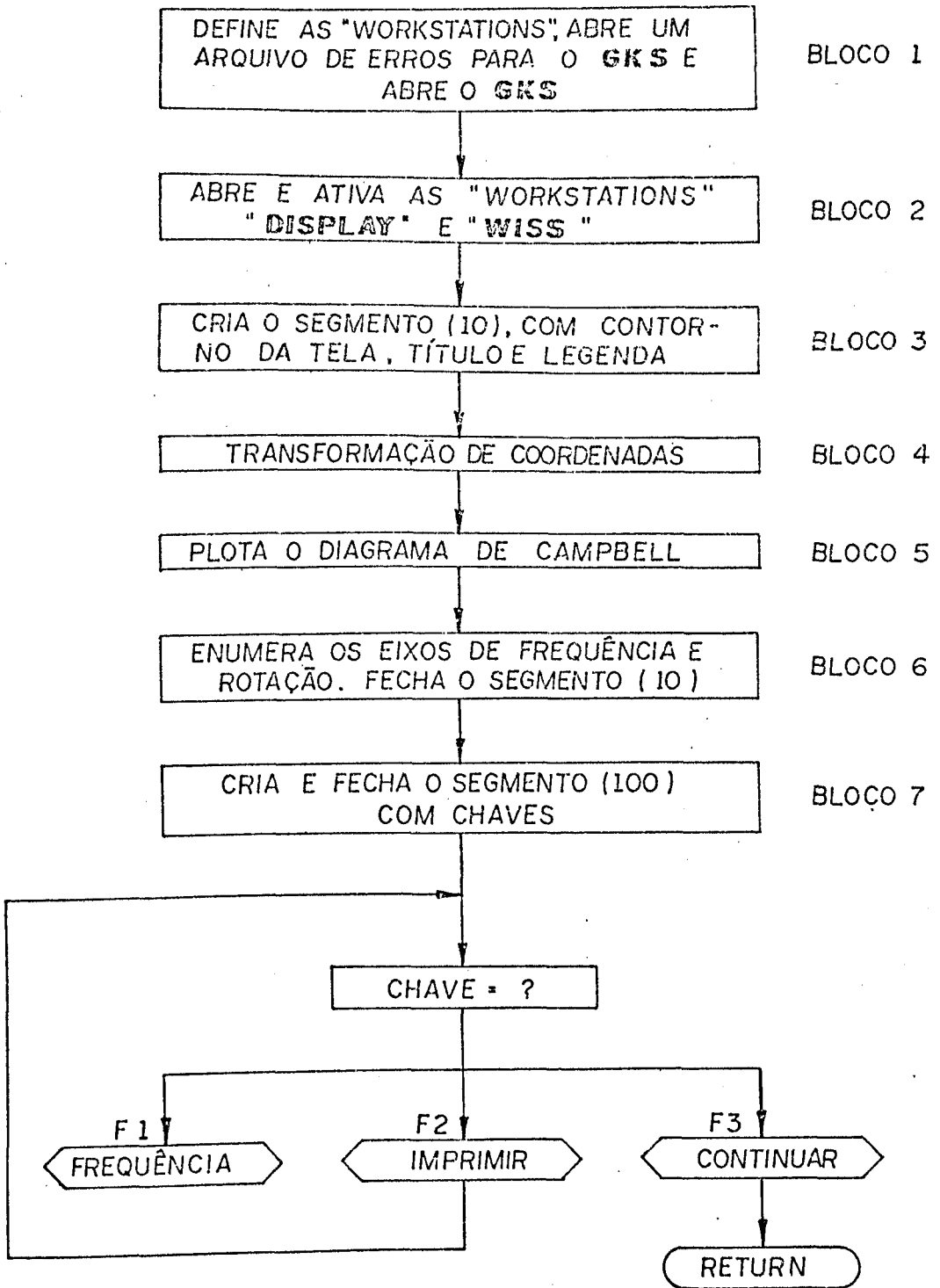
O menu expõe ao usuário as opções que o programa oferece:

- 1 - Diagrama de Campbell
- 2 - Resposta a excitações síncronas ou assíncronas
- 3 - Representação dos modos com animação
- 4 - Encerrar programa

As três primeiras opções acima podem enviar uma mensagem de erro, caso os valores escolhidos para as variáveis INDIC e KMOD não sejam capazes de representar tais opções.



III - ORGANOGRAMA DO SUB-PROGRAMA CAMPBELL



## IV - Estrutura do sub-programa CAMPBELL

## BLOCO 1

Idem ao BLOCO 1 do sub-programa GRAFIC

## BLOCO 2

Idem ao BLOCO 2 do sub-programa GRAFIC

## BLOCO 3

Cria-se um segmento (10) com contorno sobre a área de exibição na tela do vídeo e coloca-se o título do gráfico, "DIAGRAMA DE CAMPBELL", na parte superior da tela. A área de traçagem das curvas é também contornada, onde sob os eixos são colocados as legendas: [rpm] no eixo das abcissas e [Hz] no eixo das ordenadas.

## BLOCO 4

Faz-se uma transformação de coordenadas sobre os valores de rotação e frequência de coordenadas universais para coordenadas normalizadas do dispositivo e, sobre estes valores, faz-se uma interpolação utilizando o método de FORSYTHE [16].

## BLOCO 5

São traçadas as curvas representativas dos modos e, sobre o diagrama, traça-se a reta onde os valores de rotação são coincidentes com os de frequência ( $\Omega = \omega$ ), para facilitar a determinação das velocidades críticas.

## BLOCO 6

Sobre o eixo de rotação, são colocados cinco valores de rotação que, são os dois valores extremos da faixa escolhida para os resultados, além de três valores intermediários equidistantes e, sobre o eixo de frequência, são colocados também cinco valores sendo estes o valor zero, o primeiro inteiro abaixo da frequência máxima e mais três valores intermediários equidistantes. Fecha-se o segmento (10).

## BLOCO 7

Cria-se um segmento (100) somente com as opções sobre o gráfico do diagrama de Campbell, utilizando-se das seguintes chaves:

F1 - Frequência

F2 - Imprimir

F3 - Continuar

## ESCOLHA DE UMA CHAVE ENTRE AS OPÇÕES:

F1 - Frequência

Ao teclá-la, surge na tela um "crosshairs" (marcador de posição) que, através das teclas de movimento horizontal e vertical do cursor, movimenta-se para qualquer posição da tela.

Ao posicioná-lo sobre uma das curvas e teclar "enter", está é identificada, fazendo com que o "crosshairs" desapareça da tela, sendo reposicionado imediatamente na parte inferior esquerda da área especificada para a traçagem do gráfico. Feito isso, basta posicionar o "crosshairs" sobre a curva e posição

desejadas e teclar "enter". Assim obtém-se os valores da frequência e rotação para aquela posição especificada, que são exibidos na parte inferior esquerda do vídeo. Verificados os valores, estes são deletados após teclar-se "enter", voltando-se ao ambiente da tela principal.

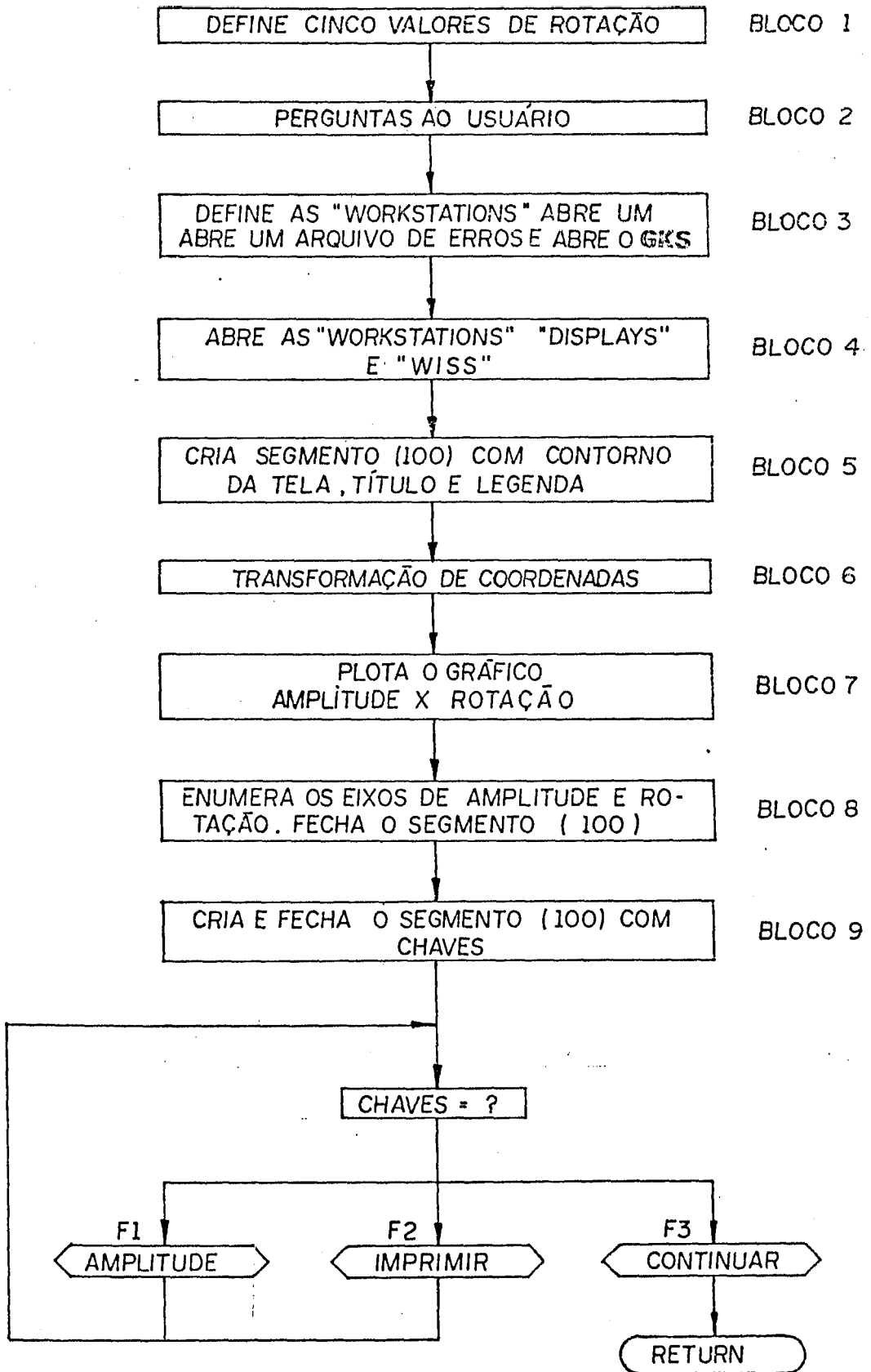
#### F2 - Imprimir

Ao teclá-la, a "workstation" "DISPLAY" é fechada dando lugar à "workstation" "PRINTER", para onde o segmento (10) é enviado e impresso. Após a impressão, a "workstation" "PRINTER" é fechada e a "DISPLAY" é novamente aberta, e o segmento (10) retornado.

#### F3 - Continuar

Esta tem a função de deletar todos os segmentos exibidos na tela, fechar e desativar as "workstations" "DISPLAY" e "WISS" e fechar o GKS e seu arquivo de erros.

## V - ORGANOGRAMA DO SUB-PROGRAMA RESPOSTA



## VI - Estrutura do sub-programa RESPOSTA

## BLOCO 1

Este define os cinco valores de rotação a serem exibidos na tela. Seus valores são os dois extremos da faixa utilizada para o cálculo da resposta, e os três restantes são valores intermediários equidistantes. Devido ao fato de os valores de amplitude serem plotados na escala logarítmica, o valor de amplitude zero para rotação zero rpm é indefinido. Define-se então 10 rpm como valor inicial da rotação, e um valor próximo de zero para a amplitude correspondente.

## BLOCO 2

Para a construção do gráfico Amplitude X Rotação são necessárias mais algumas informações que são introduzidas neste BLOCO, que são: número do nó onde se deseja observar a resposta e em que plano, X-Y ou Z-Y.

## BLOCO 3

Idem ao BLOCO 1 do sub-programa GRAFIC.

## BLOCO 4

Idem ao BLOCO 2 do sub-programa GRAFIC.

## BLOCO 5

Cria-se um segmento (10) com contorno da área de exibição da tela, título do gráfico: "RESPOSTA DO SISTEMA A UM DESBALANCEAMENTO" ou "RESPOSTA DO SISTEMA A UMA FORÇA ASSÍNCRONA", número do nó e plano correspondente à resposta que

está sendo exibida, todos posicionados na parte superior da tela. A área onde é exibido o gráfico é também contornada e, sobre os eixos das abcissas e ordenadas são posicionadas as legendas, para rotação [rpm] e, amplitude [m], respectivamente.

#### BLOCO 6

Calcula-se o logaritmo na base 10 dos valores de amplitude, e faz-se uma transformação de coordenadas universais para coordenadas normalizadas do dispositivo.

#### BLOCO 7

Sobre os pontos normalizados faz-se uma interpolação utilizando-se o método de FORSYTHE [16].

#### BLOCO 8

Sobre o eixo de rotação são colocados os cinco valores de rotação calculados anteriormente e, sobre o eixo das amplitudes são colocadas as ordens de grandeza em potencias de dez, a partir do valor mínimo, até o valor máximo encontrado. Fecha-se o segmento (10).

#### BLOCO 9

Cria-se um segmento (100) somente com as opções sobre o gráfico Amplitude X Rotação utilizando-se das seguintes chaves:

F1 - Amplitude

F2 - Imprimir

F3 - Continuar

## ESCOLHA DE UMA DAS CHAVES ENTRE AS OPÇÕES:

## F1 - Amplitude

Repete-se aqui o mesmo princípio aplicado no gráfico do diagrama de Campbell, ou seja, ao teclar-se F1 surge na tela um "crosshairs" que ao posiciona-lo sobre a curva e teclar "enter", esta será identificada e o "crosshairs" desaparece da tela reaparecendo novamente na parte inferior esquerda da área especificada para a traçagem do gráfico. Feito isto, basta posicionar o "crosshairs" sobre a curva na posição desejada e teclar "enter". Os valores de amplitude e rotação para aquela posição especificada tem seus valores exibidos na parte inferior esquerda do vídeo. Verificado os valores, estes são deletados ao teclar-se "enter", voltando-se ao ambiente da tela principal, com as opções aparecendo novamente no vídeo.

## F2 - Imprimir

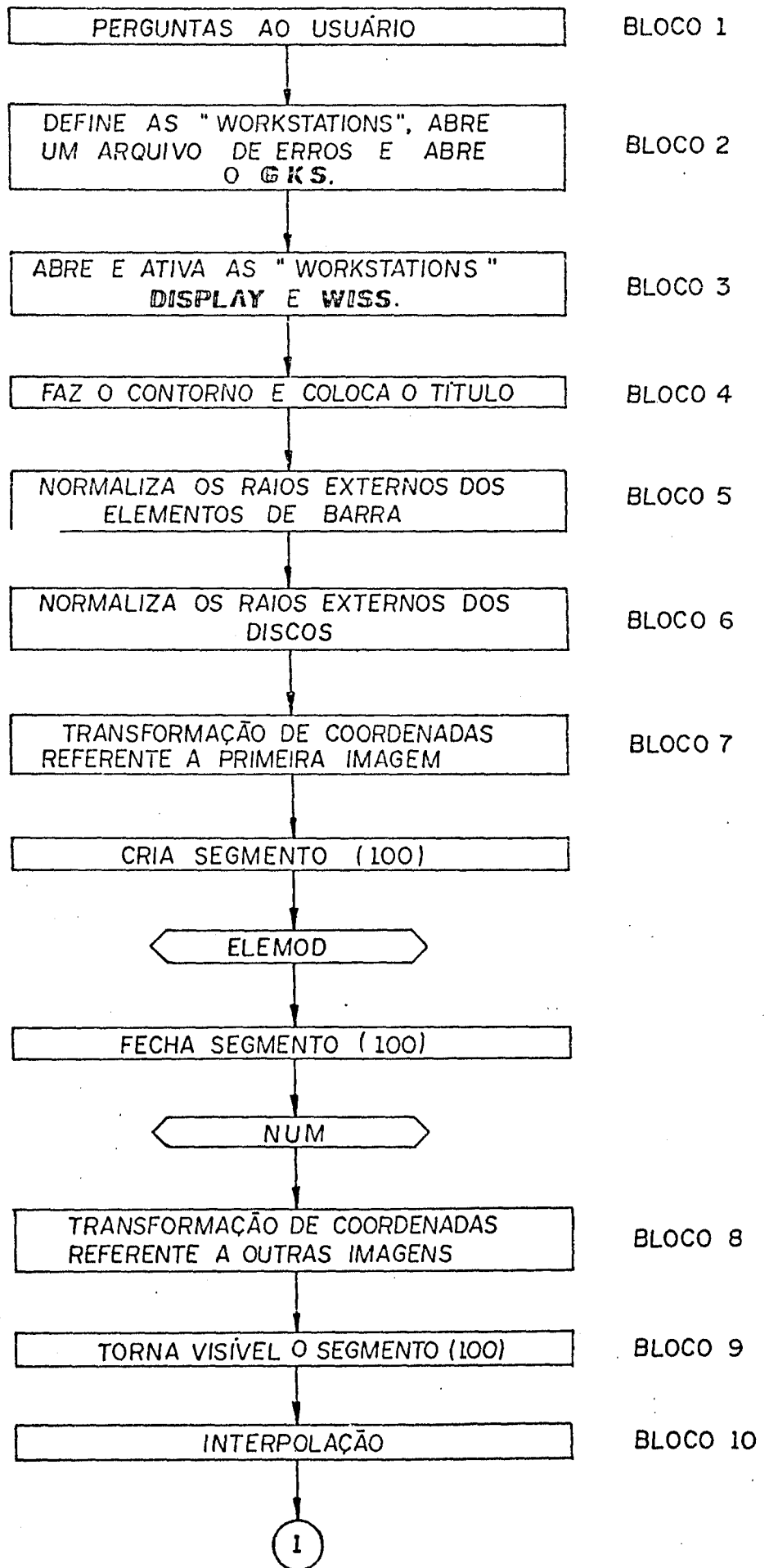
Idem a chave F2 do sub-programa CAMPBELL.

## F3 - Continuar

Idem a chave F3 do sub-programa CAMPBELL.

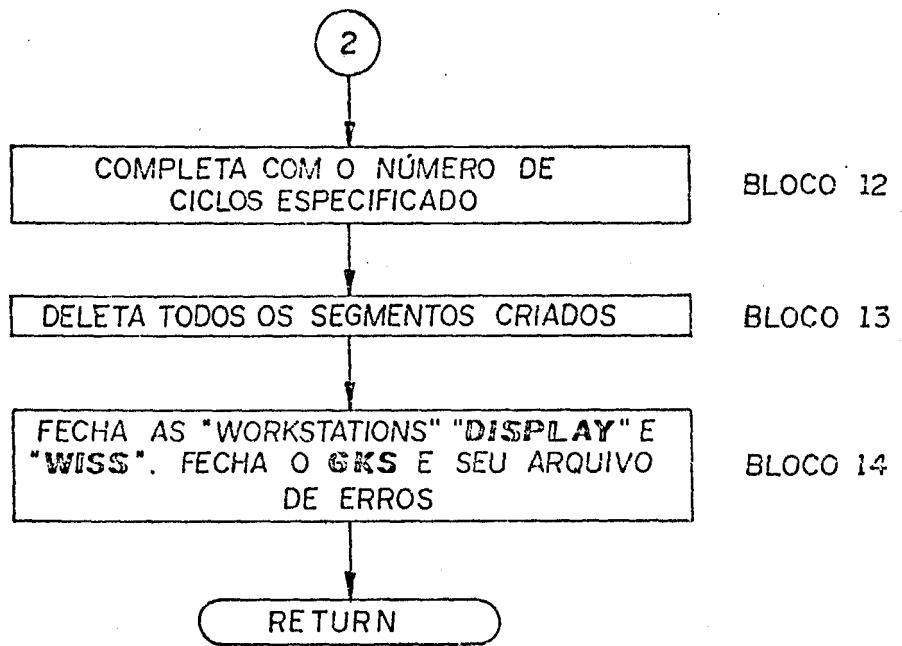


VII - ORGANOGRAMA DO SUB-PROGRAMA MODOS





BLOCO 11



## VIII - Estrutura do subprograma MODOS

## BLOCO 1

Para a representação animada dos modos são necessárias algumas informações que são perguntadas ao usuário, que são: número do modo desejado, plano de visualização dos modos ( X-Y ou Z-Y), rotação na qual será representado o modo, e um valor de equivalência para o tempo da animação, que pode ser um valor inteiro a partir de 1 (um). Este valor será o contador de um comando "DO" Fortran, que estipula o número de ciclos a ser exibido na tela.

## BLOCO 2

Por razões óbvias, somente as "workstations" "WISS" e "DISPLAY" são definidas para o GKS já que a animação pode ser feita somente no vídeo. Como já comentado anteriormente, deve ser aberto um arquivo de erros de utilização do GKS, cuja unidade é passada ao GKS na forma de argumento, para a subrotina que o abre.

## BLOCO 3

Idem ao BLOCO 2 do sub-programa GRAFIC.

## BLOCO 4

Faz-se o contorno da área de exibição da tela e na parte superior é colocado o título com as seguintes informações: número do modo, velocidade de rotação em rpm e o plano escolhido para a representação do modo.

## BLOCO 5

Idem ao BLOCO 4 do sub-programa GRAFIC, com exceção de que não é criado nenhum segmento.

## BLOCO 6

Idem ao BLOCO 6 do sub-programa GRAFIC.

## BLOCO 7

Faz-se uma transformação de coordenadas universais em coordenadas normalizadas do dispositivo da imagem do rotor suposto parado.

Sub-programa ELEM0D

(Ver pg. )

Sub-programa NUM

Idem ao sub-programa NUM do sub-programa GRAFIC.

## BLOCO 8

Monta quatro vetores referente às quatro imagens restantes e sobre eles faz-se uma transformação de coordenadas universais em coordenadas normalizadas do dispositivo.

## BLOCO 9

Como na tela ainda está sendo exibida uma imagem, para dar a noção de movimento, é necessário tornar esta imagem invisível (sem deletá-la) para que outras imagens possam ser exibidas.

## BLOCO 10

Sobre os valores do segundo vetor montado no BLOCO 8, é feita uma interpolação.

## BLOCO 11

A seguir, todas as operações feitas seguem o mesmo procedimento usado na criação do segmento (100), BLOCO 9 e BLOCO 10, isto é, à medida que um segmento (imagem) é criado, ele é feito invisível na tela para dar lugar à criação de outro segmento (imagem), e assim sucessivamente até que sejam criados todos os segmentos (as 5 imagens), e exibidos numa sequência lógica.

## BLOCO 12

Após a criação dos 5 segmentos, estes são apenas feitos visíveis e invisíveis de maneira a dar impressão do movimento vibratório do rotor, de acordo com o modo em estudo. O número de ciclos é aquele especificado anteriormente.

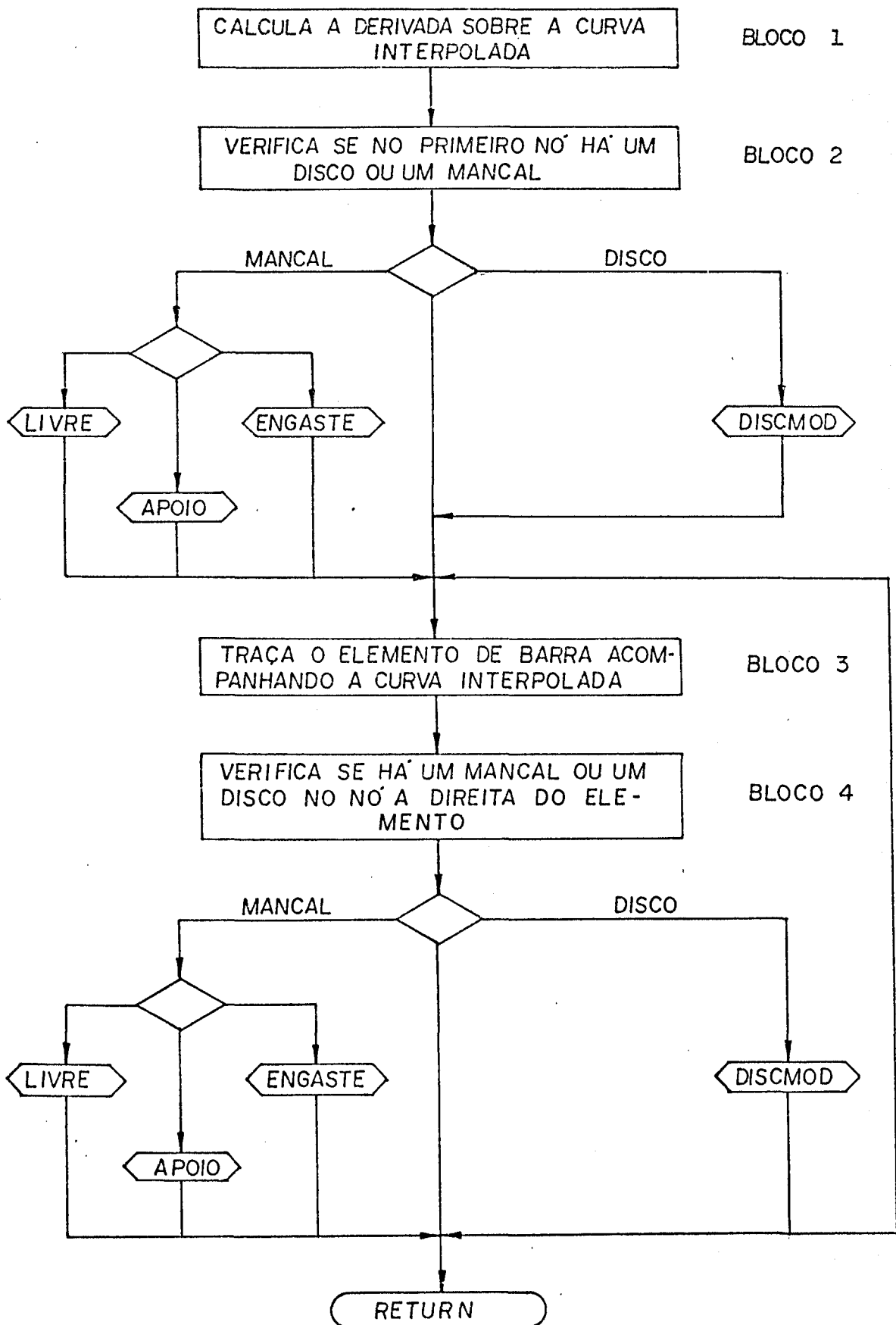
## BLOCO 13

Encerrada a exibição das imagens representativas do modo escolhido, estas são deletadas da "workstation".

## BLOCO 14

Fecha-se e desativa-se as "workstations" "DISPLAY" e "WISS" fecha-se o GKS e o seu arquivo de erros.

IX - ORGANOGRAMA DE SUB-PROGRAMA ELEMCD



## X - Estrutura do sub-programa ELEMODO

## BLOCO 1

Sobre a curva interpolada, é calculada a derivada de uma maneira muito simples, já que esta é um polinômio. Assim, calcula-se a inclinação em todo ponto da curva, cujo valor é dado pelo arco tangente.

## BLOCO 2

Se no primeiro nó há um mancal ou um disco, é calculada a inclinação neste nó e enviada aos seguintes sub-programas: LIVRE, APOIO ou ENGASTE, ou DISCMOD.

## Sub-programa LIVRE

Na posição deste mancal será traçada uma "mola" que de acordo com a posição do nó pode se alongar ou comprimir, dependendo do modo.

## Sub-programa APOIO

Na posição deste mancal será traçado um "triângulo" sob o nó. Devido a possíveis desvios na curva interpolada, o mancal "seguirá" a posição do nó.

## Sub-programa ENGASTE

Na posição deste mancal serão traçados dois "colchetes", um acima e outro abaixo do nó e, também devido a possíveis desvios na curva interpolada, o mancal "seguirá" a posição do nó como também poderá haver uma pequena inclinação da seção do elemento no nó.



### Sub-programa DISCMOD

O traçado do disco é feito da mesma forma que no sub-programa DISCO do sub-programa GRAFIC, com exceção de que, neste, ele é traçado de acordo com a inclinação dada pelo nó.

### BLOCO 3

Cada elemento de barra é dividido em cinco partes iguais, tendo portanto seis seções com inclinações diferentes uma da outra, de forma a acompanhar a deformação do rotor.

### BLOCO 4

Terminada a construção do elemento de barra, é verificado se no nó à direita há um mancal ou um disco. Se houver, a inclinação calculada para a sexta seção do elemento de barra será a mesma a ser utilizada pelos sub-programas LIVRE, APOIO ou ENGASTE, ou DISCMOD.

## 5. APLICAÇÕES

### 5.1. Exemplo didático.

Para fins de demonstração das potencialidades de aplicação do trabalho realizado, serão apresentados resultados relativos a um rotor modelado com 7 nós e 2 discos rígidos, suportado por 2 mancais, sendo um rígido e outro flexível, nas extremidades esquerda e direita respectivamente. O modelo elementos finitos foi o adotado abaixo.

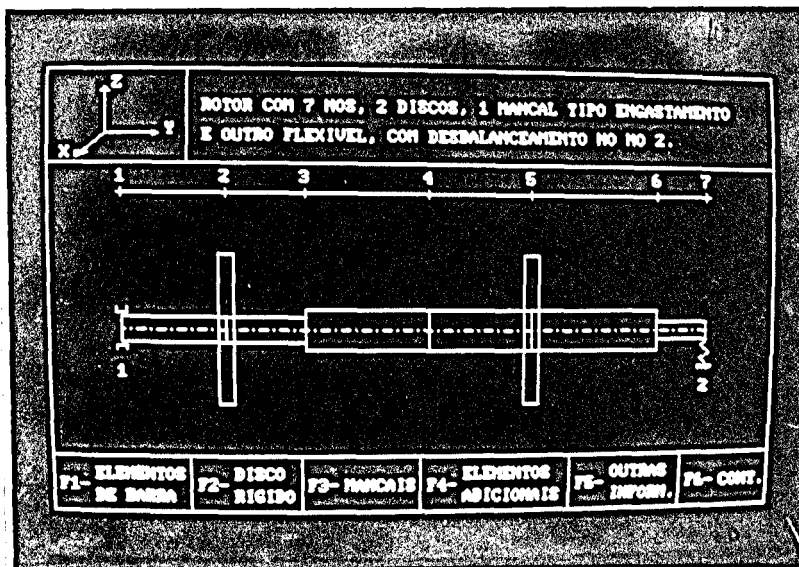


fig. 5.1.1 - Modelo elementos finitos para o exemplo de aplicação didática

Os dados são os seguintes:

Para os elementos de barra.

elementos 1 e 2:

- raio da seção transversal = 0.004 m

elementos 3, 4 e 5:

- raio da seção transversal = 0.006 m

elemento 6:

- raio da seção transversal = 0.003 m

material dos elementos de barra:

-  $E = 2.0 \times E 11 \text{ N/m}^2$

-  $\rho = 7800 \text{ Kg/m}^3$

-  $\nu = 0.3$

Para os discos rígidos.

discos 1 e 2:

- massa = 7.85 Kg

- momento de inércia de massa/XX = 0.122 Kg m<sup>2</sup>

- momento de inércia de massa/YY = 0.245 Kg m<sup>2</sup>

material dos discos rígidos:

-  $E = 2.0 \times E 10 \text{ N/m}^2$

-  $\rho = 8000 \text{ Kg/m}^3$

-  $\nu = 0.30$

Para os mancais.

mancal 1:

- rigidez ao longo do eixo X ( $K_{xx}$ ) = 0.37 x E 9 N/m

- rigidez ao longo do eixo Z ( $K_{zz}$ ) = 0.74 x E 9 N/m

- rigidez cruzada ( $K_{xz}$ ) = 0.0 N/m

- rigidez cruzada ( $K_{zx}$ ) = 0.0 N/m

- Os coeficientes de amortecimento são nulo

mancal 2:

- rigidez ao longo do eixo X ( $K_{xx}$ ) = 0.225 x E 6 N/m

- rigidez ao longo do eixo Z ( $K_{zz}$ ) = 0.450 x E 6 N/m

- rigidez cruzada ( $K_{xz}$ ) = 0.0 N/m

- rigidez cruzada ( $K_{zx}$ ) = 0.0 N/m

- Os coeficientes de amortecimento são nulos

A seguir são apresentadas algumas janelas contendo as informações referentes às características dos elementos constituintes da máquina rotativa em estudo.

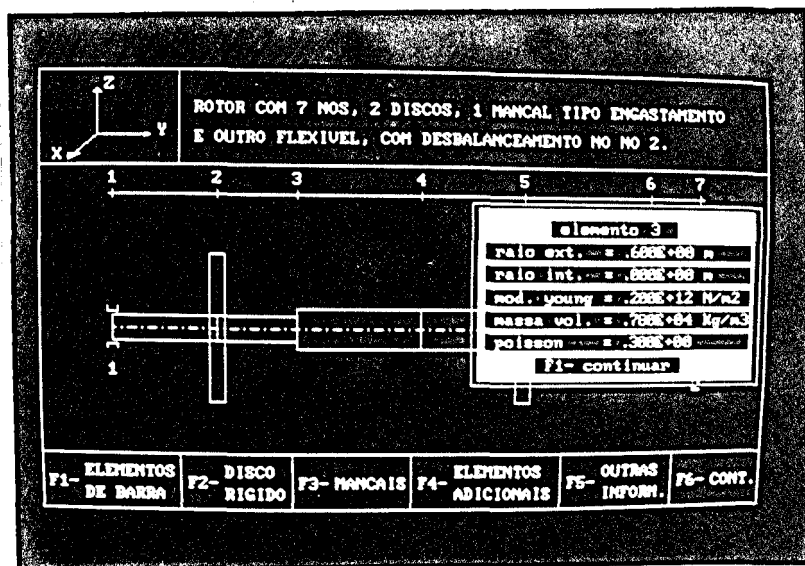


fig. 5.1.2 - Características de um elemento de barra

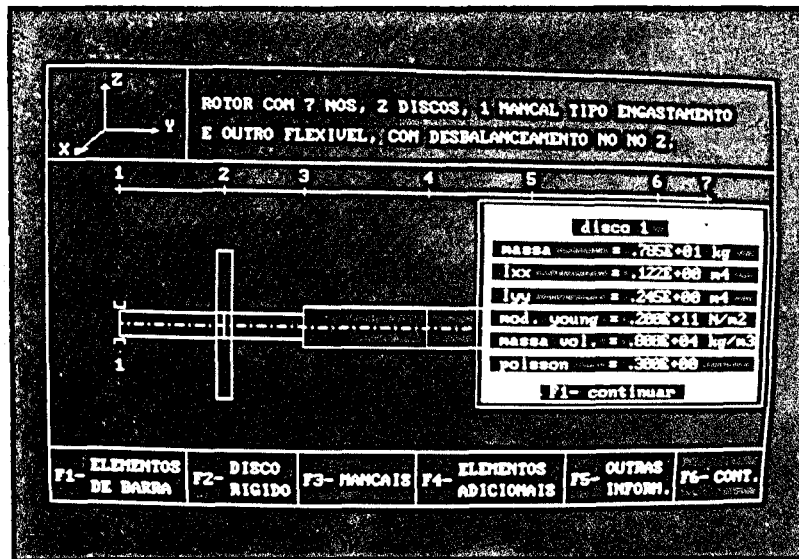


fig. 5.1.3 - Características de um disco rígido

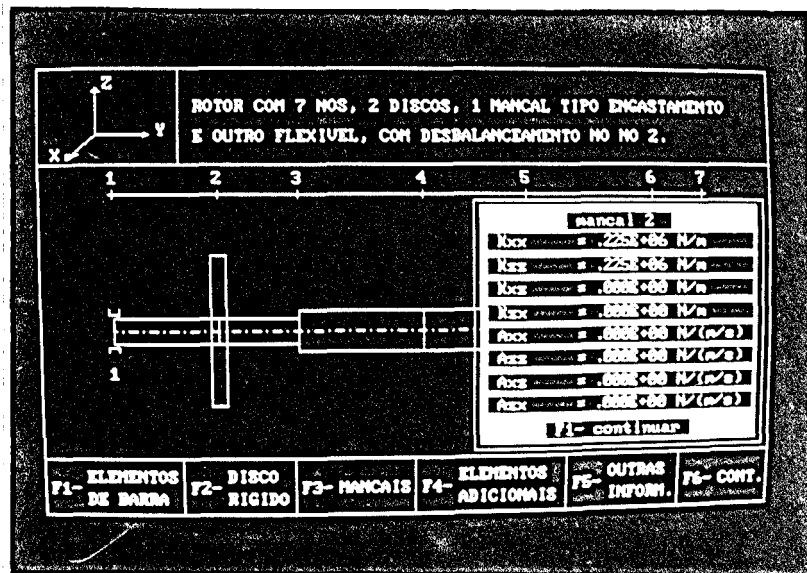


fig. 5.1.4 - Características de um mancal

Foram realizados os seguintes cálculos:

Dinâmica do sistema não-giroscópico e dinâmica do sistema giroscópico, além da resposta do sistema a um desbalanceamento.

Para o cálculo das frequências e modos foi utilizada uma faixa de  $\Omega = 0$  a 50000 rpm com passo de 1000 rpm.

Foi colocado um desbalanceamento no nó 2 sendo a massa excêntrica igual a 0.04 kg e a excentricidade de 0.05 m. A resposta é calculada dentro de uma faixa variando de  $\Omega = 0$  a 5000 rpm, com passo de 200 rpm.

Os resultados obtidos a partir do programa de pós-processamento do programa Rotor são os seguintes:

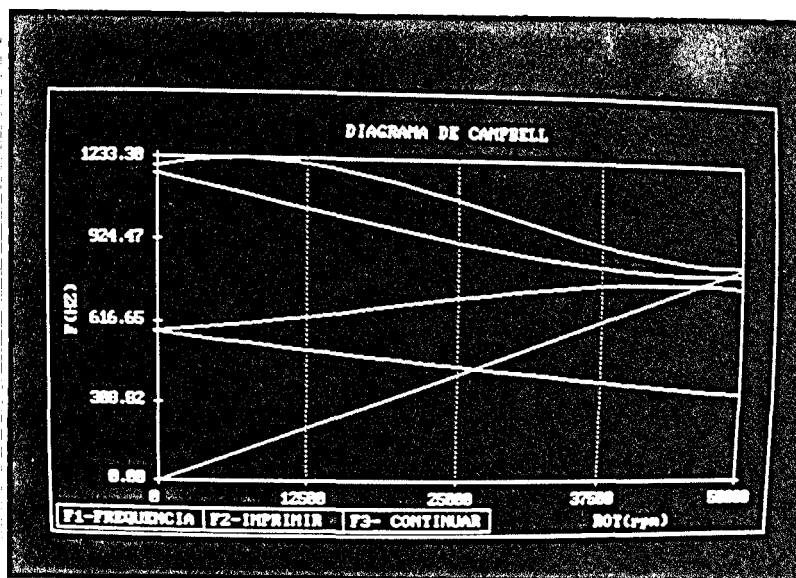


fig. 5.1.5 - Diagrama de Campbell

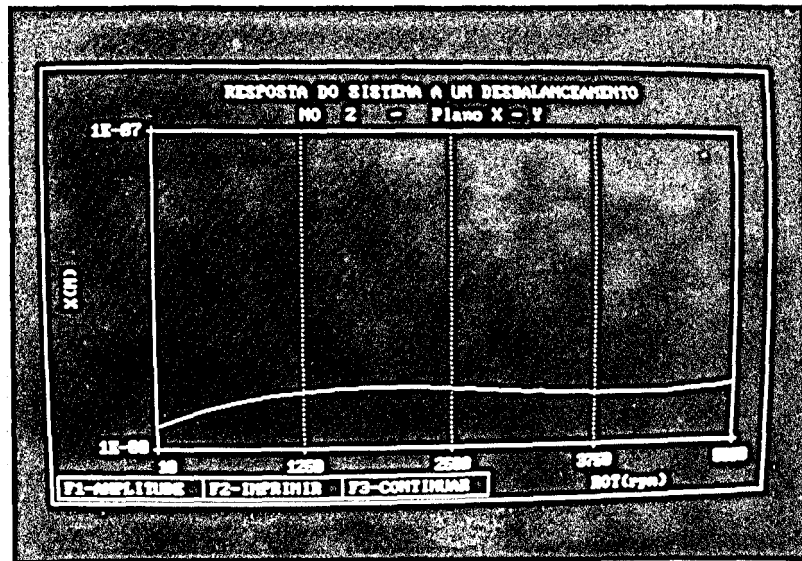


fig 5.1.6 - Resposta do sistema a um desbalanceamento

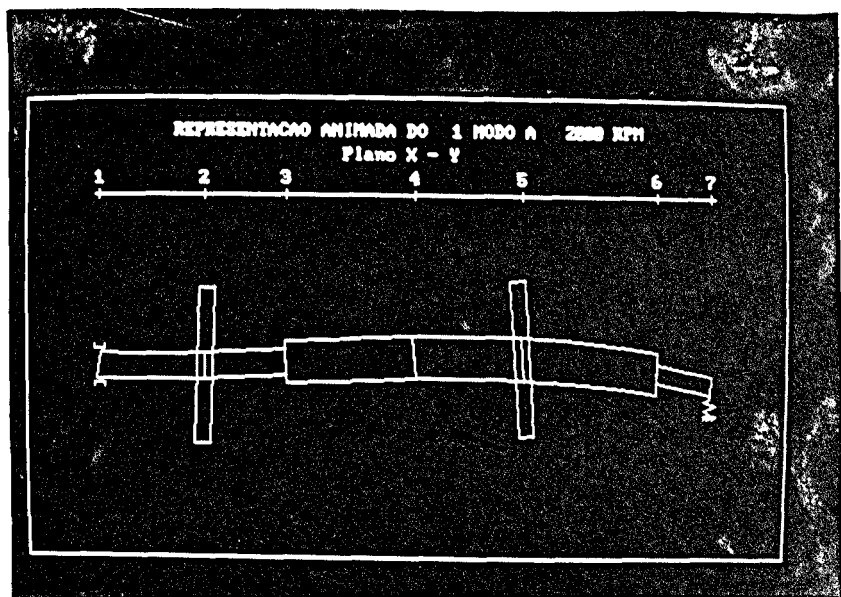


fig. 5.1.7 - Representação de uma imagem referente à animação do 1 modo a 2000 rpm

### 5.2. Exemplo de aplicação industrial.

O exemplo abaixo corresponde a uma aplicação real. A figura 5.2.1 mostra o modelo e a discretização por elementos finitos.

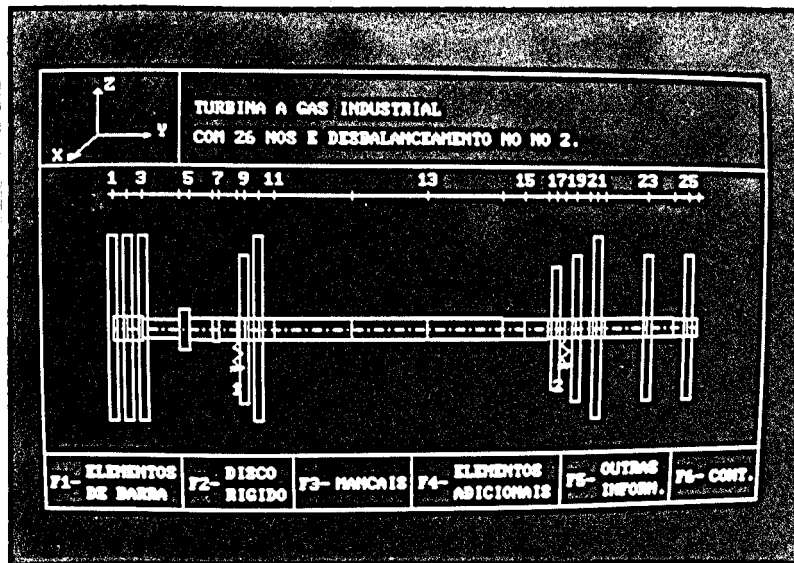


fig.5.2.1. Modelo elementos finitos para o exemplo de aplicação industrial

Os resultados obtidos para este exemplo de aplicação real foram:



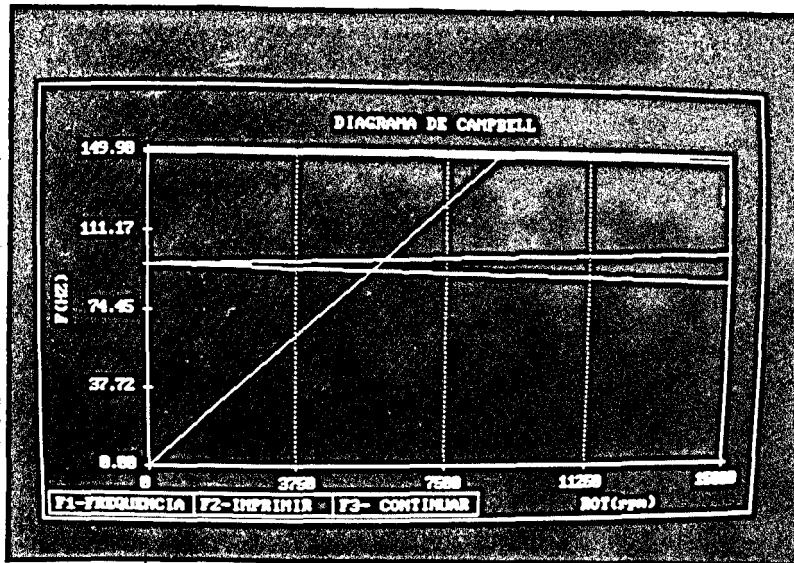


fig. 5.2.2. Diagrama de Campbell

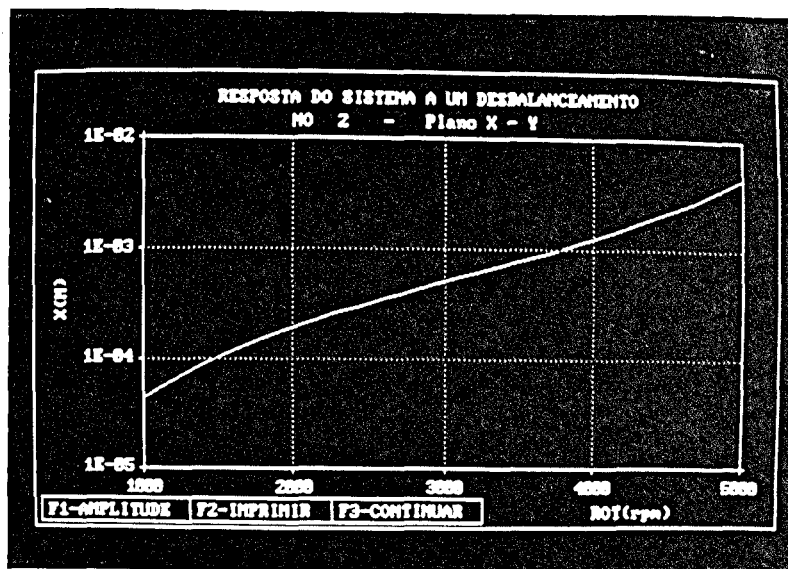


fig. 5.2.3. Resposta do sistema a um desbalanceamento colocado sobre o disco no nó 2

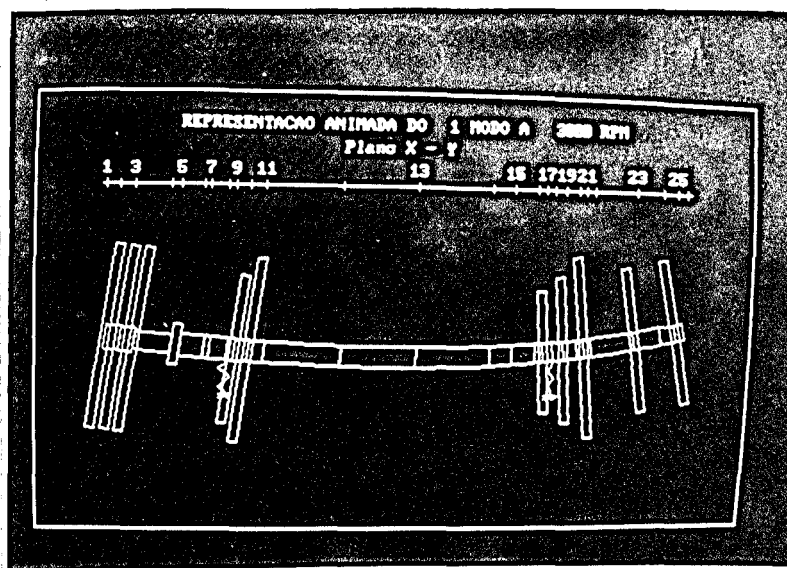


fig. 5.2.4. Representação de uma imagem referente à animação do 1 modo a 3000 rpm

## 6. CONCLUSÃO

O programa Rotor tem sido amplamente utilizado nas mais diferentes aplicações. São estas aplicações que permitem compreender a importância destas rotinas gráficas de auxílio ao programa Rotor, facilitando grandemente o trabalho do projetista.

A possibilidade de conferência do arquivo de dados de entrada do programa Rotor pela observação do modelo da máquina rotativa no vídeo, e a posterior visualização gráfica dos resultados da análise dinâmica, provocam a interação do usuário com o programa de análise, melhorando sem dúvida alguma a qualidade do trabalho executado.

Acredita-se que a escolha do GKS foi apropriada, devido à diversidade de funções que coloca à disposição do usuário e à sua portabilidade, podendo os programas migrar para diferentes instalações computacionais.

Os programas criados são compatíveis para qualquer microcomputador do tipo IBM-PC, porém, para uma representação mais nítida da imagem e melhor qualidade na animação dos modos próprios, são necessários vídeos de alta resolução e processadores de alta frequência, ou então utilizar estações de trabalho que são específicas para esta natureza de processamento computacional. Resultados bastantes satisfatórios foram obtidos em um monitor gráfico com 640 x 350 pontos, instalado em um microcomputador tipo PC-AT com velocidade de processamento de 12 MHz.

Todas as curvas são traçadas utilizando o mesmo método de interpolação ( FORSYTHE ), sendo que todos os pontos calculados

são usados para a obtenção da curva interpolada.

Um próximo passo a ser dado, seria a utilização das rotinas gráficas criadas neste trabalho, para equipar o processo de otimização dinâmica do projeto de máquinas rotativas, permitindo a observação, passo a passo, das modificações feitas, a partir de uma dada configuração inicial, na busca da solução ótima, obedecidas as várias restrições impostas.

## 7. REFERÊNCIAS BIBLIOGRÁFICAS

1. Ruhl, R. L., A Finite Element Model for Distributed Parameter Turborotor Systems, Journal of Engineering for Industry, Vol. 94, p. 126, Feb. 1972.
2. Steffen Jr., V., Estudo Analítico e Experimental de Dinâmica de Rotores, Rev. Bras. de Ciências Mec., 3 (3): 3 - 8, Set. 1981.
3. Hashish, E., and Sankar, T. S., Finite Element and Modal Analyses of Rotor-Bearing Systems Under Stochastic Loading Conditions, Journal of Vibration, Acoustics, Stress, and Reliability in Design, Vol 106, pp. 80-89, Jan. 1984.
4. Thomas, J., and Abbas, B. A., Finite Element Model for Dynamic Analysis of Timoshenko Beam, Journal of Sound and Vibration, 41 (3): 291 - 299, Jan. 1975.
5. Dawe, D. J., A Finite Element for the Vibration Analysis of Timoshenko Beams, Journal of Sound and Vibration, 60 (1): 11 - 20, Feb. 1978.
6. Nelson, H. D., and Mac Vaugh, The Dynamic of Rotor-Bearing Systems Using Finite Element, Journal of Engineering for Industry, Vol. 98, pp 593-600, May 1976.
7. Rossi, M. A., and Squarzoni, A. D. Finite Element Modal Approach to Large Rotor-Bearing System Analysis, Institute of Applied Mechanics, University of Genoa, Italy
8. Berthier, P., Ferraris, G., Lalanne, M., Rapport Technique Relatif au Code de Calcul Rotor, Laboratoire de Mecanique des Structures de L'INSA de Lyon, France, Juin 1984.
9. Steffen Jr, V. and Marcelin, J. L., Dynamic Optimization of Rotors, COBEM 1987, Florianópolis - SC.

10. Steffen Jr., V. and Marcelin, J. L., On the Optimization of Vibration Frequencies of Rotors, The International journal of Analytical and Experimental Modal Analysis, USA, July 1988.
11. Berthier, P., Ferraris, G., Lalanne, M., Prediction of Critical Speeds, Unbalance and Nonsynchronous Forced response of rotors, The shock and Vibration bulletin, May 1983.
12. Persiano, R. C. M., Oliveira, A. A. F., Introdução à computação gráfica, Rio de Janeiro, LTC editora ltda, 1986, 225 p.
13. IBM Personal Computer Graphical Kernel System, Language Bindings.
14. IBM Personal Computer Graphical Kernel System, Programmer's Guide.
15. Hopgood, F. R. A., Duce, D. A., Gallop, J. R., Sutcliffe, D. C., Introduction to the Graphical Kernel System, Academic Press, Inc., 1983.
16. Kelly, L. G., Handbook of numerical methods and applications, Addison-Wesley Publishing Company, 1967, 354 p.

## 8. ANEXOS

### 8.1. Anexo A - Matrizes massa e Coriolis do disco.

$$[M] = \begin{bmatrix} M_d & 0 & 0 & 0 \\ 0 & M_d & 0 & 0 \\ 0 & 0 & I_{dx} & 0 \\ 0 & 0 & 0 & I_{dx} \end{bmatrix}$$

$$[C] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -I_{dy} \\ 0 & 0 & I_{dy} & 0 \end{bmatrix}$$

8.2. Anexo B - Matrizes massa e Coriolis de um elemento de barra.

$$[M] = \left| \frac{\rho L}{840} \right| \begin{array}{cccccccc} u_1 & w_1 & \theta_1 & \psi_1 & u_2 & w_2 & \theta_2 & \psi_2 \\ \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} & m_{17} & m_{18} \\ & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} & m_{27} & m_{28} \\ & & m_{33} & m_{34} & m_{35} & m_{36} & m_{37} & m_{38} \\ & & & m_{44} & m_{45} & m_{46} & m_{47} & m_{48} \\ & & & & m_{55} & m_{56} & m_{57} & m_{58} \\ & & \text{simétrica} & & & m_{66} & m_{67} & m_{68} \\ & & & & & & m_{77} & m_{78} \\ & & & & & & & m_{88} \end{bmatrix} \end{array}$$

onde:

$$m_{11} = 240 S_1 + 72 S_2 + \frac{504}{L^2} (I_1 + I_2)$$

$$m_{12} = m_{13} = m_{16} = m_{17} = 0$$

$$m_{14} = - m_{23}$$

$$m_{15} = m_{26}$$

$$m_{18} = - m_{27}$$

$$m_{22} = m_{11}$$

$$m_{23} = L (30 S_1 + 14 S_2) + \frac{84}{L} I_2$$

$$m_{24} = m_{25} = m_{28} = 0$$

$$m_{26} = 54 (S_1 + S_2) - \frac{504}{L^2} (I_1 + I_2)$$

$$m_{27} = - L (14 S_1 + 12 S_2) + \frac{84}{L} I_1$$

$$m_{33} = L^2 (5 S_1 + 3 S_2) + 84 I_1 + 28 I_2$$

$$m_{34} = m_{35} = m_{38} = 0$$

$$m_{36} = L (12 S_1 + 14 S_2) - \frac{84}{L} I_2$$

$$m_{37} = - 3 L^2 (S_1 + S_2) - 14 (I_1 + I_2)$$

$$m_{44} = m_{33}$$

$$m_{45} = - m_{36}$$

$$m_{46} = m_{47} = 0$$

$$m_{48} = m_{37}$$

$$m_{55} = 72 S_1 + 240 S_2 - \frac{504}{L^2} (I_1 + I_2)$$

$$m_{56} = m_{57} = 0$$

$$m_{58} = - m_{67}$$

$$m_{66} = m_{55}$$



$$m_{67} = -L (14 S_1 + 30 S_2) - \frac{84}{L} I_1$$

$$m_{68} = 0$$

$$m_{77} = -L^2 (3 S_1 + 5 S_2) + 28 I_1 + 84 I_2$$

$$m_{78} = 0$$

$$m_{88} = m_{77}$$

$$[C] = \left| \frac{\rho \Omega}{30} \right| \begin{array}{cccccccc} u_1 & w_1 & \theta_1 & \psi_1 & u_2 & w_2 & \theta_2 & \psi_2 \\ \left[ \begin{array}{cccccccc} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} & c_{17} & c_{18} \\ & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} & c_{27} & c_{28} \\ & & c_{33} & c_{34} & c_{35} & c_{36} & c_{37} & c_{38} \\ & & & c_{44} & c_{45} & c_{46} & c_{47} & c_{48} \\ & & & & c_{55} & c_{56} & c_{57} & c_{58} \\ & & & & & \text{anti-simétrica} & c_{66} & c_{67} & c_{67} \\ & & & & & & & c_{77} & c_{78} \\ & & & & & & & & c_{88} \end{array} \right. \end{array}$$

onde:

$$c_{11} = c_{14} = c_{15} = c_{18} = 0$$

$$c_{12} = \frac{36}{L} (I_1 + I_2)$$

$$c_{13} = 6 I_2$$

$$c_{16} = -c_{12}$$

$$c_{17} = c_{13}$$

$$c_{22} = c_{23} = c_{26} = c_{27} = 0$$

$$c_{24} = c_{13}$$

$$c_{25} = c_{12}$$

$$c_{28} = c_{13}$$

$$c_{33} = c_{36} = c_{37} = 0$$

$$c_{34} = L (6 I_1 + 2 I_2)$$

$$c_{35} = c_{13}$$

$$c_{38} = - L (I_1 + I_2)$$

$$c_{44} = c_{45} = c_{48} = 0$$

$$c_{46} = c_{13}$$

$$c_{47} = - c_{38}$$

$$c_{55} = c_{58} = 0$$

$$c_{56} = c_{12}$$

$$c_{57} = - c_{13}$$

$$c_{66} = c_{67} = 0$$

$$c_{68} = - c_{13}$$

$$c_{77} = 0$$

$$c_{78} = L (2 I_1 + 6 I_2)$$

$$c_{88} = 0$$

8.3. Anexo C - Matrizes rigidez elástica (com efeito de cisalhamento) e rigidez geométrica de um elemento de barra.

$$\begin{array}{c}
 \begin{array}{cccccccc}
 u_1 & w_1 & \theta_1 & \psi_1 & u_2 & w_2 & \theta_2 & \psi_2
 \end{array} \\
 \left[ K_e \right] = \left| \frac{E}{L^3(1+\alpha)} \right| \begin{array}{c}
 \left[ \begin{array}{cccccccc}
 k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} & k_{17} & k_{18} \\
 & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} & k_{27} & k_{28} \\
 & & k_{33} & k_{34} & k_{35} & k_{36} & k_{37} & k_{38} \\
 & & & k_{44} & k_{45} & k_{46} & k_{47} & k_{48} \\
 & & & & k_{55} & k_{56} & k_{57} & k_{58} \\
 \text{simétrica} & & & & & k_{66} & k_{67} & k_{68} \\
 & & & & & & k_{77} & k_{78} \\
 & & & & & & & k_{88}
 \end{array} \right]
 \end{array}
 \end{array}$$

onde:

$$k_{11} = 12 I$$

$$k_{12} = k_{13} = k_{16} = k_{17} = 0$$

$$k_{14} = -6 I L$$

$$k_{15} = -k_{11}$$

$$k_{18} = k_{14}$$

$$k_{22} = k_{11}$$

$$k_{23} = -k_{14}$$

$$k_{24} = k_{25} = k_{28} = 0$$

$$k_{26} = -k_{11}$$

$$k_{27} = -k_{14}$$

$$k_{33} = 4 I L^2 \left(1 + \frac{\alpha}{4}\right)$$

$$k_{34} = k_{35} = k_{38} = 0$$

$$k_{36} = k_{14}$$

$$k_{37} = 2 I L^2 \left(1 - \frac{\alpha}{2}\right)$$

$$k_{44} = k_{33}$$

$$k_{45} = -k_{14}$$

$$k_{46} = k_{47} = 0$$

$$k_{48} = k_{37}$$

$$k_{55} = k_{11}$$

$$k_{56} = k_{57} = 0$$

$$k_{58} = -k_{14}$$

$$k_{66} = k_{11}$$

$$k_{67} = k_{14}$$

$$k_{68} = 0$$

$$k_{77} = k_{33}$$

$$k_{78} = 0$$

$$k_{88} = k_{33}$$

$$\alpha = \frac{12 E I}{G \beta L^2}$$

$$[ K_g ] = \left| \frac{F}{30 L} \right| \begin{array}{cccccccc} u_1 & w_1 & \theta_1 & \psi_1 & u_2 & w_2 & \theta_2 & \psi_2 \\ \left[ \begin{array}{cccccccc} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} & k_{17} & k_{18} \\ & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} & k_{27} & k_{28} \\ & & k_{33} & k_{34} & k_{35} & k_{36} & k_{37} & k_{38} \\ & & & k_{44} & k_{45} & k_{46} & k_{47} & k_{48} \\ & & & & k_{55} & k_{56} & k_{57} & k_{58} \\ & & & & & k_{66} & k_{67} & k_{68} \\ & & & & & & k_{77} & k_{78} \\ & & & & & & & k_{88} \end{array} \right. \\ \text{simétrica} & & & & & & & \end{array}$$

onde:

$$k_{11} = 36$$

$$k_{12} = k_{13} = k_{16} = k_{17} = 0$$

$$k_{14} = - 3 L$$

$$k_{15} = - 36$$

$$k_{18} = - 3 L$$

$$k_{22} = 36$$

$$k_{23} = 3 L$$

$$k_{24} = k_{25} = k_{28} = 0$$

$$k_{26} = - 36$$

$$k_{27} = - 3 L$$

$$k_{33} = 4 L^2$$

$$k_{34} = k_{35} = k_{38} = 0$$

$$k_{36} = - 3 L$$

$$k_{37} = - L^2$$

$$k_{44} = 4 L^2$$

$$k45 = 3 L$$

$$k46 = k47 = 0$$

$$k48 = - L^2$$

$$k55 = 36$$

$$k56 = k57 = 0$$

$$k58 = 3 L$$

$$k66 = 36$$

$$k67 = - 3 L$$

$$k68 = 0$$

$$k77 = 4 L^2$$

$$k78 = 0$$

$$k88 = 4 L^2$$

8.4. Anexo D - Orientação para instalação e uso do programa de pré-processamento do Programa Rotor, do Programa Rotor e do programa de pós-processamento do Programa Rotor.

1. Configuração mínima exigida.

- a. Microcomputador do tipo IBM-PC-XT com 512 kbytes de memória RAM
- b. Co-processor aritmético compatível com o sistema.
- c. Sistema operacional (DOS), versão 2.1 ou superior.
- d. Compilador FORTRAN, Microsoft ou IBM, versão 2.1 ou superior.
- e. "Linker" FORTRAN, IBM, versão 2.3 ou superior
- f. Disquetes com arquivos de extensão LIB, OBJ e VDI, pertencentes ao GKS.
- g. Impressora gráfica.

## 2. Instalação do GKS.

a. Edição do arquivo config.sys. Incluir os "drivers" para os dispositivos usados nos programas, como:

```
DEVICE=A:\VDIMTFIL.SYS /R
DEVICE=A:\VDIDY006.SYS /G:CRT
DEVICE=A:\VDIPRGRA.SYS /G:PRINT
DEVICE=A:\VDI.SYS
```

b. Edição do arquivo autoexec.bat. Incluir o programa executável INIT\_VDI.EXE, alocado na raiz principal. Como exemplo, tem-se:

```
DATE
TIME
INIT_VDI
```

## 3. Obtenção dos programas executáveis:

a. Programa Rotor.

a.1. Compilação dos programas de extensão FOR, o1, o2, o3, o4, o5, o6, e o7, disponíveis no disquete fonte Rotor.

a.2. "Linkagem" dos programas de extensão OBJ da seguinte forma (obtenção do programa executável ROTOR):

```
LINK o1+o2+o3+o4+o5+o6+o7,ROTOR,FORTRAN.LIB
```

b. Programa de pré-processamento do Programa Rotor.

b.1. Compilação dos programas de extensão FOR, pr1, pr2, pr3, pr4 e pr5, disponíveis no disquete fonte Pré-Pós-Rotor.

b.2. "Linkagem" dos programas de extensão OBJ da seguinte forma (obtenção do programa executável PREROT):

```
LINK      FGKS+pr1+pr2+pr3+pr4+pr5,PREROT,FGKS.LIB+FORTRAN.LIB+
GKS.LIB
```

c. Programa de pós-processamento do programa Rotor.

c.1 Compilação dos programas de extensão FOR, po1, po2, po3, po4, disponíveis no disquete fonte Pré-Pós-Rotor.

c.2. "Linkagem" dos programas de extensão OBJ da seguinte forma (obtenção do programa executável POSROT):

```
LINK FGKS+po1+po2+po3+po4,POSROT,FGKS.LIB+FORTRAN.LIB+GKS.LIB
```