

MAIK BRISCESE MÜLLER

10,0
11/08/2003
Prof. A. K.

ANÁLISE MODAL DE VIGAS E PLACAS

**Relatório Final entregue a Escola Politécnica
da Universidade de São Paulo para a
obtenção do Título de Bacharel em
Engenharia Mecânica**

São Paulo
2002

MAIK BRISCESE MÜLLER

ANÁLISE MODAL DE VIGAS E PLACAS

**Relatório Final entregue a Escola Politécnica
da Universidade de São Paulo para a
obtenção do Título de Bacharel em
Engenharia Mecânica**

Professor orientador: Marcílio Alves

**São Paulo
2002**

INDICE

1. INTRODUÇÃO..... 2

2. OBJETIVOS..... 3

3. REVISÃO BIBLIOGRÁFICA 4

4. MODELO ANALÍTICO DE UMA VIGA SIMPESMENTE ENGASTADA..... 6

5. MODELO ANALÍTICO DE UMA PLACA RETANGULAR BIAPOIADA..... 9

6. A TÉCNICA DA ANÁLISE MODAL EXPERIMENTAL..... 16

7. ANÁLISE MODAL EXPERIMENTAL DE UMA VIGA SIMPLESMENTE ENGASTADA..... 19

8. ANÁLISE MODAL EXPERIMENTAL DE UMA PLACA BIAPOIADA..... 22

9. DISCUSSÃO..... 27

10. BIBLIOGRAFIA..... 28

11. APÊNDICES..... 29

 Programa para ajuste de um conjunto de pontos por uma circunferência..... 29

 Programa para análise modal da viga engastada estudada..... 30

 Programa para análise modal da placa biapoiada estudada..... 50

 Programa para determinação dos modos e frequências naturais analíticas da placa..... 111

1. INTRODUÇÃO

A análise modal desempenha papel importante em compreender o comportamento dinâmico de uma estrutura através da determinação de suas frequências naturais, modos de vibrar e fatores de amortecimento. Tal compreensão acaba por propiciar, por um lado, um adequado projeto de isoladores e controladores de vibração e, por outro, tomar vantagem das vibrações presentes no sistema.

As aplicações da análise modal são inúmeras: verificar a validade de modelos teóricos e permitir usá-los com certa confiabilidade; verificar a durabilidade de um componente submetido a um determinado ambiente dinâmico; monitorar a integridade de uma estrutura ao longo do tempo; diagnosticar a necessidade de manutenção de um sistema mecânico, etc...

Este trabalho apresenta uma metodologia de ensino, aplicada junto a estudantes de graduação em engenharia, que possibilita o ensino de algumas técnicas de análise modal, tanto teóricas como experimentais. Estas técnicas são discutidas à luz de procedimentos experimentais em vigas e placas, levados a cabo em laboratório.

É descrito como as formulações teóricas são exploradas dentro do contexto experimental e discutido a validade das abordagens usadas. Por fim, apresenta-se um programa de computador desenvolvido no ambiente Matlab que permite ao usuário trabalhar com os dados experimentais e proceder à análise modal de estruturas mais gerais.

2. OBJETIVOS

A análise modal experimental de um sistema simples, tal como uma viga engastada ou uma placa biapoiada, permite verificar a validade dos seus resultados através de soluções analíticas disponíveis na literatura. Se verificada a validade, estabelecemos um grau de confiança para o método experimental, permitindo o uso da técnica para análise de estruturas mais complexas, eg carroçarias de automóveis.

Através da análise modal experimental podemos obter as frequências naturais e os respectivos modos de vibrar destas estruturas. Essas informações são de grande valor, desde o projeto inicial – evitando-se frequências de trabalho próximas às frequências de ressonância – até a manutenção preditiva, monitorando-se o comportamento da estrutura ao longo do tempo. De fato, existe hoje na literatura, vasta pesquisa sobre integridade estrutural, baseando-se na análise modal (Hu et al, 2001; Wang et al, 2001; Kim & Bartkowicz, 2001). Na Seção seguinte, serão descritos resumidamente alguns artigos atuais sobre o tema.

Este trabalho descreve as principais etapas a serem seguidas em uma análise modal experimental. As Seções 4 e 5 apresentam, respectivamente, os modelos analíticos de: uma viga engastada em uma extremidade e livre na outra; uma placa retangular biapoiada em duas extremidades opostas. Na Seção 6, apresentam-se as técnicas de análise modal experimental. Nas Seções seguintes, descreve-se em detalhes o aparato experimental a ser utilizado, bem como os procedimentos experimentais da análise modal destas estruturas. Finalmente serão analisados alguns resultados experimentais e analíticos obtidos nestes programas através de gráficos comparativos, permitindo verificar a validade do método experimental. São apresentados também os programas desenvolvidos no Matlab para tratamento de dados destas estruturas, através de interfaces com o usuário, nos apêndices.

3. REVISÃO BIBLIOGRÁFICA

Avaliação de danos em estruturas usando dados de teste modais

Recentemente, a identificação de danos estruturais baseado em técnicas de monitoramento de vibrações tem chamado muita atenção. Vários algoritmos de identificação de danos tem sido desenvolvidos para lidar com três tipos de problemas, i. e. detecção da presença de dano, localização do dano na estrutura e estimativa da extensão do dano. Para os dois últimos problemas citados, a maioria dos métodos existentes pode ser pensado de como um algoritmo de dois estágios, onde a localização do dano é detectada primeiro, e então a extensão deste é estimada.

A indústria de máquinas giratórias tem usado a medição de vibrações, que é um ensaio não destrutivo, como uma ferramenta de diagnóstico para monitorar a integridade estrutural. Mudanças nas frequências obtidas entre medições subsequentes indicam dano ou desgaste estrutural. Apesar destes procedimentos geralmente não gerarem informações quantitativas sobre o dano, o monitoramento contínuo permite a substituição de peças da máquina antes que uma falha catastrófica ocorra.

A maioria das pontes ou estruturas “offshore” são inspecionadas visualmente contra danos estruturais em intervalos regulares especificados. No entanto, este método de avaliação nem sempre revela anomalias estruturais, pois muitas partes da estrutura são inacessíveis a inspeção visual. Além disso, este método não fornece um valor quantitativo da resistência restante da estrutura. Avaliações rápidas e eficientes da saúde estrutural após furacões ou terremotos são altamente desejáveis. Estas necessidades crescentes motivaram um considerável esforço em pesquisa para desenvolver sistemas de monitoramento da integridade estrutural.

Estruturas imponentes, tais como prédios altos e chaminés, podem ser modeladas como colunas engastadas sujeitas a deformação devido a flexão e a força cortante. Uma aproximação da flexibilidade para a identificação de dano em estruturas engastadas utilizando dados dinâmicos dos modos é proposta por Li et al (1998). Uma vantagem importante desta aproximação é que esta requer poucos modos, tornando-a conveniente para aplicações práticas.

A maioria das estruturas de engenharia civil, tais como prédios altos, torres e pontes podem sofrer danos causados pelos efeitos do meio ambiente, tais como terremotos, ventos fortes, fogo e corrosão durante suas vidas. Para assegurar seus funcionamento e segurança, é importante identificar não destrutivamente o estado de dano destas estruturas. Para este propósito, grande numero de promessas de técnicas de avaliação não destrutivas têm sido propostas. A mais pratica entre estas, para estruturas de engenharia civil, é a técnica baseada nos dados de resposta dinâmica da estrutura para identificar um modelo matemático apropriado para representar a estrutura, usando a teoria de identificação de sistemas. Os autores Li et al. (1998) propõem a identificação de estruturas altas, sujeitas a deformação devido a flexão e a força cortante.

Medições de vibrações têm sido usadas com sistemas de algoritmos de identificação para determinar remotamente o local e a extensão do dano estrutural. Um modelo matemático da estrutura íntegra, geralmente associado a dados de teste desta, é usado com as informações de vibrações medidas na estrutura danificada. Este uso não tradicional das medidas de vibrações é uma promessa para aplicação em ensaio não destrutivo remoto para estruturas grandes e complexas. Com esta abordagem implementada, a integridade estrutural pode ser facilmente monitorada e o tempo necessário para uma inspeção visual pode ser significativamente reduzido.

A abordagem da vibração é um método global de ensaio não destrutivo e é limitado a identificar um grau de dano que produza mudanças detectáveis nas características dinâmicas da estrutura. Assim, trincas, fraturas de solda ou pequenos danos podem não ser detectados usando esta abordagem. Este método não pretende substituir métodos tradicionais de ensaios não destrutivos microscópicos tais como raio-X, emissões acústicas ou ultrassônicas. Porém, deve ser útil para grandes estruturas, identificando uma região suspeita que poderá então ser avaliadas por um desses métodos tradicionais de ensaios não destrutivos.

Recentemente, consideráveis esforços em pesquisa têm focado no desenvolvimento de um método não destrutivo para substituir e/ou aumentar a confiabilidade da inspeção visual para a detecção de danos. Testes modais de estruturas (intencionalmente) danificadas têm sido conduzidos para avaliar a performance de diversas abordagens de detecção de danos que usem medidas estruturais dinâmicas. No entanto, é necessário encontrar uma abordagem para a detecção de danos que possa ser usada para estruturas grandes e complexas com instrumentação limitada.

Para aplicações práticas, pode ser mais atrativo evitar empregar muita informação dos modelos analíticos. Consequentemente, vários algoritmos tem sido apresentados na literatura: Hu et al (2000) apresentam uma revisão completa dos modelos analíticos usados para detectar o dano em estruturas e propõem dois algoritmos que possam ser utilizados em estruturas mais complexas. Chen e Bicanic (1999) propõem uma técnica de identificação baseada nas equações características para sistemas de estruturas dinâmicas para obter o dano em estruturas contínuas. Khan et al (2000) identificam deformações estruturais, como trincas, usando um scanner LDV (Vibrômetro a Laser Doppler).

Apesar do desenvolvimento nesta área, muitas dificuldades ainda precisam ser superadas, como a incerteza de medidas e dados provenientes de testes inadequados com os quais se baseiam os algoritmos. Outra dificuldade importante recai sobre a diferença entre modelos analíticos e estruturas reais. Hu et al (2000) citam os seguintes erros: (1) aproximação na condição de contorno pode fazer com que uma matriz de rigidez analítica e algorítmica divirja da prática, (2) condições de conectividade de elementos nos modelos podem não refletir o real estado conetivo de membros estruturais, (3) distorção de alguns parâmetros importantes do material, tal como o módulo de Young, (4) existem muitas fontes de rigidez em estruturas práticas que são ignoradas pelo modelo analítico devido a capacidade computacional e (5) a malha grossa ou tipos de elemento inadequados podem causar erros nos modelos analíticos. Atualmente, para estruturas complexas, nenhum modelo analítico razoável pode ser encontrado.

4. MODELO ANALÍTICO DE UMA VIGA SIMPESMENTE ENGASTADA

Considere a vibração transversal ou de flexão de uma viga com uma de suas extremidades engastada e a outra livre, conforme indicado na Fig. (1). A viga em questão tem comprimento l , largura $b(x)$, espessura $h(x)$, densidade ρ , módulo de elasticidade E , momento de inércia em relação ao eixo z $I(x)$, área da seção transversal $A(x) = b(x) h(x)$. As variáveis $w(x,t)$, $M(x,t)$, $V(x,t)$ e $f(x,t)$ representam a deflexão, o momento fletor, a força cortante e a força externa por unidade de comprimento da viga, respectivamente. A resposta analítica pode ser encontrada na literatura (Inman, 2000; Dimarogonas, 1995) e será brevemente resumida nesta seção.

Para o campo de pequenas deformações e dx suficientemente pequeno, segue, da resistência dos materiais, que momento fletor e esforço cortante estão relacionados à deflexão através das equações:

$$M(x,t) = EI(x) \frac{\partial^2 w(x,t)}{\partial x^2} \qquad V(x,t) = -\frac{\partial M(x,t)}{\partial x} = -\frac{\partial}{\partial x} \left(EI \frac{\partial^2 w(x,t)}{\partial x^2} \right) \qquad (1)$$

Assumindo pequenos deslocamentos, a deformação devido ao cisalhamento é desprezível em relação à $w(x,t)$ e o somatório das forças na direção de y resulta:

$$\left(V(x,t) + \frac{\partial V(x,t)}{\partial x} dx \right) - V(x,t) + f(x,t) dx = \rho \cdot A(x) dx \frac{\partial^2 w(x,t)}{\partial t^2} \qquad (2)$$

Substituindo as Eqs. (1) na Eq. (2), resulta:

$$\frac{\partial^2 w(x,t)}{\partial t^2} + c^2 \frac{\partial^4 w(x,t)}{\partial x^4} = 0, \quad c = \sqrt{\frac{EI}{A\rho}} \qquad (3)$$

onde, por simplificação, considera-se $A(x) = A$ e $EI(x) = EI$ constantes, e $f(x,t) = 0$, i.e. vibração livre.

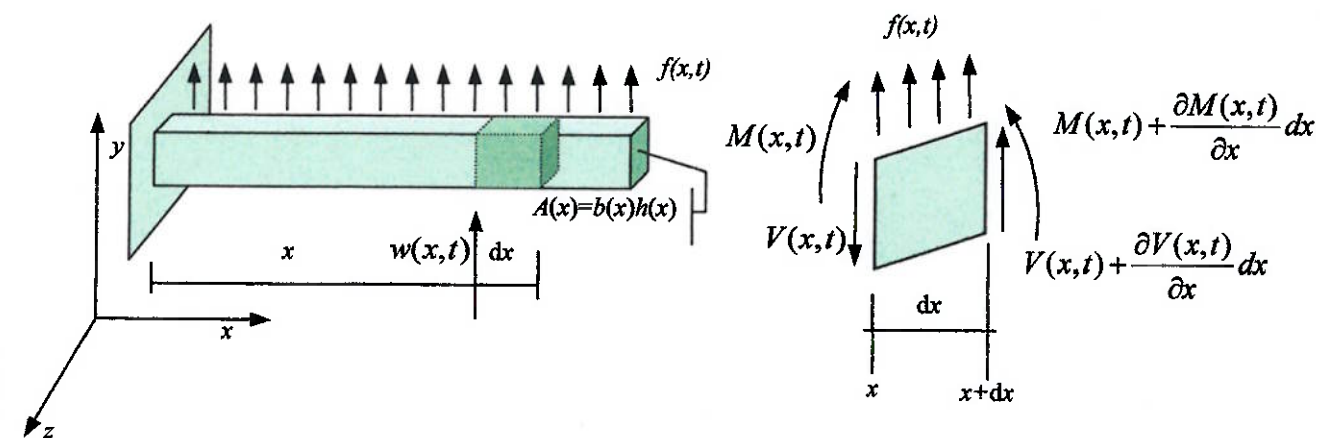


Figura 1. (a) Viga engastada sob vibração transversal livre.
(b) Elemento de viga infinitesimal e seus esforços.

Assumindo que as variáveis da Eq. (3) possam ser separadas na forma $w(x,t) = X(x)T(t)$:

$$c^2 \frac{X^{(4)}(x)}{X(x)} = -\frac{\ddot{T}(t)}{T(t)} = \omega^2 \quad (4)$$

Dessa forma, para o lado direito da equação acima, tem-se que,

$$\ddot{T}(t) + \omega^2 T(t) = 0 \quad (5)$$

cujas soluções são dadas na forma,

$$T(t) = A \sin(\omega t) + B \cos(\omega t) \quad (6)$$

As constantes A e B dependem das condições iniciais do problema. A equação espacial pode ser escrita como,

$$X^{(4)}(x) - \beta^4 X(x) = 0, \quad \beta^4 = \frac{\omega^2}{c^2} = \frac{A \rho \omega^2}{EI} \quad (7)$$

cujas soluções são governadas pela equação,

$$X(x) = a_1 \sin(\beta x) + a_2 \cos(\beta x) + a_3 \sinh(\beta x) + a_4 \cosh(\beta x) \quad (8)$$

onde β e três das quatro constantes de integração a_1 , a_2 , a_3 e a_4 serão determinadas a partir das condições de contorno. As condições de contorno da Eq. (8) são obtidas examinando a deflexão $w(x, t)$, a rotação $\theta(x, t) = \partial w(x, t) / \partial x$, o momento fletor (Eq. (1a)), e força cortante (Eq. (1b)) em cada uma das extremidades da viga. Para o caso de viga engastada-livre, tem-se que:

$$M(x, t)|_{x=l} = V(x, t)|_{x=l} = w(x, t)|_{x=0} = \theta(x, t)|_{x=0} = 0 \quad (9)$$

ou

$$X(x)|_{x=0} = X'(x)|_{x=0} = EI X''(x)|_{x=l} = EI X'''(x)|_{x=l} = 0 \quad (10)$$

A solução do sistema de equações (8) e (10) é diferente da solução trivial se:

$$\cos(\beta l) \cdot \cosh(\beta l) = 1 \quad (11)$$

E os modos de vibrar ficam:

$$X_n = a_4 \left[-\cos(\beta_n x) + \cosh(\beta_n x) + \frac{(\cos(\beta_n l) + \cosh(\beta_n l))}{\sin(\beta_n l) + \sinh(\beta_n l)} (\sin(\beta_n x) - \sinh(\beta_n x)) \right] \quad (12)$$

para $n = 1, 2, 3, \dots$

A quarta constante pode ser determinada em função das condições iniciais, constantes A e B da equação temporal.

A figura 2 abaixo apresenta alguns modos de vibrar da viga em questão, regidos pela equação (12).

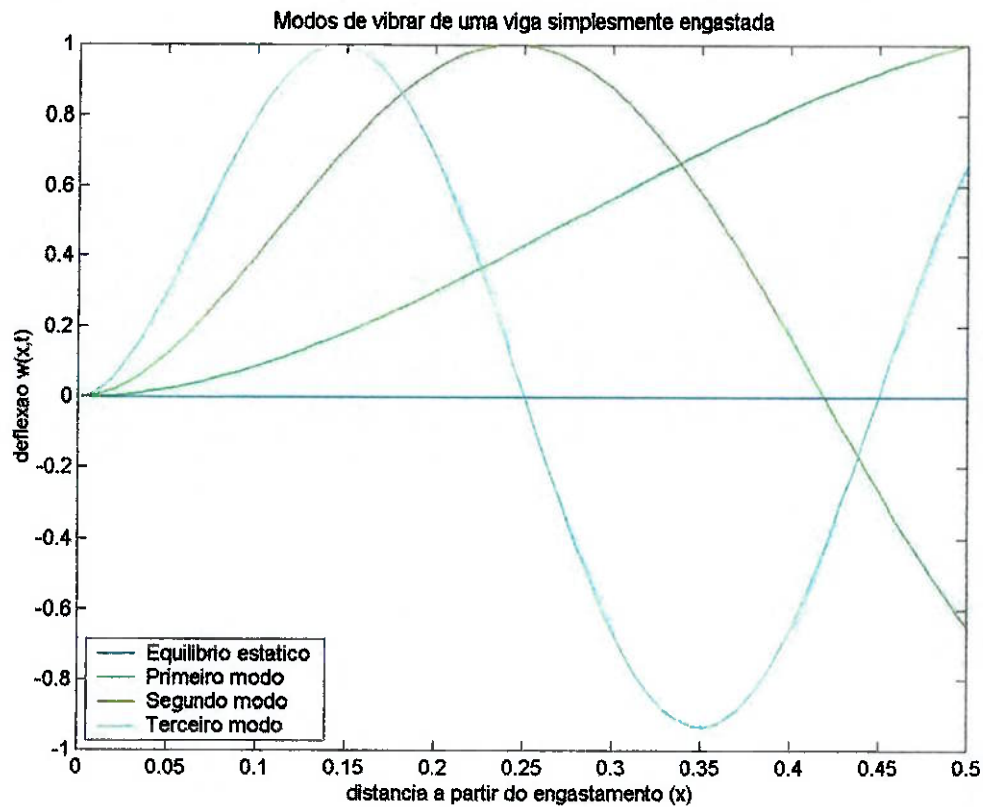


Figura 2. Modos de vibrar analíticos de uma viga simplesmente engastada.

5. MODELO ANALÍTICO DE UMA PLACA RETANGULAR BIAPOIADA

Estudaremos agora as vibrações transversais de placas planas, de espessura h conhecida.

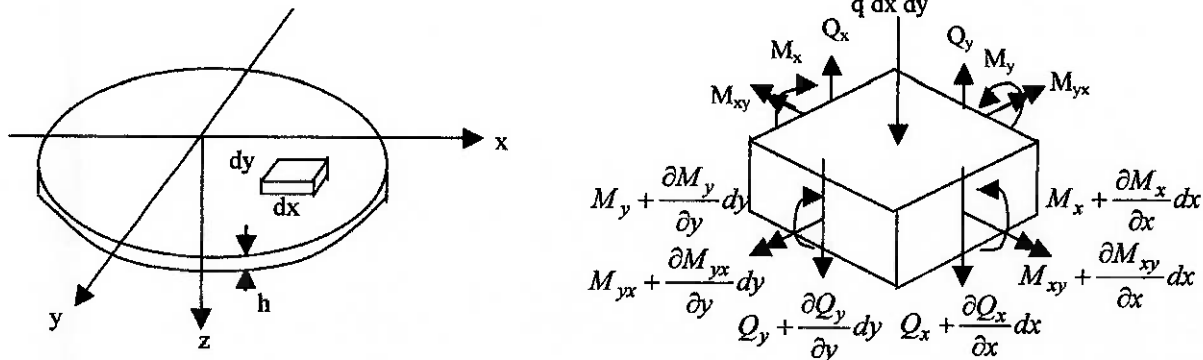


Figura 3. (a) Região de uma placa (b) Elemento de placa sujeito a forças e momentos

A partir das equações de equilíbrio de forças na direção de z e de equilíbrio de momentos na direção x e y , após algumas manipulações matemáticas, resulta em:

$$\frac{\partial^2 M_x}{\partial x^2} + \frac{\partial^2 M_{yx}}{\partial x \partial y} - \frac{\partial^2 M_{xy}}{\partial y \partial x} + \frac{\partial^2 M_y}{\partial y^2} + q = \rho h \frac{\partial^2 w}{\partial t^2} \tag{13}$$

Escrevendo os momentos em função da deflexão w , fica:

$$M_x = -D \left(\frac{\partial^2 w}{\partial x^2} + \nu \frac{\partial^2 w}{\partial y^2} \right); \quad M_y = -D \left(\frac{\partial^2 w}{\partial y^2} + \nu \frac{\partial^2 w}{\partial x^2} \right); \quad M_{xy} = D(1 - \nu) \frac{\partial^2 w}{\partial x \partial y} = -M_{yx}; \tag{14}$$

onde $D = \frac{Eh^3}{12(1 - \nu^2)}$;

Assim, substituindo as equações (14) na equação (13), obtemos:

$$D \left(\frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} \right) - q = -\rho h \frac{\partial^2 w}{\partial t^2} \tag{15}$$

Vamos considerar as vibrações livres de uma placa retangular de comprimento a e largura b . Considerando que a separação de variáveis da solução da Eq. (15) ocorra, impomos:

$$w(x, y, t) = X(x)Y(y)e^{-i\omega t} \tag{16}$$

Obtendo a seguinte equação:

$$X^{IV}Y + 2X^{II}Y^{II} + XY^{IV} - \beta^4 XY = 0 \quad (17)$$

onde $\beta^4 = \omega^2 \rho h / D$. Para que ocorra a separação de variáveis, devemos ter que:

$$Y^{II} = -\gamma^2 Y, \quad Y^{IV} = \gamma^4 Y \quad \text{ou} \quad X^{II} = -\alpha^2 X, \quad X^{IV} = \alpha^4 X \quad \text{ou ambos.}$$

Supondo que o segundo caso seja válido. A solução deve ser da forma:

$$X = A_1 \sin(\alpha x) + A_2 \cos(\alpha x), \quad \alpha^2 \geq 0 \quad (18)$$

Considerando-se as possibilidades de condições de contorno simples ao longo das bordas $x=0$ e $x=a$, temos as seguintes possibilidades ao longo de $x=0$: Apoiada, Engastada, Livre; resultando:

- A: $X(0)Y(y) = [-\alpha^2 Y(y) + \nu Y^{II}(y)]X(0) = 0$,
- E: $X(0)Y(y) = X^I(0)Y(y) = 0$,
- L: $X(0)[- \alpha^2 Y(y) + \nu Y^{II}(y)] = X^I(0)[- \alpha^2 Y(y) + (2 - \nu)Y^{II}(y)] = 0$,

com condições similares em $x=a$. É fácil perceber que o único conjunto de soluções que satisfaz a Eq. (18) é o caso da placa apoiada nas duas bordas (biapoiada), para a qual

$$X_n = A_1 \sin(\alpha_n x), \quad \alpha_n = n \frac{\pi}{a} \quad (n=1,2,\dots) \quad (20)$$

é uma solução. Para qualquer outra combinação de condições de contorno, não haveria solução analítica. Assim, escolhamos uma placa com duas bordas apoiadas (em $x=0$ e $x=a$) e duas bordas livres (em $y=0$ e $y=b$), para que o sistema tenha solução analítica exata e para que a placa tenha alguns modos de vibrar de vigas biapoiadas e outros exclusivos de placa. Infelizmente, o problema de vibrações livres em placas livres no espaço, de fácil montagem em laboratório, não tem solução analítica exata para a equação que governa o movimento – Eq. (17) – tornando as frequências naturais impossíveis de serem obtidas analiticamente.

Agora, devemos ter que Y satisfaça:

$$Y^{IV} - \alpha_n^2 Y^{II} - (\beta^4 - \alpha_n^4)Y = 0 \quad (21)$$

A equação (21) apresenta diferentes soluções de acordo com os valores de α e β . Distingue-se os casos:

Caso 1: $\beta^4 > \alpha_n^4$, Y será da forma:

$$Y = B_1 \sin(sy) + B_2 \cos(sy) + B_3 \sinh(ry) + B_4 \cosh(ry) \quad (22)$$

onde $r = \sqrt{\beta^2 + \alpha_n^2}$ e $s = \sqrt{\beta^2 - \alpha_n^2}$.

As condições de contorno das bordas livres da placa em $y=0$ e $y=b$ são:

- $M_x(x,0) = -D \left(\frac{\partial^2 w(x,0)}{\partial x^2} + \nu \frac{\partial^2 w(x,0)}{\partial y^2} \right) = M_x(x,b) = -D \left(\frac{\partial^2 w(x,b)}{\partial x^2} + \nu \frac{\partial^2 w(x,b)}{\partial y^2} \right) = 0$, ou seja, momento nulo nas bordas;

- $Q(x,0) = \frac{\partial M(x,0)}{\partial x} = Q(x,b) = \frac{\partial M(x,b)}{\partial x} = 0$, ou seja, cortante nula nas bordas livres, decorre que:

$$\begin{cases} Y''(0)X(x) - \nu Y(0)\alpha^2 X(x) = 0 \\ Y''(b)X(x) - \nu Y(b)\alpha^2 X(x) = 0 \\ Y'''(0)X(x) - (2-\nu)Y'(0)\alpha^2 X(x) = 0 \\ Y'''(b)X(x) - (2-\nu)Y'(b)\alpha^2 X(x) = 0 \end{cases}, \forall x \in [0, a] \quad (23)$$

que na forma matricial em função das incógnitas B_1, B_2, B_3 e B_4 , fica:

$$\begin{bmatrix} 0 & -(s^2 + \nu\alpha^2) & 0 & (r^2 - \nu\alpha^2) \\ -(s^2 + \nu\alpha^2)\sin(sb) & -(s^2 + \nu\alpha^2)\cos(sb) & (r^2 - \nu\alpha^2)\sinh(rb) & (r^2 - \nu\alpha^2)\cosh(rb) \\ -s(s^2 + (2-\nu)\alpha^2) & 0 & r(r^2 - (2-\nu)\alpha^2) & 0 \\ -s(s^2 + (2-\nu)\alpha^2)\cos(sb) & s(s^2 + (2-\nu)\alpha^2)\sin(sb) & r(r^2 - (2-\nu)\alpha^2)\cosh(rb) & r(r^2 - (2-\nu)\alpha^2)\sinh(rb) \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix} = 0$$

Para que a solução do sistema seja diferente da trivial, o determinante da matriz 4x4 deve ser igual a zero, ou seja:

$$\sinh(rb)\sin(sb) \left\{ r^2 [\beta^2 - \alpha^2(1-\nu)]^4 - s^2 [\beta^2 + \alpha^2(1-\nu)]^4 \right\} - 2sr [\cosh(rb)\cos(sb) - 1] [\beta^4 - \alpha^4(1-\nu)^2] = 0 \quad (24)$$

Para uma placa de aço com as seguintes características:

$$E=210 \text{ GPa}; \quad \rho=7860 \text{ kg/m}^3; \quad \nu=0,3; \quad a=0,3 \text{ m}; \quad b=0,2 \text{ m}; \quad h=0,002 \text{ m};$$

e sabendo que $f = \frac{\beta^2}{2\pi} \sqrt{\frac{D}{\rho h}}$, obtemos as seguintes frequências naturais (até 2200 Hz):

n=1	$f_{11}=121,1 \text{ Hz}$	n=2	$f_{21}=303,5 \text{ Hz}$	n=3	$f_{31}=581,8 \text{ Hz}$
	$f_{12}=363,3 \text{ Hz}$		$f_{22}=575,1 \text{ Hz}$		$f_{32}=880,5 \text{ Hz}$
	$f_{13}=845,3 \text{ Hz}$		$f_{23}=1059 \text{ Hz}$		$f_{33}=1382 \text{ Hz}$
	$f_{14}=1577 \text{ Hz}$		$f_{24}=1784 \text{ Hz}$		$f_{34}=2110 \text{ Hz}$

Os modos de vibrar ficam:

$$W_{nm} = A_1 \sin(\alpha_n x) [B_1 \sin(sy) + B_2 \cos(sy) + B_3 \sinh(ry) + B_4 \cosh(ry)] \quad (25)$$

$$W_{nm} = C \sin(\alpha_n x) \left\{ \begin{aligned} & r [\cos(sb) - \cosh(rb)] [\beta^4 - \alpha_n^4(1-\nu)^2] [\beta^2 - \alpha_n^2(1-\nu)] \sin(sy) + \\ & \left\{ s [\beta^2 + \alpha_n^2(1-\nu)]^2 \sinh(rb) - r [\beta^2 - \alpha_n^2(1-\nu)]^2 \sin(sb) \right\} [\beta^2 + \alpha_n^2(1-\nu)] \cos(sy) + \\ & s [\cos(sb) - \cosh(rb)] [\beta^4 - \alpha_n^4(1-\nu)^2] [\beta^2 + \alpha_n^2(1-\nu)] \sinh(ry) + \\ & \left\{ s [\beta^2 + \alpha_n^2(1-\nu)]^2 \sinh(rb) - r [\beta^2 - \alpha_n^2(1-\nu)]^2 \sin(sb) \right\} [\beta^2 - \alpha_n^2(1-\nu)] \cosh(ry) \end{aligned} \right\} \quad (26)$$

Caso 2: $\beta^4 = \alpha_n^4$, Y será da forma:

$$Y = C_1 + C_2 y + C_3 e^{-\sqrt{2}\alpha_n y} + C_4 e^{\sqrt{2}\alpha_n y} \quad (27)$$

Aplicando as condições de contorno das bordas livres da placa em $y=0$ e $y=b$:

$$\begin{bmatrix} -\nu \cdot \alpha_n^2 & 0 & \alpha_n^2(2-\nu) & \alpha_n^2(2-\nu) \\ -\nu \cdot \alpha_n^2 & -\nu \cdot \alpha_n^2 \cdot b & \alpha_n^2(2-\nu)e^{-\sqrt{2}\alpha_n b} & \alpha_n^2(2-\nu)e^{\sqrt{2}\alpha_n b} \\ 0 & -\alpha_n^2(2-\nu) & -\sqrt{2}\alpha_n^3 \cdot \nu & \sqrt{2}\alpha_n^3 \cdot \nu \\ 0 & -\alpha_n^2(2-\nu) & -\sqrt{2}\alpha_n^3 \cdot \nu e^{-\sqrt{2}\alpha_n b} & \sqrt{2}\alpha_n^3 \cdot \nu e^{\sqrt{2}\alpha_n b} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = 0 \quad (28)$$

Para este caso, não existe solução diferente da trivial ($C_1=C_2=C_3=C_4=0$), ou seja, o determinante da matriz 4x4 acima é sempre diferente de zero. Portanto, não existem modos de vibrar da forma como descrita acima.

Caso 3: $\beta^4 < \alpha_n^4$, Y será da forma:

$$Y = D_1 \sinh(py) + D_2 \cosh(py) + D_3 \sinh(ry) + D_4 \cosh(ry) \quad (29)$$

onde $r = \sqrt{\beta^2 + \alpha_n^2}$ e $p = \sqrt{\alpha_n^2 - \beta^2}$.

Das condições de contorno da placa:

$$\begin{bmatrix} 0 & p^2 - \nu\alpha^2 & 0 & (r^2 - \nu\alpha^2) \\ (p^2 - \nu\alpha^2)\sinh(pb) & (p^2 - \nu\alpha^2)\cosh(pb) & (r^2 - \nu\alpha^2)\sinh(rb) & (r^2 - \nu\alpha^2)\cosh(rb) \\ p(p^2 - (2-\nu)\alpha^2) & 0 & r(r^2 - (2-\nu)\alpha^2) & 0 \\ p(p^2 - (2-\nu)\alpha^2)\cosh(pb) & p(p^2 - (2-\nu)\alpha^2)\sinh(pb) & r(r^2 - (2-\nu)\alpha^2)\cosh(rb) & r(r^2 - (2-\nu)\alpha^2)\sinh(rb) \end{bmatrix} \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \end{bmatrix} = 0$$

Para que a solução do sistema seja diferente da trivial, o determinante da matriz 4x4 deve ser igual a zero:

$$\sinh(rb)\sinh(pb) \left\{ r^2 [\beta^2 - \alpha^2(1-\nu)]^2 + p^2 [\beta^2 + \alpha^2(1-\nu)]^2 \right\} - 2rp [\cosh(rb)\cosh(pb) - 1] [\beta^4 - \alpha^4(1-\nu)^2] = 0 \quad (30)$$

Para a placa de aço descrita anteriormente, obtemos as seguintes frequências naturais:

$$n=1 \quad f_{01}=52,9 \text{ Hz} \quad n=2 \quad f_{02}=214,2 \text{ Hz} \quad n=3 \quad f_{03}=484,8 \text{ Hz} \quad n=4 \quad f_{04}=864,4 \text{ Hz}.$$

Os modos de vibrar ficam:

$$W_{nm} = A_1 \sin(\alpha_n x) [D_1 \sinh(py) + D_2 \cosh(py) + D_3 \sinh(ry) + D_4 \cosh(ry)] \quad (31)$$

$$D_1 = \left[a_{21} \sinh(pb) - \frac{a_{23}a_{12}}{a_{14}} \sinh(rb) + a_{12}(\cosh(pb) - \cosh(rb)) \right]; D_3 = -\frac{a_{21}}{a_{23}} D_1;$$

$$D_2 = \left[a_{12} \sinh(pb) - \frac{a_{14}a_{21}}{a_{23}} \sinh(rb) + a_{21}(\cosh(pb) - \cosh(rb)) \right]; D_4 = -\frac{a_{12}}{a_{14}} D_2;$$

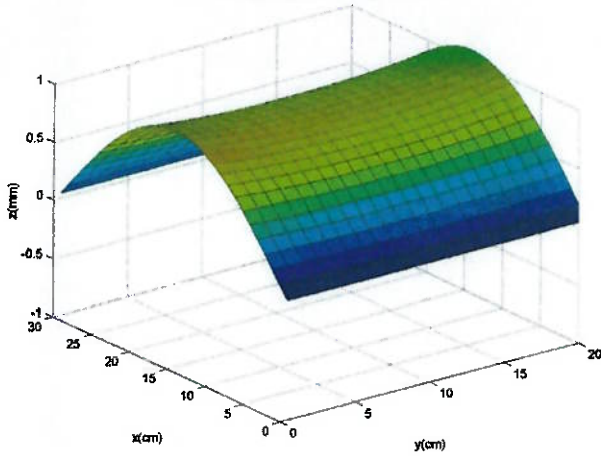
(32)

onde a_{ij} é o elemento da linha i , coluna j da matriz 4×4 acima.

Usando a teoria de viga apresentada anteriormente e as dimensões e condições de contorno da placa, encontramos as seguintes frequências naturais: $f_{01}=52,1$ Hz, $f_{02}=208,3$ Hz, $f_{03}=468,8$ Hz e $f_{04}=833,4$ Hz.

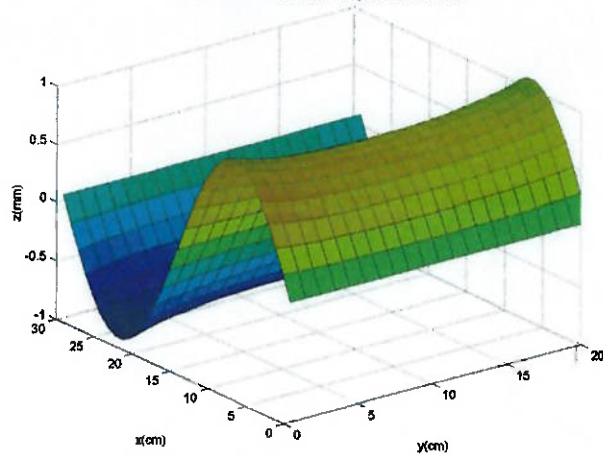
As figuras (4a, 4b, 4c e 4d) representam os modos de vibrar da placa estudada para o Caso 3, enquanto as figuras (4e – 4q) representam os modos de vibrar para o Caso 1.

1o. modo de vibrar analítico - frequência: 52.8774 Hz



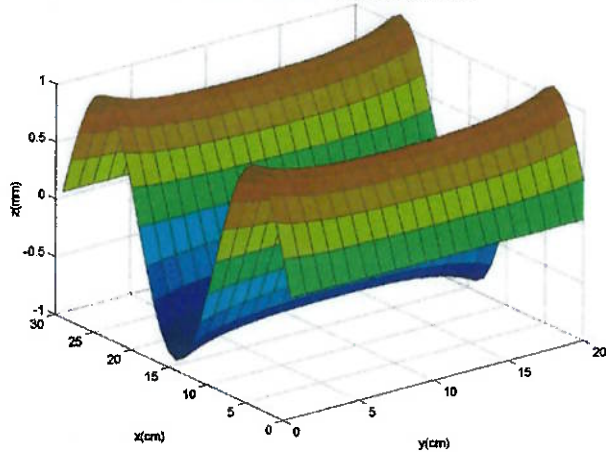
a) Modo 01.

2o. modo de vibrar analítico - frequência: 214.213 Hz



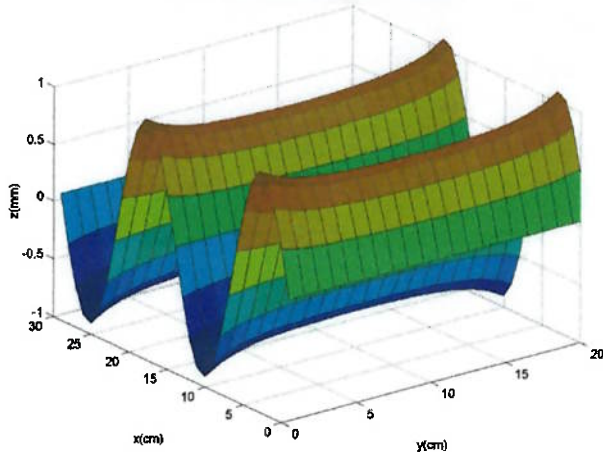
b) Modo 02.

3o. modo de vibrar analítico - frequência: 484.7619 Hz



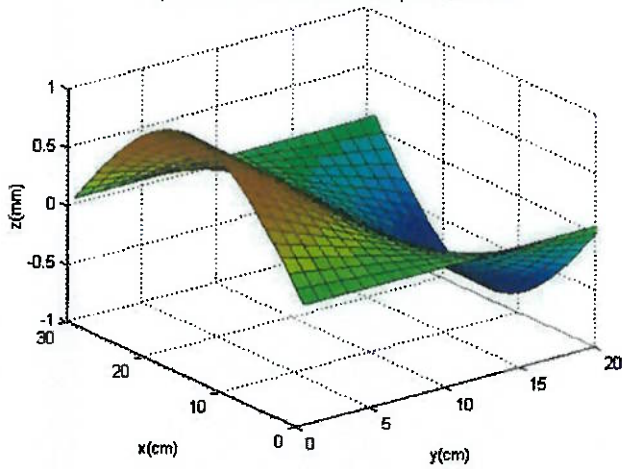
c) Modo 03.

4o. modo de vibrar analítico - frequência: 864.4423 Hz



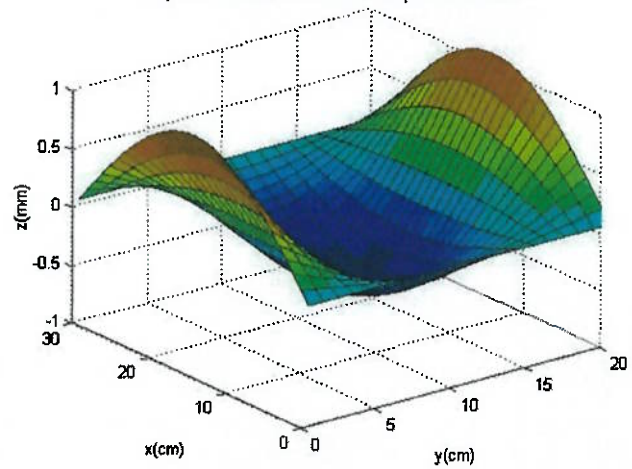
d) Modo 04.

n=1; 1o. modo de vibrar analítico - frequência: 121.14Hz



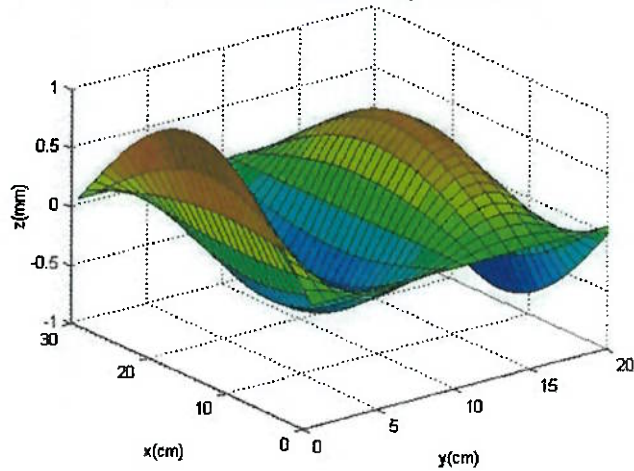
e) Modo 11.

n=1; 2o. modo de vibrar analítico - frequência: 363.31Hz



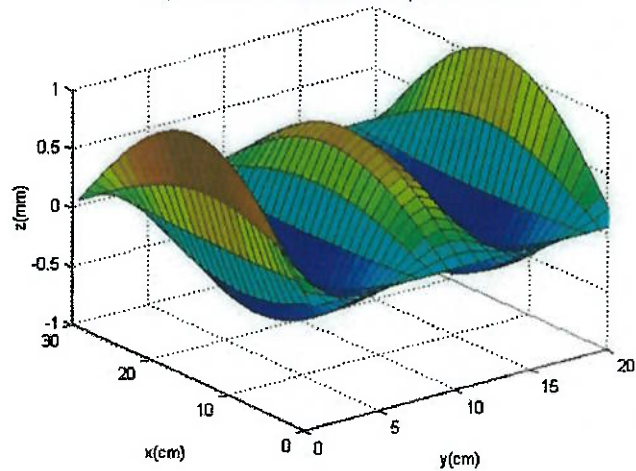
f) Modo 12.

n=1; 3o. modo de vibrar analítico - frequência: 845.27Hz



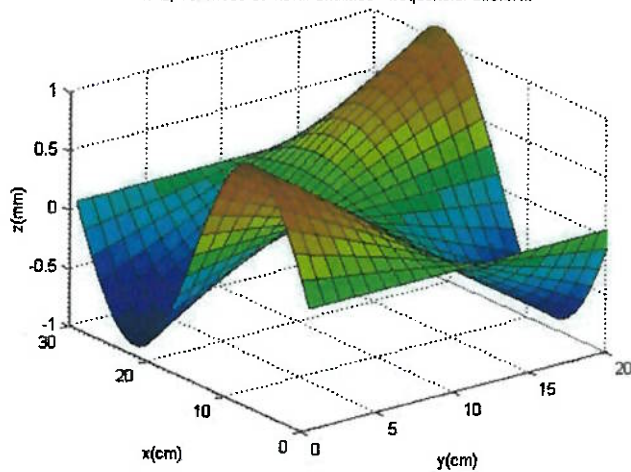
g) Modo 13.

n=1; 4o. modo de vibrar analítico - frequência: 1576.6Hz



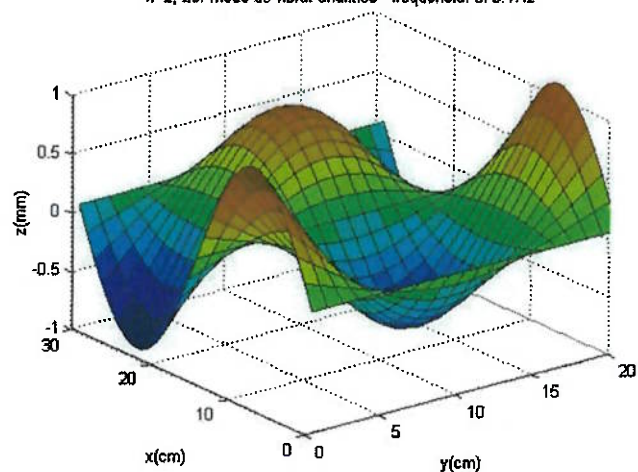
h) Modo 14.

n=2; 1o. modo de vibrar analítico - frequência: 303.5Hz



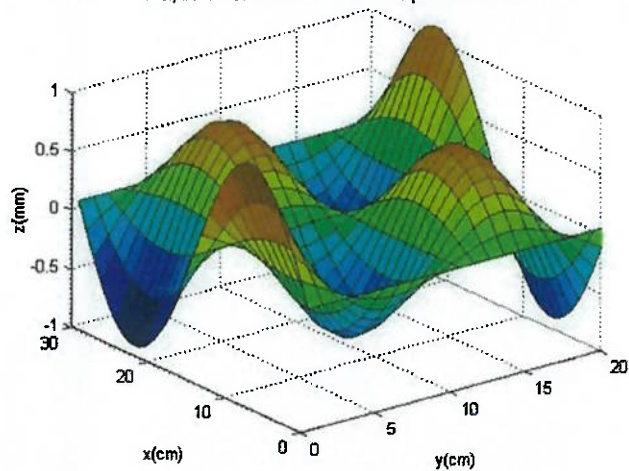
i) Modo 21.

n=2; 2o. modo de vibrar analítico - frequência: 575.1Hz



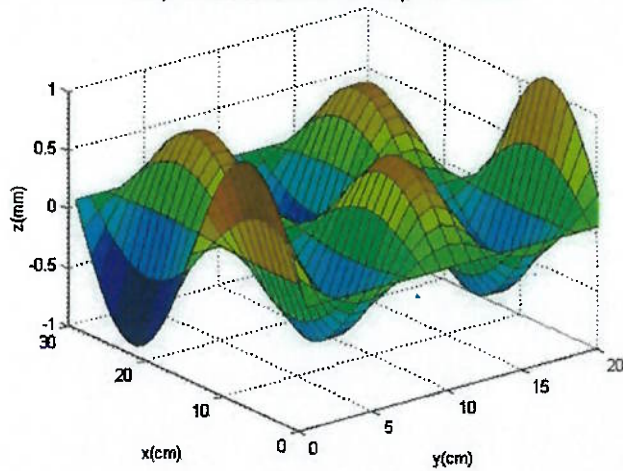
j) Modo 22.

$n=2$; 3o. modo de vibrar analítico - frequência: 1059Hz



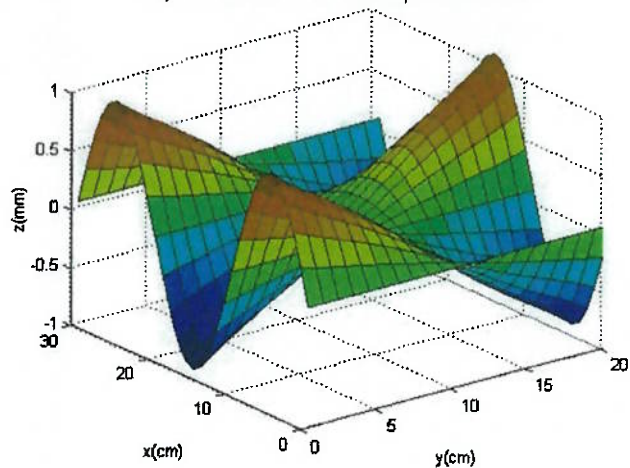
l) Modo 23.

$n=2$; 4o. modo de vibrar analítico - frequência: 1783.8Hz



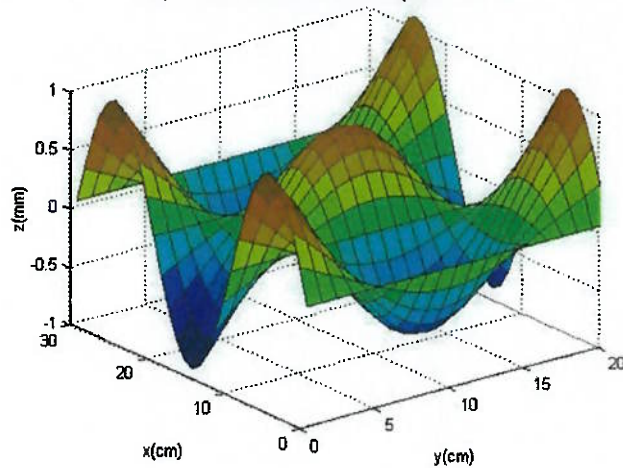
m) Modo 24.

$n=3$; 1o. modo de vibrar analítico - frequência: 581.77Hz



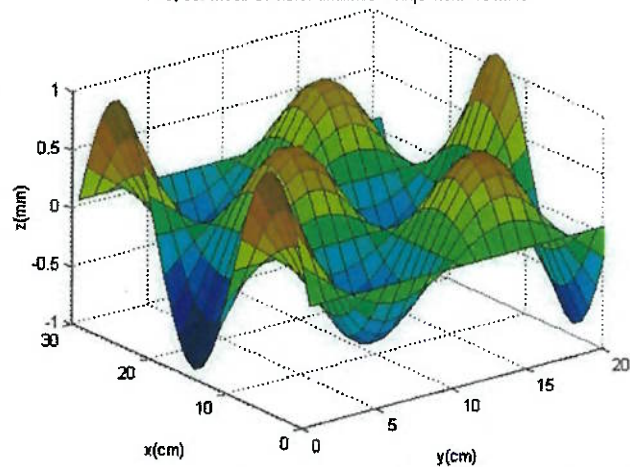
n) Modo 31.

$n=3$; 2o. modo de vibrar analítico - frequência: 860.52Hz



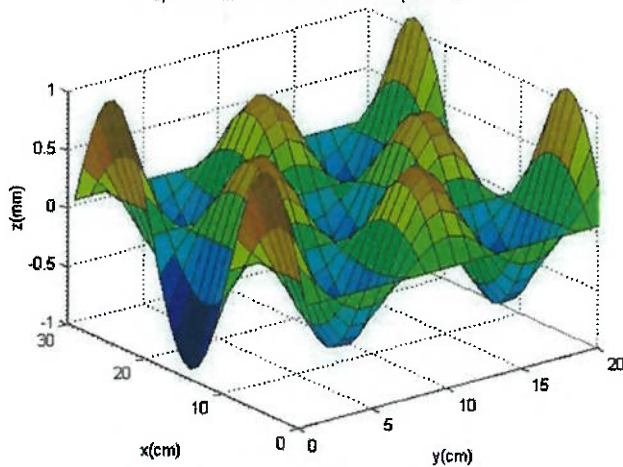
o) Modo 32.

$n=3$; 3o. modo de vibrar analítico - frequência: 1362Hz



p) Modo 33.

$n=3$; 4o. modo de vibrar analítico - frequência: 2110.3Hz



q) Modo 34.

Figura 4: Modos de vibrar e frequências naturais de uma placa retangular biapoada.

6. A TÉCNICA DA ANÁLISE MODAL EXPERIMENTAL

A análise modal experimental de uma estrutura consiste em excitar suas frequências naturais, seja utilizando-se um shaker e um variador de frequência, seja aplicando-se uma função impulso à viga através de um pequeno martelo. No caso em estudo, utilizaremos o segundo método. A técnica consiste em aplicar uma função impulso à estrutura, através de uma batida com um pequeno martelo, e medir a resposta a essa excitação através de um acelerômetro, conforme detalhado na Fig. (5). Tal forma de excitação, se corretamente aplicada, estimula a estrutura a vibrar livremente em todo seu espectro de frequência.

No caso da viga, o acelerômetro é fixado a extremidade livre da mesma. A cada aquisição de sinal, excita-se a estrutura em um dos pontos determinados para este fim. Já para o caso da placa, escolheu-se fixar o ponto de excitação, e a cada aquisição o que muda é a posição do acelerômetro na estrutura. Esses pontos de excitação/aquisição podem ser convenientemente escolhidos, por exemplo, localizando, através dos respectivos modelos analíticos, os pontos da estrutura onde cada modo de vibrar tem seu pico de amplitude. Porém, como geralmente não temos disponível esse modelo, discretizamos as estruturas em N pontos igualmente espaçados. A viga foi discretizada em dez pontos espaçados de 5 cm entre si, a partir da extremidade livre, e foram realizadas três medições por ponto de excitação para eliminar possíveis fontes de erro. A placa recebeu uma malha quadriculada de 2 cm de lado, e as aquisições foram realizadas ao longo de duas linhas (uma transversal, onde foram feitas 14 aquisições, e outra longitudinal, mais 11 aquisições).

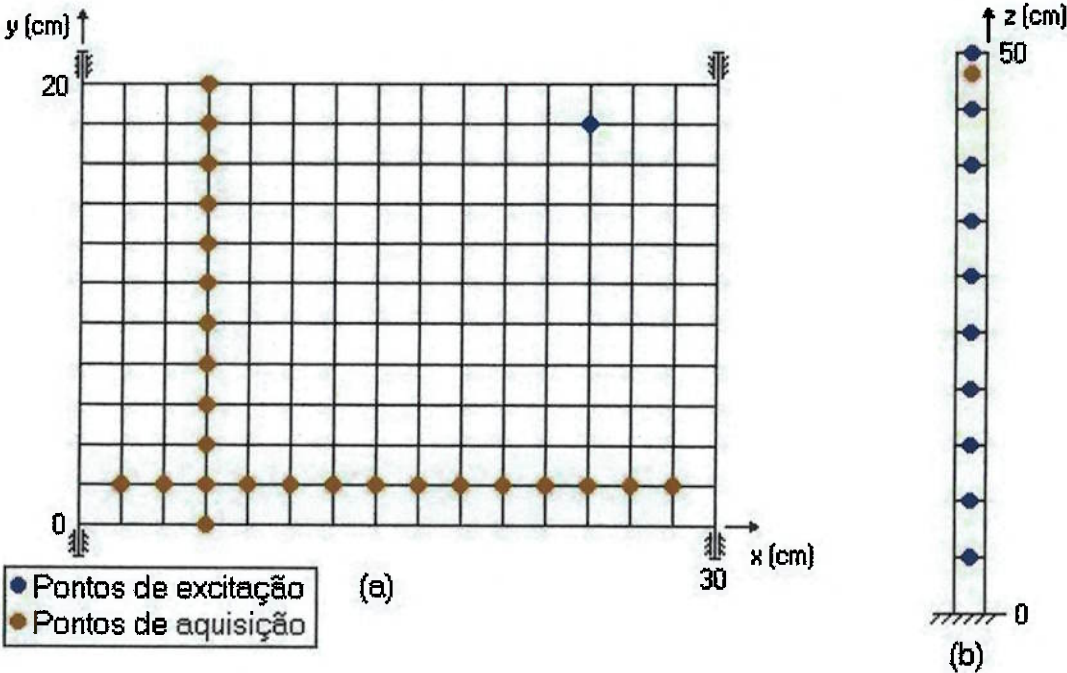


Figura 5. (a) Discretização da placa biapojada.

(b) Discretização da viga engastada.

Os dados referentes à velocidade transversal de cada um dos pontos das estruturas foram armazenados em arquivos no disco rígido do computador. Estes arquivos são os parâmetros de entrada para os programas de análise de dados desenvolvido no Matlab, detalhados na próxima seção. Os programas transformam os dados no domínio do tempo para o domínio da frequência fazendo-se a FFT (Fast Fourier Transform) do sinal, obtendo as funções de resposta em frequência $\alpha(\omega)_r$ para cada um dos pontos estruturais adquiridos, onde s refere-se ao ponto de excitação e r refere-se ao ponto de leitura. Os picos dos gráficos $\alpha(\omega)_r \times \omega$ representam, na maioria das vezes, frequências naturais. Os dados extraídos experimentalmente necessitam de uma série

de transformações para se obter os modos de vibrar do sistema. Grande parte destas transformações consiste na diagonalização de sistemas de matrizes. Essa diagonalização permite analisar os modos de vibrar de maneira desacoplada. Desse modo, tem-se uma implementação numérica de fácil resolução.

$\alpha(\omega_{dr})$ é a matriz de receptância e define $u = \alpha(\omega_{dr})f$, relacionando magnitude do vetor de resposta, vetor u , com a magnitude do vetor de entrada f para frequência de trabalho ω_{dr} . Tem-se uma conexão entre a matriz de receptância e os modos de vibrar do sistema,

$$\alpha(\omega_{dr}) = \sum_{i=1}^n \left[\frac{u_i u_i^T}{(\omega_i^2 - \omega_{dr}^2) + (2\zeta_i \omega_i \omega_{dr})j} \right] \quad (33)$$

A Eq. (33) pode ser obtida mediante as transformações triviais usadas para derivar coordenadas modais (Inman, 2000), onde u_i são os vetores modos de vibrar; ζ_i é a constante de amortecimento relacionada à i -ésima frequência natural ω_i ; n é o número de frequências naturais que podem ser obtidas pelo teste; j é o número complexo $j = \sqrt{-1}$. O vetor u_i tem dimensão $n \times 1$ e cada termo j desse vetor pode ser interpretado como o deslocamento relativo da posição j quando a estrutura está vibrando no modo i . Portanto, determinar o modo i de vibrar de uma viga, considerando-se n graus de liberdade, significa determinar as componentes do vetor u_i ,

$$|u_i^T| = [u_1 \ u_2 \ u_3 \ u_4 \ u_5 \ u_6 \ u_7 \ u_8 \ u_9 \ u_{10}] \quad (34)$$

Como uma generalização do conceito de função de transferência para um grau de liberdade, pode-se dizer que o elemento da matriz de receptância localizado na interseção da linha s com a coluna r de $\alpha(\omega_{dr})$ é a função de transferência entre a resposta no ponto s e a entrada no ponto r quando todas as outras entradas são nulas. Se, além disso, assume-se que os modos do sistema estão suficientemente espaçados, de maneira que não se sobreponham, o somatório da Eq. (33), quando avaliado na frequência natural i , isto é, $\omega_{dr} = \omega_i$, será dominado pelo termo correspondente a essa frequência:

$$|u_i u_i^T|_{sr} = 2\zeta_i \omega_i^2 |\alpha_{sr}(\omega_i)| \quad (35)$$

onde é assumido que as contribuições de outros termos no somatório são nulas. Como a matriz $n \times n$ $|u_i u_i^T|_{sr}$ apresenta apenas n incógnitas, deve-se obter n equações através da relação (35), ie n medidas de $|\alpha_{sr}(\omega_i)|$. No caso da viga analisada, a estrutura será excitada em cada um dos n graus de liberdade, ie, $s: 1 \rightarrow 10$, enquanto que a função resposta será lida sempre no mesmo ponto, $r = 10$. Dessa maneira, obtém-se a décima coluna da matriz $|\alpha_{sr}(\omega_i)|$, que pode ser relacionada com a décima coluna da matriz $|u_i u_i^T|$ conforme Eq. (35). Já para a placa, a estrutura é excitada no mesmo ponto, ie, $s = (24, 18)$, enquanto que a função resposta será lida nos n graus de liberdade, ie, $s: (2, 2) \rightarrow (28, 2)$ e $(6, 0) \rightarrow (6, 20)$. Então, da Eq. (35) resulta a magnitude de um elemento da matriz $|u_i u_i^T|_{sr}$, cujo sinal será determinado através do gráfico da fase ϕ em função da frequência ω .

As constantes de amortecimento da viga, relativas a cada uma das frequências naturais para cada um dos

pontos de excitação, podem ser obtidas diretamente do gráfico da FFT da resposta. Tem-se que:

$$\zeta_i = \frac{\omega_{bi} - \omega_{ai}}{2\omega_i}, \omega_{bi} > \omega_i > \omega_{ai} \tag{36}$$

onde ω_i é a i-ésima frequência natural, ω_{bi} é a frequência onde amplitude cai a $\alpha_{sr}(\omega_i)/\sqrt{2}$ e ω_{ai} é a frequência onde a amplitude atinge $\alpha_{sr}(\omega_i)/\sqrt{2}$. As constantes de amortecimento da placa são obtidas de maneira semelhante a partir das FFTs, através dos círculos de Nyquist que se formam no plano complexo em torno de cada uma das frequências naturais.

7. ANÁLISE MODAL EXPERIMENTAL DE UMA VIGA SIMPLEMENTE ENGASTADA

Para a proceder a análise modal experimental de uma viga engastada, foram utilizados os seguintes materiais:

- (a) Uma haste de aço de aproximadamente 60 cm de comprimento, com seção transversal retangular (31,6 mm x 9,5 mm);
- (b) Uma morsa presa a uma bancada;
- (c) Um martelo pequeno;
- (d) Um acelerômetro piezelétrico (CEC 4-274-0001);
- (e) Um amplificador de sinal (Brüer&Kjaer type 2635);
- (f) Uma placa de aquisição de dados (DaqBoard 2000);
- (g) Um microcomputador com programas para aquisição e para tratamento de dados (DaqView 7.8.0 e Matlab R12, respectivamente).

Na montagem do aparato experimental, fixam-se a extremidade inferior da haste à morsa, de modo que se obtenha uma viga de 50 cm, com sua extremidade inferior engastada. O acelerômetro é fixado 15 mm abaixo da extremidade superior da viga, rosqueando-o através de um furo passante central a mesma. A Fig.(6) mostra a montagem final do aparato experimental.

7.1 Cadeia de montagem

A viga é excitada por uma função impulso. Os sinais captados pelo acelerômetro são amplificados e filtrados através do amplificador de cargas, e então enviados para placa de aquisição de dados, onde são digitalizados e armazenados em arquivos no disco rígido do computador através do software de aquisição. Os dados são então tratados num programa específico desenvolvido no Matlab, que fornece as frequências naturais e os modos de vibrar experimentais. Detalhes do programa estão na Seção 10 deste trabalho. Os resultados obtidos são comparados com os resultados analíticos.

7.2 Parâmetros utilizados

Amplificador de carga: (a) Sensibilidade do transdutor: 1,00 pC/m/s² (1 - 11 pC/m/s²); (b) Ganho: 1000 mV/unidade de saída; (c) Unidade de saída: 0,01 m/s; (d) Frequência mínima de corte: 10 Hz; (e) Frequência máxima de corte: 1 kHz.

Software de aquisição (DaqView): (a) Frequência de aquisição: 1100 Hz; (b) Tempo de aquisição: 2 s; (c) Canal selecionado: P1-1 (CH01, bipolar, em V); (d) Modo de disparo: Acima de 0,3 V; (e) Número de aquisições: 30 (3 para cada uma das posições de excitação da viga); (f) Posições de batida: 10 (de 5 em 5 cm, a partir da extremidade livre da viga).

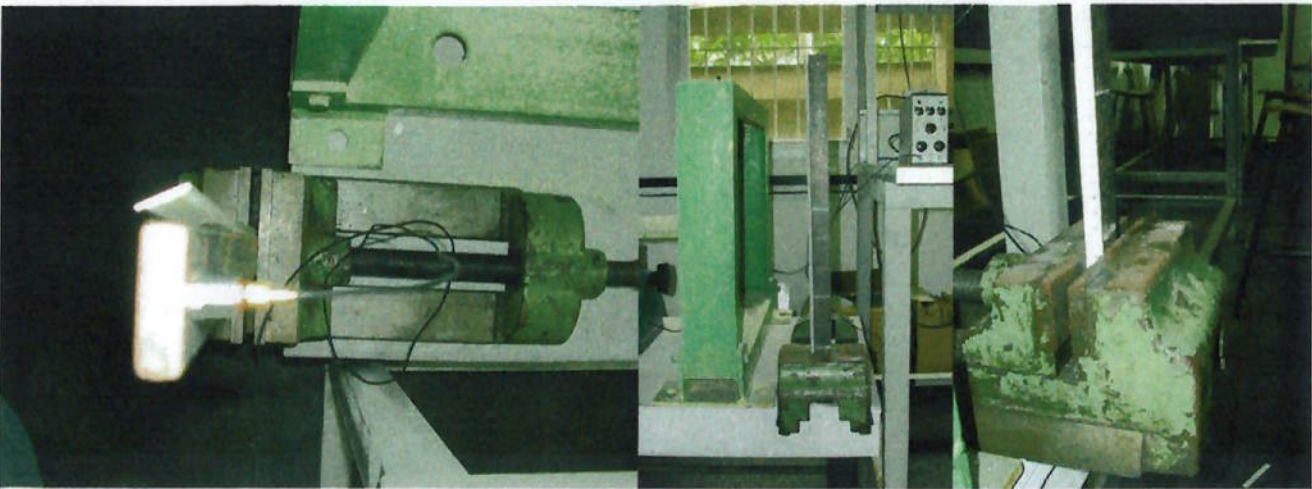


Figura 6. Aparato experimental da viga engastada-livre ensaiada.

7.3 Resultados Experimentais

As frequências naturais da viga engastada obtidas através dos métodos experimental e analítico estão resumidas na Tab. (1). As frequências naturais obtidas experimentalmente encontram-se ligeiramente abaixo do previsto pelo modelo analítico, mas dentro de uma faixa de erro perfeitamente aceitável. Acredita-se que a maior fonte de erro se encontre no engastamento da viga, já que o modelo analítico considera o engastamento perfeito.

Modo de vibrar	Analítico (Hz)	Experimental (Hz)	Diferença (%)
Primeiro	31,7	29,5	-6,9
Segundo	198,8	182,0	-8,4
Terceiro	556,7	507,5	-8,8

Tabela 1. Frequências naturais da viga obtidas pelos métodos analítico e experimental.

A Figura (7a), a FFT mostra os picos referentes às primeiras frequências naturais. Percebe-se que grande parte da energia vai para o primeiro modo, e os demais picos decrescem. Conforme demonstram os gráficos das Fig. (7b) e (7c), a formação do círculo de Nyquist e a mudança em 180° na fase da função de transferência em torno das frequências naturais, comprovam a validade dos resultados obtidos experimentalmente. Os resultados obtidos para os modos de vibrar são comparados nas Figs. (8).

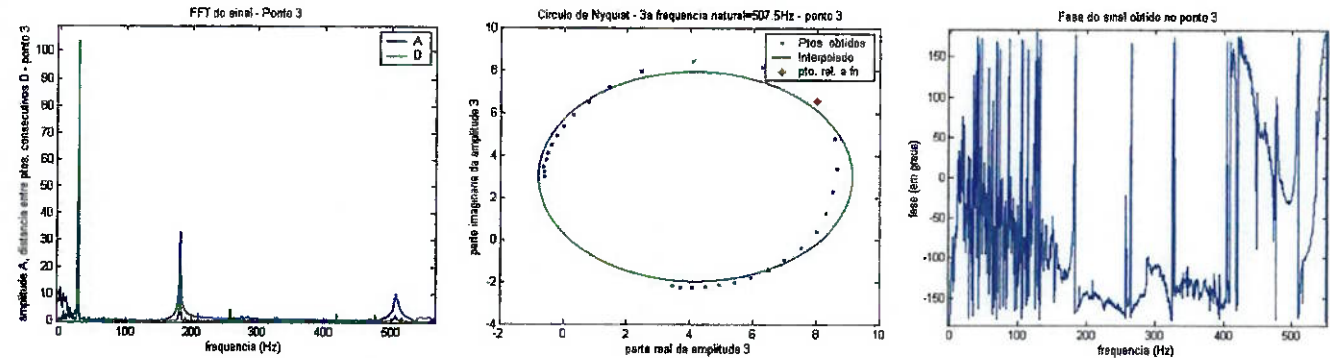


Figura 7. (a) FFT, (b) Círculo de Nyquist e (c) Fase da função de transferência.

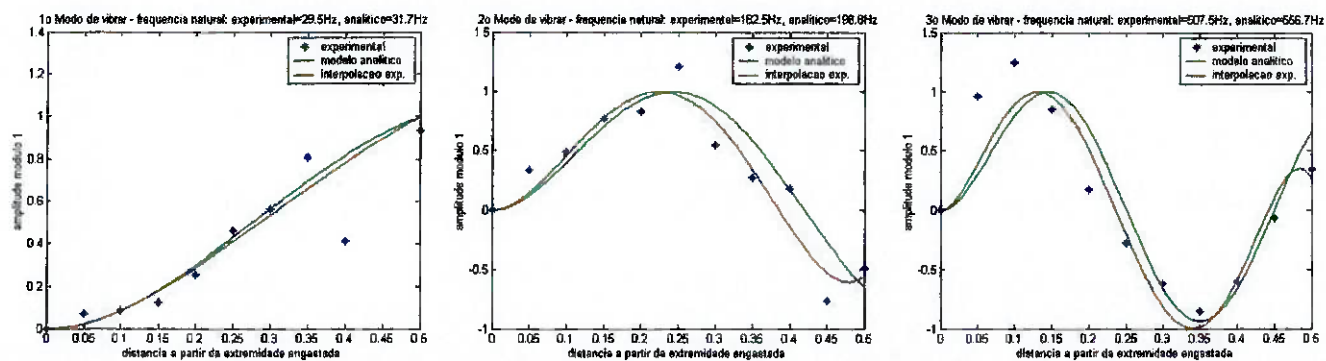


Figura 8. (a) Primeiro, (b) segundo e (c) terceiro modo de vibrar da viga.

8. ANÁLISE MODAL EXPERIMENTAL DE UMA PLACA BIAPOIADA

Para a proceder a análise modal experimental de uma placa biapojada, foram utilizados os seguintes materiais:

- (a) Uma placa retangular de aço (300 mm x 200 mm x 2 mm);
- (b) Um dispositivo para a fixação da placa na configuração desejada;
- (c) Um martelo pequeno;
- (d) Um acelerômetro piezelétrico (B&K 4384);
- (e) Um amplificador de sinal (Brüer&Kjaer type 2635);
- (f) Uma placa de aquisição de dados (DaqBoard 2000);
- (g) Um microcomputador com programas para aquisição e para tratamento de dados (DaqView 7.8.0 e Matlab R12, respectivamente).

Na montagem do aparato experimental, fixam-se dois eixos às bordas menores da placa. Estes eixos apoiam-se em mancais de rolamentos em suas extremidades. Estes mancais são fixados a uma base plana rígida de alumínio, através de parafusos “allen”. O acelerômetro é fixado nas posições apropriadas sobre a placa através de ímas. A Fig.(9) mostra a placa biapojada a ser analisada.

8.1 Cadeia de montagem

A placa é excitada por uma função impulso. Os sinais captados pelo acelerômetro são amplificados e filtrados através do amplificador de cargas, e então enviados para placa de aquisição de dados, onde são digitalizados e armazenados em arquivos no disco rígido do computador através do software de aquisição. Os dados são então tratados num programa específico desenvolvido no Matlab, que fornece as frequências naturais e os modos de vibrar experimentais. Detalhes do programa estão na Seção 10 deste trabalho. Os resultados obtidos são comparados com os resultados analíticos.

8.2 Parâmetros utilizados

Amplificador de carga: (a) Sensibilidade do transdutor: 1,00 pC/m/s² (1 - 11 pC/m/s²); (b) Ganho: 100 mV/unidade de saída; (c) Unidade de saída: 0,01 m/s; (d) Frequência mínima de corte: 10 Hz; (e) Frequência máxima de corte: 10 kHz.

Software de aquisição (DaqView): (a) Frequência de aquisição: 4400 Hz; (b) Tempo de aquisição: 1 s; (c) Canal selecionado: P1-1 (CH01, bipolar, em V); (d) Modo de disparo: Acima de 0,5 V; (e) Número de aquisições: 25; (f) Posições de aquisição: 25 (14 transversais e 11 longitudinais, seguindo uma malha quadriculada de 2 cm x 2 cm).

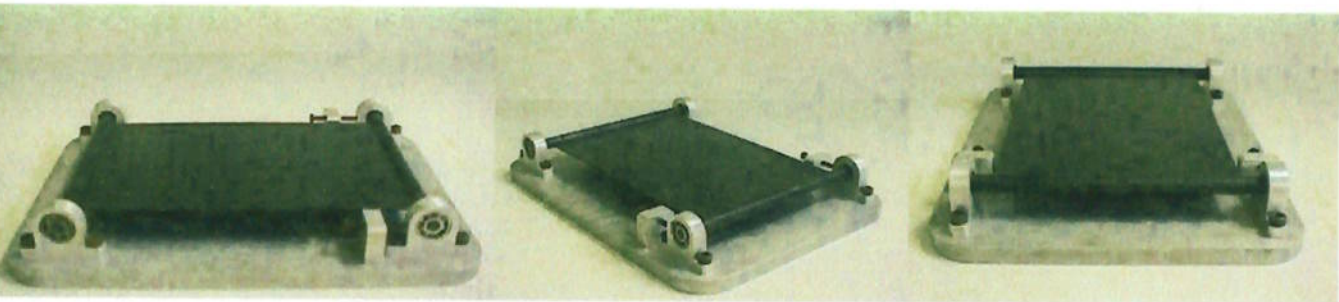


Figura 9. Aparato experimental da placa biapojada ensaiada.

8.3 Resultados experimentais

Os resultados experimentais obtidos para a placa estão agrupados na Tab. (2), juntamente com as respectivas frequências naturais analíticas. As frequências naturais do tipo viga biapoada (01, 02 e 03) encontram-se muito próximas dos resultados analíticos. Já as frequências naturais de placa encontradas (11, 21 e 31), apresentaram discrepâncias maiores em relação ao modelo analítico. Como os modos mais altos necessitam de mais energia, os modos de placa 12, 22 e 32 não foram observados experimentalmente.

A Figura (10) mostra os Círculos de Nyquist do ponto (3,6) da placa em torno das frequências naturais 01, 02 e 03. As Figuras (11) e (12) mostram, respectivamente, os modos de vibrar (01, 02 e 03) e (11, 21 e 31) experimentais e analíticos, nas direções x (embaixo) e y (em cima) da placa. As Figuras (13-18) mostram, lado a lado, os modos de vibrar, experimentais e analíticos, em três dimensões.

Modo de vibrar	Analítico (Hz)	Experimental (Hz)	Diferença (%)
01	52,9	55	4,0
02	214,2	208	-2,9
03	484,8	537	10,8
11	121,1	164	35,4
21	303,5	356	17,3
31	581,8	684	17,6

Tabela 2. Frequências naturais da placa obtidas pelos métodos analítico e experimental.

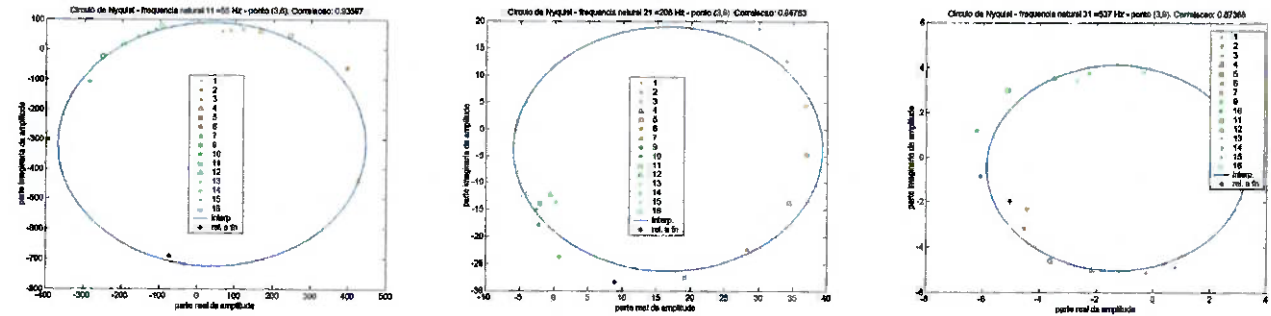


Figura 10. Círculos de Nyquist em torno das frequências naturais 01, 02 e 03.

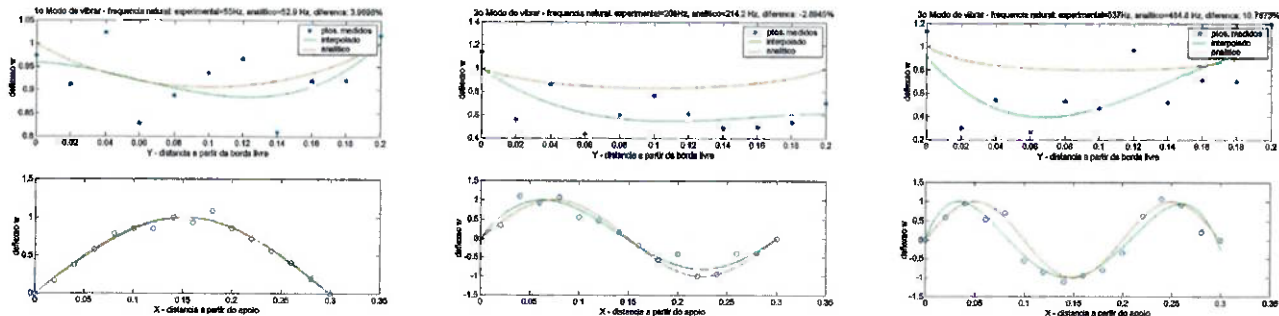


Figura 11. Modos de vibrar 01, 02 e 03, experimentais e analíticos.

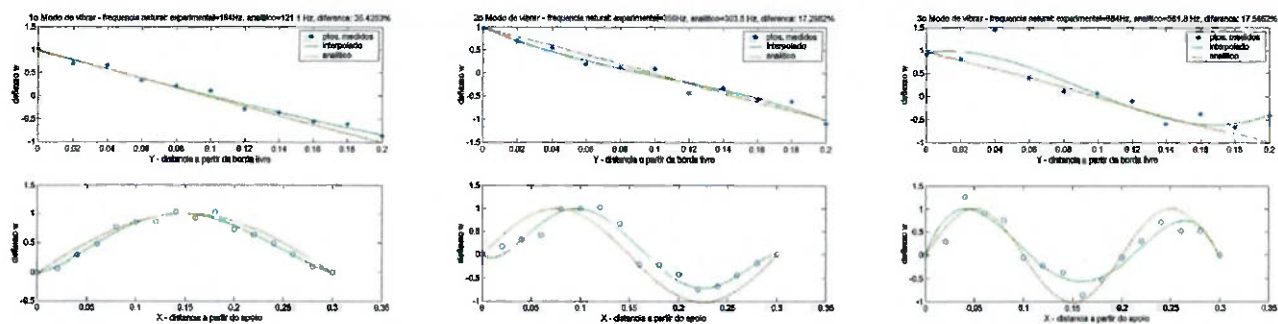


Figura 12. Modos de vibrar 11, 21 e 31, experimentais e analíticos.

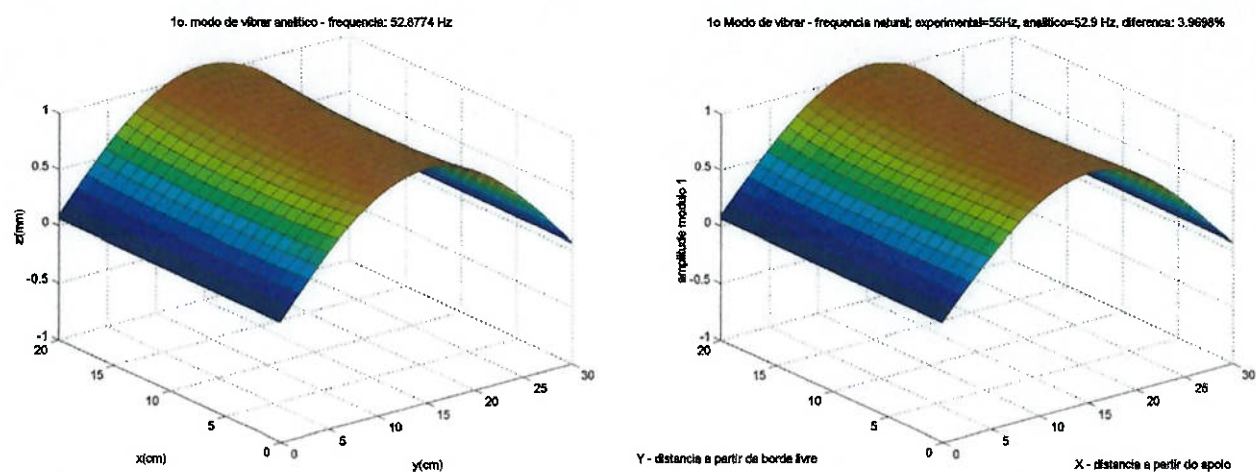


Figura 13. Modo de vibrar 01 analítico e experimental.

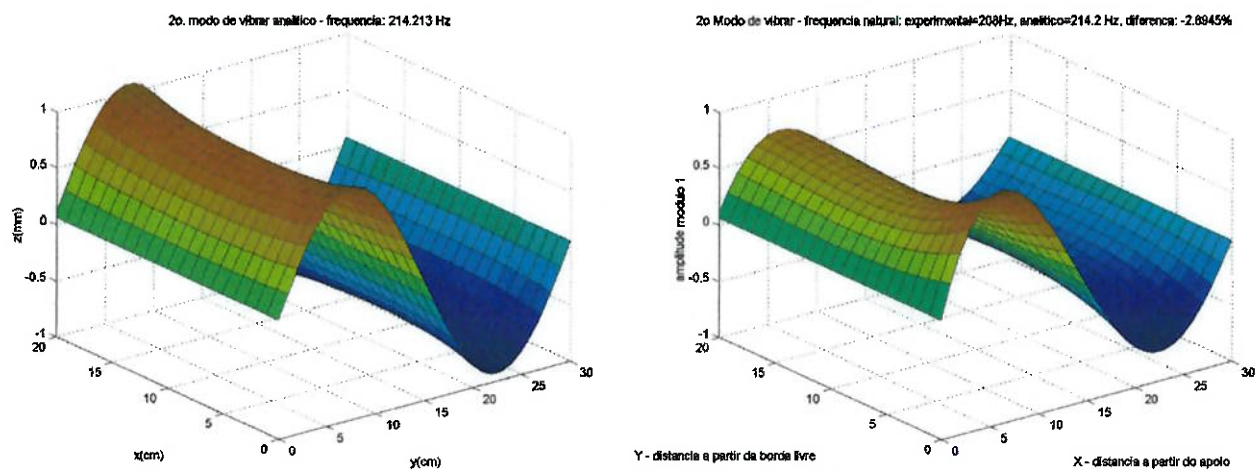


Figura 14. Modo de vibrar 02 analítico e experimental.

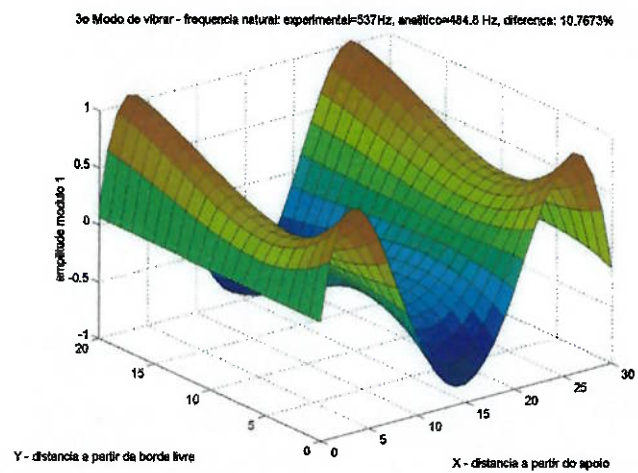
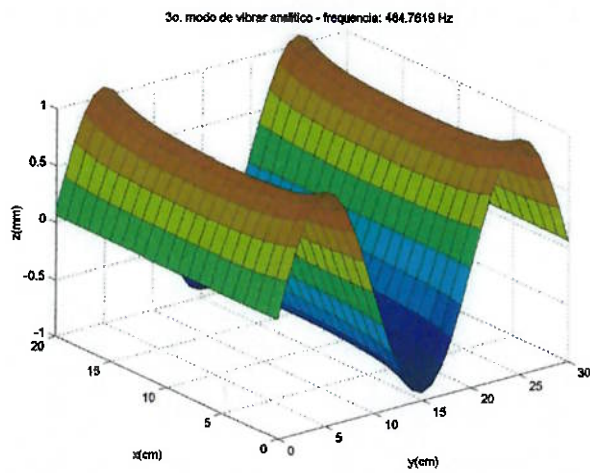


Figura 15. Modo de vibrar 03 analítico e experimental.

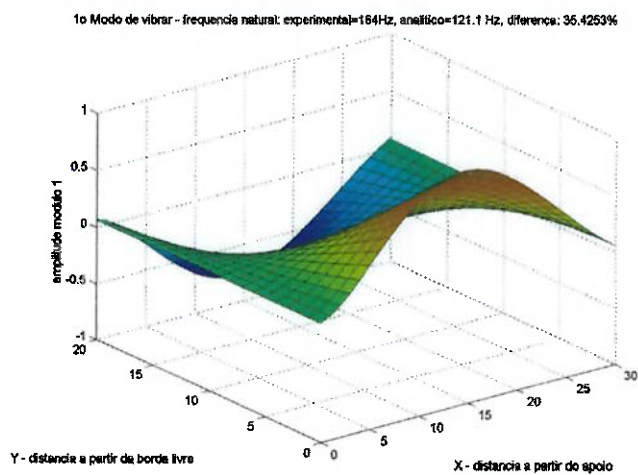
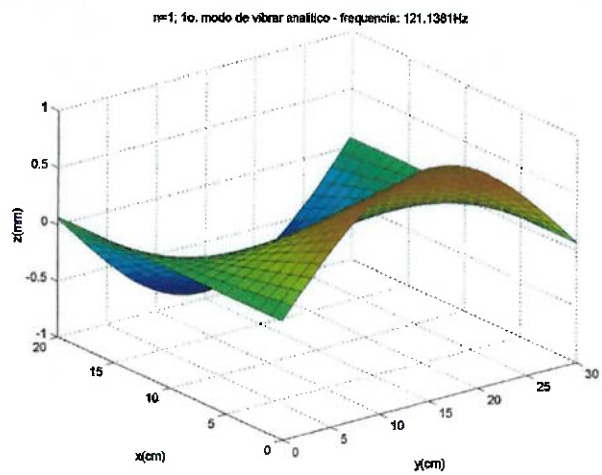


Figura 16. Modo de vibrar 11 analítico e experimental.

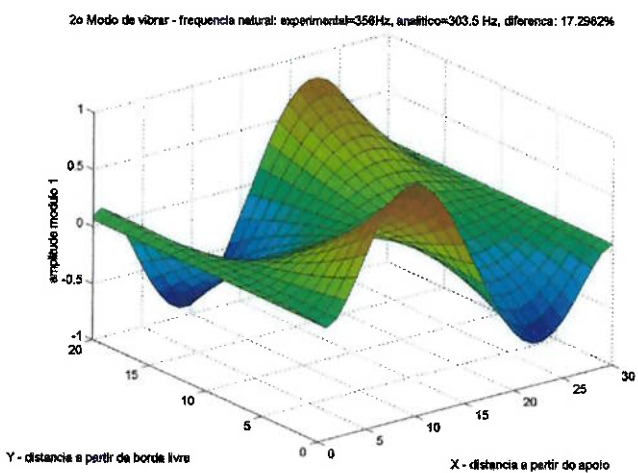
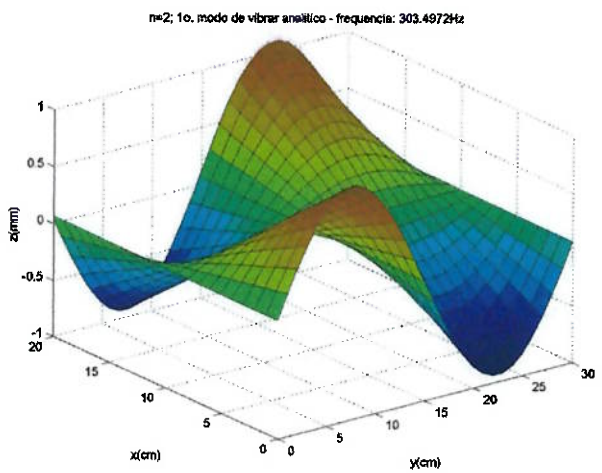


Figura 17. Modo de vibrar 21 analítico e experimental.

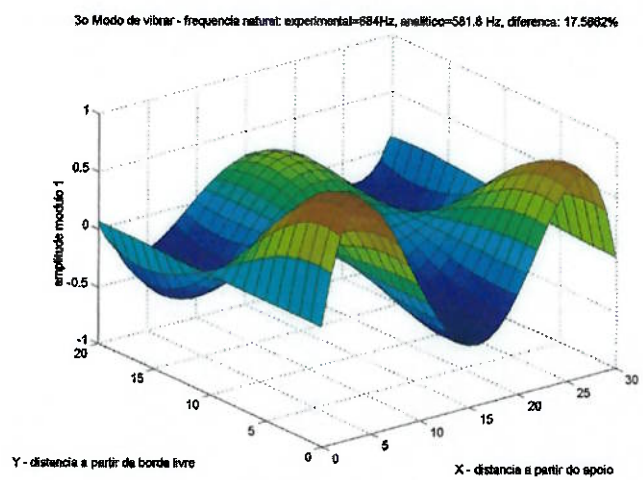
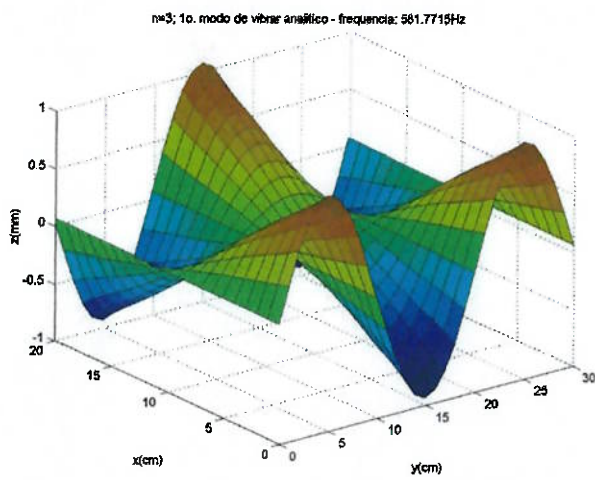


Figura 18. Modo de vibrar 31 analítico e experimental.

9. DISCUSSÃO

Para o caso da viga, pode-se verificar a validade dos resultados obtidos experimentalmente através da comparação com as frequências naturais e os modos de vibrar provenientes do modelo analítico, com erros inferiores a 10%, o que é perfeitamente aceitável, posto que o modelo analítico assume que o engastamento é perfeito.

No caso da placa, as frequências naturais relativas aos modos de viga biapoiada apresentaram pequenas diferenças entre os resultados teórico e experimental, inferiores a 15%. Já as frequências naturais dos modos exclusivamente de placa, estas diferenças chegaram a até 35%. Acredita-se que estas diferenças devem-se aos eixos que apoiam as extremidades da placa, aumentando a rigidez à flexão e a torção. As frequências naturais e os modos obtidos foram verificados na prática, utilizando-se um shaker ligado a um gerador de frequência para excitar a placa. Com isso, estabelecemos um grau de confiabilidade sobre o método da análise modal experimental, permitindo estender sua aplicação a estruturas mais complexas.

10. BIBLIOGRAFIA

- Chen, H-P., Bicanic, N., 2000, "Assessment of damage in continuum structures based on incomplete modal Information", *Computers and Structures*, Vol 74, pp. 559-570.
- Dimaragonas, A., 1996. "Vibration for Engineers", 2^a edição, Prentice Hall.
- Hu, N., Wang, X., Fukunaga, H., Yao, Z. H., Zhang, H. X., Wu, Z. S., 2001, "Damage assesment of Structures using modal test data", *International Journal of Solids and Structures*, Vol 38, pp. 3111-3126.
- Inman, D. J., 2000, "Engineering vibration", 2^a edição, Prentice Hall.
- Khan, A. Z., Stanbridge, A. B., Ewins, D. J., 2000, "Detecting damage in vibrating structures with a scanning LDV", *Optics and Lasers in Engineering*, Vol 32, pp. 583-592.
- Kim, H. M., Bartkowicz, T. J., 2001, "An experimental study for damage detection using a hexagonal truss", *Computers and Structures*, Vol 79, pp. 173-182.
- Li, G-Q., Hao, K-C., Lu, Y., Chen, S. W., 1999, "A flexibility approach for damage identification of cantilever-type structures with bending and shear deformation", *Computers and Structures*, Vol 73, pp. 565-572.
- Wang, X., Hu, N., Fukunaga, H. and Yao, Z. H., 2001, "Structural damage identification using static test data and changes in frequencies", *Engineering Structures*, Vol. 23, pp. 610-621.

11. APÊNDICES

Programa para ajuste de um conjunto de pontos por uma circunferência (Nyquist.m), através do método dos mínimos quadrados:

```
function Nyquist(x,y)
A=[sum(x.*x) sum(x.*y) sum(x); sum(x.*y) sum(y.*y) sum(y); sum(x) sum(y) length(x)];
C=[-sum(x.*x.*x)-sum(x.*y.*y); -sum(x.*x.*y)-sum(y.*y.*y); -sum(x.*x)-sum(y.*y)];
B=pinv(A)*C;
global x0 y0 raio rq
x0=-B(1)/2;
y0=-B(2)/2;
raio=(x0^2+y0^2-B(3)).^.5;
rq=(1-sum(abs(1-(sqrt((x-x0).^2+(y-y0).^2)./raio)))/length(x))^2;
```

Programa para análise modal da viga engastada estudada (Viga.m):

```
% Modulo 1: Aquisicao de dados:

clear all
clc
global x0 y0 raio % variaveis globais usadas na funcao nyquist
l=0.5; % Comprimento da barra, em metros
fa=1100; % Frequencia de aquisicao de dados, em Hz
xbatida=[0:0.05:0.5]; % Posicao das batidas com o martelo, medidas a partir da
extremidade engastada, em metros
na=2200; % Numero de dados (pontos) aquisitados
t=[1:1:na]; % Vetor tempo de aquisicao de dados
w=[fa/na:fa/na:fa]; % Vetor frequencia varrida pelo sinal
x=[0:0.005:0.5]; % Vetor posicao na viga
Beta1=3.75021;
Beta2=9.38818;
Beta3=15.70951;
a4=1;
a31=-a4*((cos(Beta1*.5)+cosh(Beta1*.5))/(sin(Beta1*.5)+sinh(Beta1*.5)));
a32=-a4*((cos(Beta2*.5)+cosh(Beta2*.5))/(sin(Beta2*.5)+sinh(Beta2*.5)));
a33=-a4*((cos(Beta3*.5)+cosh(Beta3*.5))/(sin(Beta3*.5)+sinh(Beta3*.5)));
X1=-a31*sin(Beta1*x)-a4*cos(Beta1*x)+a31*sinh(Beta1*x)+a4*cosh(Beta1*x); % modo
de vibrar da 1a freq. natural (analitico)
X2=-a32*sin(Beta2*x)-a4*cos(Beta2*x)+a32*sinh(Beta2*x)+a4*cosh(Beta2*x); % modo
de vibrar da 2a freq. natural (analitico)
X3=-a33*sin(Beta3*x)-a4*cos(Beta3*x)+a33*sinh(Beta3*x)+a4*cosh(Beta3*x); % modo
de vibrar da 3a freq. natural (analitico)
load c:\matlabr11\1_1 % Aquisicao de dados do acelerometro, ponto 1
ch1_1=A;
load c:\matlabr11\1_2 % Aquisicao de dados do acelerometro, ponto 1
ch1_2=A;
load c:\matlabr11\1_3 % Aquisicao de dados do acelerometro, ponto 1
ch1_3=A;
load c:\matlabr11\2_1 % Aquisicao de dados do acelerometro, ponto 2
ch2_1=A;
load c:\matlabr11\2_2 % Aquisicao de dados do acelerometro, ponto 2
ch2_2=A;
load c:\matlabr11\2_3 % Aquisicao de dados do acelerometro, ponto 2
ch2_3=A;
load c:\matlabr11\3_1 % Aquisicao de dados do acelerometro, ponto 3
ch3_1=A;
load c:\matlabr11\3_2 % Aquisicao de dados do acelerometro, ponto 3
ch3_2=A;
load c:\matlabr11\3_3 % Aquisicao de dados do acelerometro, ponto 3
ch3_3=A;
load c:\matlabr11\4_1 % Aquisicao de dados do acelerometro, ponto 4
ch4_1=A;
load c:\matlabr11\4_2 % Aquisicao de dados do acelerometro, ponto 4
ch4_2=A;
load c:\matlabr11\4_3 % Aquisicao de dados do acelerometro, ponto 4
ch4_3=A;
load c:\matlabr11\5_1 % Aquisicao de dados do acelerometro, ponto 5
ch5_1=A;
load c:\matlabr11\5_2 % Aquisicao de dados do acelerometro, ponto 5
ch5_2=A;
```

```

load c:\matlabr11\5_3 % Aquisicao de dados do acelerometro, ponto 5
ch5_3=A;
load c:\matlabr11\6_1 % Aquisicao de dados do acelerometro, ponto 6
ch6_1=A;
load c:\matlabr11\6_2 % Aquisicao de dados do acelerometro, ponto 6
ch6_2=A;
load c:\matlabr11\6_3 % Aquisicao de dados do acelerometro, ponto 6
ch6_3=A;
load c:\matlabr11\7_1 % Aquisicao de dados do acelerometro, ponto 7
ch7_1=A;
load c:\matlabr11\7_2 % Aquisicao de dados do acelerometro, ponto 7
ch7_2=A;
load c:\matlabr11\7_3 % Aquisicao de dados do acelerometro, ponto 7
ch7_3=A;
load c:\matlabr11\8_1 % Aquisicao de dados do acelerometro, ponto 8
ch8_1=A;
load c:\matlabr11\8_2 % Aquisicao de dados do acelerometro, ponto 8
ch8_2=A;
load c:\matlabr11\8_3 % Aquisicao de dados do acelerometro, ponto 8
ch8_3=A;
load c:\matlabr11\9_1 % Aquisicao de dados do acelerometro, ponto 9
ch9_1=A;
load c:\matlabr11\9_2 % Aquisicao de dados do acelerometro, ponto 9
ch9_2=A;
load c:\matlabr11\9_3 % Aquisicao de dados do acelerometro, ponto 9
ch9_3=A;
load c:\matlabr11\10_1 % Aquisicao de dados do acelerometro, ponto 10
ch10_1=A;
load c:\matlabr11\10_2 % Aquisicao de dados do acelerometro, ponto 10
ch10_2=A;
load c:\matlabr11\10_3 % Aquisicao de dados do acelerometro, ponto 10
ch10_3=A;
clear A; % Apagar a variavel de auxilio A
ch1soma=ch1_1+ch1_2+ch1_3; % Soma dos dados no ponto 1 (xbatida=0.05)
ch2soma=ch2_1+ch2_2+ch2_3; % Soma dos dados no ponto 2 (xbatida=0.1)
ch3soma=ch3_1+ch3_2+ch3_3; % Soma dos dados no ponto 3 (xbatida=0.15)
ch4soma=ch4_1+ch4_2+ch4_3; % Soma dos dados no ponto 1 (xbatida=0.2)
ch5soma=ch5_1+ch5_2+ch5_3; % Soma dos dados no ponto 2 (xbatida=0.25)
ch6soma=ch6_1+ch6_2+ch6_3; % Soma dos dados no ponto 3 (xbatida=0.3)
ch7soma=ch7_1+ch7_2+ch7_3; % Soma dos dados no ponto 1 (xbatida=0.35)
ch8soma=ch8_1+ch8_2+ch8_3; % Soma dos dados no ponto 2 (xbatida=0.4)
ch9soma=ch9_1+ch9_2+ch9_3; % Soma dos dados no ponto 3 (xbatida=0.45)
ch10soma=ch10_1+ch10_2+ch10_3; % Soma dos dados no ponto 1 (xbatida=0.5)

```

% Modulo 2: Tratamento de dados:

% Transformada de Fourier: passa a amplitude do sinal para o espectro de frequencia

```

s1soma=fft(ch1soma); % amplitude em funcao da frequencia: s1soma
r1soma=real(s1soma); % parte real de s1soma
i1soma=imag(s1soma); % parte imaginaria de s1soma
s2soma=fft(ch2soma); % amplitude em funcao da frequencia: s2soma
r2soma=real(s2soma); % parte real de s2soma
i2soma=imag(s2soma); % parte imaginaria de s2soma
s3soma=fft(ch3soma); % amplitude em funcao da frequencia: s3soma
r3soma=real(s3soma); % parte real de s3soma

```

```

is3soma=imag(s3soma); % parte imaginaria de s3soma
s4soma=fft(ch4soma); % amplitude em funcao da frequencia: s4soma
rs4soma=real(s4soma); % parte real de s4soma
is4soma=imag(s4soma); % parte imaginaria de s4soma
s5soma=fft(ch5soma); % amplitude em funcao da frequencia: s5soma
rs5soma=real(s5soma); % parte real de s5soma
is5soma=imag(s5soma); % parte imaginaria de s5soma
s6soma=fft(ch6soma); % amplitude em funcao da frequencia: s6soma
rs6soma=real(s6soma); % parte real de s6soma
is6soma=imag(s6soma); % parte imaginaria de s6soma
s7soma=fft(ch7soma); % amplitude em funcao da frequencia: s7soma
rs7soma=real(s7soma); % parte real de s7soma
is7soma=imag(s7soma); % parte imaginaria de s7soma
s8soma=fft(ch8soma); % amplitude em funcao da frequencia: s8soma
rs8soma=real(s8soma); % parte real de s8soma
is8soma=imag(s8soma); % parte imaginaria de s8soma
s9soma=fft(ch9soma); % amplitude em funcao da frequencia: s9soma
rs9soma=real(s9soma); % parte real de s9soma
is9soma=imag(s9soma); % parte imaginaria de s9soma
s10soma=fft(ch10soma); % amplitude em funcao da frequencia: s10soma
rs10soma=real(s10soma); % parte real de s10soma
is10soma=imag(s10soma); % parte imaginaria de s10soma
for n=1:na-1
    dq1(n)=sqrt((rs1soma(n+1)-rs1soma(n))^2+(is1soma(n+1)-is1soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 1
    dq2(n)=sqrt((rs2soma(n+1)-rs2soma(n))^2+(is2soma(n+1)-is2soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 2
    dq3(n)=sqrt((rs3soma(n+1)-rs3soma(n))^2+(is3soma(n+1)-is3soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 3
    dq4(n)=sqrt((rs4soma(n+1)-rs4soma(n))^2+(is4soma(n+1)-is4soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 4
    dq5(n)=sqrt((rs5soma(n+1)-rs5soma(n))^2+(is5soma(n+1)-is5soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 5
    dq6(n)=sqrt((rs6soma(n+1)-rs6soma(n))^2+(is6soma(n+1)-is6soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 6
    dq7(n)=sqrt((rs7soma(n+1)-rs7soma(n))^2+(is7soma(n+1)-is7soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 7
    dq8(n)=sqrt((rs8soma(n+1)-rs8soma(n))^2+(is8soma(n+1)-is8soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 8
    dq9(n)=sqrt((rs9soma(n+1)-rs9soma(n))^2+(is9soma(n+1)-is9soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 9
    dq10(n)=sqrt((rs10soma(n+1)-rs10soma(n))^2+(is10soma(n+1)-is10soma(n))^2); %
    distancia entre 2 pontos consecutivos, ponto 10
end
for n=1:10
    dq1(n)=0;
    dq2(n)=0;
    dq3(n)=0;
    dq4(n)=0;
    dq5(n)=0;
    dq6(n)=0;
    dq7(n)=0;
    dq8(n)=0;
    dq9(n)=0;
    dq10(n)=0;
end
for n=na-9:na

```

```

dq1(n)=0;
dq2(n)=0;
dq3(n)=0;
dq4(n)=0;
dq5(n)=0;
dq6(n)=0;
dq7(n)=0;
dq8(n)=0;
dq9(n)=0;
dq10(n)=0;

```

```
end
```

```
% Modulo 3: Saida grafica da amplitude do sinal em funcao da frequencia:
```

```

figure(1)
plot(w,abs(s1soma),w,dq1)
title(['FFT do sinal - Ponto 1']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 1')
figure(2)
plot(w,abs(s2soma),w,dq2)
title(['FFT do sinal - Ponto 2']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 2')
figure(3)
plot(w,abs(s3soma),w,dq3)
title(['FFT do sinal - Ponto 3']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 3')
figure(4)
plot(w,abs(s4soma),w,dq4)
title(['FFT do sinal - Ponto 4']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 4')
figure(5)
plot(w,abs(s5soma),w,dq5)
title(['FFT do sinal - Ponto 5']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 5')
figure(6)
plot(w,abs(s6soma),w,dq6)
title(['FFT do sinal - Ponto 6']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 6')
figure(7)
plot(w,abs(s7soma),w,dq7)
title(['FFT do sinal - Ponto 7']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 7')
figure(8)

```

```

plot(w,abs(s8soma),w,dq8)
title(['FFT do sinal - Ponto 8']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 8')
figure(9)
plot(w,abs(s9soma),w,dq9)
title(['FFT do sinal - Ponto 9']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 9')
figure(10)
plot(w,abs(s10soma),w,dq10)
title(['FFT do sinal - Ponto 10']);
legend('A','D');
xlabel('frequencia (Hz)');
ylabel('amplitude A, distancia entre ptos. consecutivos D - ponto 10')

% Modulo 4: Pre-selecao das frequencias candidatas a frequencia natural:
for n=1:na
    if(abs(s1soma(n))>5)
        as1(n)=abs(s1soma(n));
    else
        as1(n)=0;
    end
    if(abs(s2soma(n))>10)
        as2(n)=abs(s2soma(n));
    else
        as2(n)=0;
    end
    if(abs(s3soma(n))>10)
        as3(n)=abs(s3soma(n));
    else
        as3(n)=0;
    end
    if(abs(s4soma(n))>5)
        as4(n)=abs(s4soma(n));
    else
        as4(n)=0;
    end
    if(abs(s5soma(n))>2)
        as5(n)=abs(s5soma(n));
    else
        as5(n)=0;
    end
    if(abs(s6soma(n))>7)
        as6(n)=abs(s6soma(n));
    else
        as6(n)=0;
    end
    if(abs(s7soma(n))>5)
        as7(n)=abs(s7soma(n));
    else
        as7(n)=0;
    end
    if(abs(s8soma(n))>3)

```

```

        as8(n)=abs(s8soma(n));
    else
        as8(n)=0;
    end
    if(abs(s9soma(n))>1)
        as9(n)=abs(s9soma(n));
    else
        as9(n)=0;
    end
    if(abs(s10soma(n))>5)
        as10(n)=abs(s10soma(n));
    else
        as10(n)=0;
    end
end
as1(2)=0;
as1(na)=0;
as2(2)=0;
as2(na)=0;
as3(2)=0;
as3(na)=0;
as4(2)=0;
as4(na)=0;
as5(2)=0;
as5(na)=0;
as6(2)=0;
as6(na)=0;
as7(2)=0;
as7(na)=0;
as8(2)=0;
as8(na)=0;
as9(2)=0;
as9(na)=0;
as10(2)=0;
as10(na)=0;

% detector de frecuencias de pico
fpicos=[];
for n=3:(na/2)+2
    if(as1(n)>as1(n-1))
        if(as1(n)>as1(n+1))
            if(as2(n)>as2(n-1)|as2(n-1)>as2(n-2)|as2(n+1)>as2(n))
                if(as2(n)>as2(n+1)|as2(n-1)>as2(n)|as2(n+1)>as2(n+2))
                    if(as3(n)>as3(n-1)|as3(n-1)>as3(n-2)|as3(n+1)>as3(n))
                        if(as3(n)>as3(n+1)|as3(n-1)>as3(n)|as3(n+1)>as3(n+2))
                            if(as4(n)>as4(n-1)|as4(n-1)>as4(n-
2)|as4(n+1)>as4(n))
                                if(as4(n)>as4(n+1)|as4(n-
1)>as4(n)|as4(n+1)>as4(n+2))
                                    if(as5(n)>as5(n-1)|as5(n-1)>as5(n-
2)|as5(n+1)>as5(n))
                                        if(as5(n)>as5(n+1)|as5(n-
1)>as5(n)|as5(n+1)>as5(n+2))
                                            if(as6(n)>as6(n-1)|as6(n-1)>as6(n-
2)|as6(n+1)>as6(n))
                                                if(as6(n)>as6(n+1)|as6(n-
1)>as6(n)|as6(n+1)>as6(n+2))

```



```

        rs5soma1(i)=rs5soma(n-15+i);
        is5soma1(i)=is5soma(n-15+i);
        rs6soma1(i)=rs6soma(n-15+i);
        is6soma1(i)=is6soma(n-15+i);
        rs7soma1(i)=rs7soma(n-15+i);
        is7soma1(i)=is7soma(n-15+i);
        rs8soma1(i)=rs8soma(n-15+i);
        is8soma1(i)=is8soma(n-15+i);
        rs9soma1(i)=rs9soma(n-15+i);
        is9soma1(i)=is9soma(n-15+i);
        rs10soma1(i)=rs10soma(n-15+i);
        is10soma1(i)=is10soma(n-15+i);
    end
    fpicol=n*fa/na;
end
end
end
end

% pontos do circulo de Niquist - 2a. frequencia natural
for n=3:na/2
    if(as2(n)+.00001>max(as2))
        for i=1:30
            rs1soma2(i)=rs1soma(n-15+i);
            is1soma2(i)=is1soma(n-15+i);
            rs2soma2(i)=rs2soma(n-15+i);
            is2soma2(i)=is2soma(n-15+i);
            rs3soma2(i)=rs3soma(n-15+i);
            is3soma2(i)=is3soma(n-15+i);
            rs4soma2(i)=rs4soma(n-15+i);
            is4soma2(i)=is4soma(n-15+i);
            rs5soma2(i)=rs5soma(n-15+i);
            is5soma2(i)=is5soma(n-15+i);
            rs6soma2(i)=rs6soma(n-15+i);
            is6soma2(i)=is6soma(n-15+i);
            rs7soma2(i)=rs7soma(n-15+i);
            is7soma2(i)=is7soma(n-15+i);
            rs8soma2(i)=rs8soma(n-15+i);
            is8soma2(i)=is8soma(n-15+i);
            rs9soma2(i)=rs9soma(n-15+i);
            is9soma2(i)=is9soma(n-15+i);
            rs10soma2(i)=rs10soma(n-15+i);
            is10soma2(i)=is10soma(n-15+i);
        end
        fpico2=n*fa/na;
    end
end

% pontos do circulo de Niquist - 3a. frequencia natural
for n=fpico2*2*na/fa:na/2
    aux1(n+1-fpico2*2*na/fa)=as1(n);
    aux2(n+1-fpico2*2*na/fa)=as2(n);
    aux3(n+1-fpico2*2*na/fa)=as3(n);
end
for n=fpico2*2*na/fa:na/2
    if(as1(n)==max(aux1))
        if(as2(n)==max(aux2))

```

```

    if(as3(n)==max(aux3))
        for i=1:30
            rs1soma3(i)=rs1soma(n-15+i);
            is1soma3(i)=is1soma(n-15+i);
            rs2soma3(i)=rs2soma(n-15+i);
            is2soma3(i)=is2soma(n-15+i);
            rs3soma3(i)=rs3soma(n-15+i);
            is3soma3(i)=is3soma(n-15+i);
            rs4soma3(i)=rs4soma(n-15+i);
            is4soma3(i)=is4soma(n-15+i);
            rs5soma3(i)=rs5soma(n-15+i);
            is5soma3(i)=is5soma(n-15+i);
            rs6soma3(i)=rs6soma(n-15+i);
            is6soma3(i)=is6soma(n-15+i);
            rs7soma3(i)=rs7soma(n-15+i);
            is7soma3(i)=is7soma(n-15+i);
            rs8soma3(i)=rs8soma(n-15+i);
            is8soma3(i)=is8soma(n-15+i);
            rs9soma3(i)=rs9soma(n-15+i);
            is9soma3(i)=is9soma(n-15+i);
            rs10soma3(i)=rs10soma(n-15+i);
            is10soma3(i)=is10soma(n-15+i);
        end
        fpico3=n*fa/na;
    end
end
end
end
clear aux1;
clear aux2;
clear aux3;
wn=2*pi*[fpicol fpico2 fpico3];

Modulo 6: Saida grafica dos circulos de Nyquist em torno das freqs naturais:

nyquist(rs3soma1,is3soma1);
figure(11)
plot(rs3soma1,is3soma1,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs3soma
(fpicol*na/fa),is3soma(fpicol*na/fa),'*')
title(['Circulo de Nyquist - 1a frecuencia natural=',num2str(fpicol),'Hz - ponto
3']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 3');
ylabel('parte imaginaria da amplitude 3');
nyquist(rs6soma1,is6soma1);
figure(12)
plot(rs6soma1,is6soma1,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs6soma
(fpicol*na/fa),is6soma(fpicol*na/fa),'*')
title(['Circulo de Nyquist - 1a frecuencia natural=',num2str(fpicol),'Hz - ponto
6']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 6');
ylabel('parte imaginaria da amplitude 6');
nyquist(rs10soma1,is10soma1);
figure(13)
plot(rs10soma1,is10soma1,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs10s
oma(fpicol*na/fa),is10soma(fpicol*na/fa),'*')

```

```

title(['Circulo de Nyquist - 1a frequencia natural=',num2str(fpico1),'Hz - ponto
10']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 10');
ylabel('parte imaginaria da amplitude 10');
nyquist(rs3soma2,is3soma2);
figure(14)
plot(rs3soma2,is3soma2,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs3soma
(fpico2*na/fa),is3soma(fpico2*na/fa),'*')
title(['Circulo de Nyquist - 2a frequencia natural=',num2str(fpico2),'Hz - ponto
3']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 3');
ylabel('parte imaginaria da amplitude 3');
nyquist(rs6soma2,is6soma2);
figure(15)
plot(rs6soma2,is6soma2,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs6soma
(fpico2*na/fa),is6soma(fpico2*na/fa),'*')
title(['Circulo de Nyquist - 2a frequencia natural=',num2str(fpico2),'Hz - ponto
6']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 6');
ylabel('parte imaginaria da amplitude 6');
nyquist(rs10soma2,is10soma2);
figure(16)
plot(rs10soma2,is10soma2,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs10s
oma(fpico2*na/fa),is10soma(fpico2*na/fa),'*')
title(['Circulo de Nyquist - 2a frequencia natural=',num2str(fpico2),'Hz - ponto
10']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 10');
ylabel('parte imaginaria da amplitude 10');
nyquist(rs3soma3,is3soma3);
figure(17)
plot(rs3soma3,is3soma3,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs3soma
(fpico3*na/fa),is3soma(fpico3*na/fa),'*')
title(['Circulo de Nyquist - 3a frequencia natural=',num2str(fpico3),'Hz - ponto
3']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 3');
ylabel('parte imaginaria da amplitude 3');
nyquist(rs6soma3,is6soma3);
figure(18)
plot(rs6soma3,is6soma3,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs6soma
(fpico3*na/fa),is6soma(fpico3*na/fa),'*')
title(['Circulo de Nyquist - 3a frequencia natural=',num2str(fpico3),'Hz - ponto
5']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');
xlabel('parte real da amplitude 6');
ylabel('parte imaginaria da amplitude 6');
nyquist(rs10soma3,is10soma3);
figure(19)
plot(rs10soma3,is10soma3,'.',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),rs10s
oma(fpico3*na/fa),is10soma(fpico3*na/fa),'*')
title(['Circulo de Nyquist - 3a frequencia natural=',num2str(fpico3),'Hz - ponto
10']);
legend('Ptos. obtidos','Interpolado','pto. rel. a fn');

```

```

xlabel('parte real da amplitude 10');
ylabel('parte imaginaria da amplitude 10');

% Modulo 7: Determinacao das constantes de amortecimento (damping ratio: ksi)

for n=fpicol*na/fa-10:fpicol*na/fa+10
    if(as1(n)<as1(fpicol*na/fa)/sqrt(2))
        if(as1(n+1)>as1(fpicol*na/fa)/sqrt(2))
            wa1=n*fa/na+(as1(fpicol*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as1(n)>as1(fpicol*na/fa)/sqrt(2))
        if(as1(n+1)<as1(fpicol*na/fa)/sqrt(2))
            wb1=n*fa/na+(as1(fpicol*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as2(n)<as2(fpicol*na/fa)/sqrt(2))
        if(as2(n+1)>as2(fpicol*na/fa)/sqrt(2))
            wa2=n*fa/na+(as2(fpicol*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as2(n)>as2(fpicol*na/fa)/sqrt(2))
        if(as2(n+1)<as2(fpicol*na/fa)/sqrt(2))
            wb2=n*fa/na+(as2(fpicol*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as3(n)<as3(fpicol*na/fa)/sqrt(2))
        if(as3(n+1)>as3(fpicol*na/fa)/sqrt(2))
            wa3=n*fa/na+(as3(fpicol*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
    if(as3(n)>as3(fpicol*na/fa)/sqrt(2))
        if(as3(n+1)<as3(fpicol*na/fa)/sqrt(2))
            wb3=n*fa/na+(as3(fpicol*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
    if(as4(n)<as4(fpicol*na/fa)/sqrt(2))
        if(as4(n+1)>as4(fpicol*na/fa)/sqrt(2))
            wa4=n*fa/na+(as4(fpicol*na/fa)/sqrt(2)-as4(n))*fa/(na*(as4(n+1)-
as4(n)));
        end
    end
    if(as4(n)>as4(fpicol*na/fa)/sqrt(2))
        if(as4(n+1)<as4(fpicol*na/fa)/sqrt(2))
            wb4=n*fa/na+(as4(fpicol*na/fa)/sqrt(2)-as4(n))*fa/(na*(as4(n+1)-
as4(n)));
        end
    end
    if(as5(n)<as5(fpicol*na/fa)/sqrt(2))
        if(as5(n+1)>as5(fpicol*na/fa)/sqrt(2))

```

```

        wa5=n*fa/na+(as5(fpicol*na/fa)/sqrt(2)-as5(n))*fa/(na*(as5(n+1)-
as5(n)));
    end
end
if(as5(n)>as5(fpicol*na/fa)/sqrt(2))
    if(as5(n+1)<as5(fpicol*na/fa)/sqrt(2))
        wb5=n*fa/na+(as5(fpicol*na/fa)/sqrt(2)-as5(n))*fa/(na*(as5(n+1)-
as5(n)));
    end
end
if(as6(n)<as6(fpicol*na/fa)/sqrt(2))
    if(as6(n+1)>as6(fpicol*na/fa)/sqrt(2))
        wa6=n*fa/na+(as6(fpicol*na/fa)/sqrt(2)-as6(n))*fa/(na*(as6(n+1)-
as6(n)));
    end
end
if(as6(n)>as6(fpicol*na/fa)/sqrt(2))
    if(as6(n+1)<as6(fpicol*na/fa)/sqrt(2))
        wb6=n*fa/na+(as6(fpicol*na/fa)/sqrt(2)-as6(n))*fa/(na*(as6(n+1)-
as6(n)));
    end
end
if(as7(n)<as7(fpicol*na/fa)/sqrt(2))
    if(as7(n+1)>as7(fpicol*na/fa)/sqrt(2))
        wa7=n*fa/na+(as7(fpicol*na/fa)/sqrt(2)-as7(n))*fa/(na*(as7(n+1)-
as7(n)));
    end
end
if(as7(n)>as7(fpicol*na/fa)/sqrt(2))
    if(as7(n+1)<as7(fpicol*na/fa)/sqrt(2))
        wb7=n*fa/na+(as7(fpicol*na/fa)/sqrt(2)-as7(n))*fa/(na*(as7(n+1)-
as7(n)));
    end
end
if(as8(n)<as8(fpicol*na/fa)/sqrt(2))
    if(as8(n+1)>as8(fpicol*na/fa)/sqrt(2))
        wa8=n*fa/na+(as8(fpicol*na/fa)/sqrt(2)-as8(n))*fa/(na*(as8(n+1)-
as8(n)));
    end
end
if(as8(n)>as8(fpicol*na/fa)/sqrt(2))
    if(as8(n+1)<as8(fpicol*na/fa)/sqrt(2))
        wb8=n*fa/na+(as8(fpicol*na/fa)/sqrt(2)-as8(n))*fa/(na*(as8(n+1)-
as8(n)));
    end
end
if(as9(n)<as9(fpicol*na/fa)/sqrt(2))
    if(as9(n+1)>as9(fpicol*na/fa)/sqrt(2))
        wa9=n*fa/na+(as9(fpicol*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
    end
end
if(as9(n)>as9(fpicol*na/fa)/sqrt(2))
    if(as9(n+1)<as9(fpicol*na/fa)/sqrt(2))
        wb9=n*fa/na+(as9(fpicol*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
    end
end

```

```

end
if(as10(n)<as10(fpicol*na/fa)/sqrt(2))
    if(as10(n+1)>as10(fpicol*na/fa)/sqrt(2))
        wa10=n*fa/na+(as10(fpicol*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
    end
end
end
if(as10(n)>as10(fpicol*na/fa)/sqrt(2))
    if(as10(n+1)<as10(fpicol*na/fa)/sqrt(2))
        wb10=n*fa/na+(as10(fpicol*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
    end
end
end
ksi1=[wb1-wa1 wb2-wa2 wb3-wa3 wb4-wa4 wb5-wa5 wb6-wa6 wb7-wa7 wb8-wa8 wb9-wa9
wb10-wa10]/(2*fpicol);
for n=fpico2*na/fa-10:fpico2*na/fa+10
    if(as1(n)<as1(fpico2*na/fa)/sqrt(2))
        if(as1(n+1)>as1(fpico2*na/fa)/sqrt(2))
            wa1=n*fa/na+(as1(fpico2*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as1(n)>as1(fpico2*na/fa)/sqrt(2))
        if(as1(n+1)<as1(fpico2*na/fa)/sqrt(2))
            wb1=n*fa/na+(as1(fpico2*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as2(n)<as2(fpico2*na/fa)/sqrt(2))
        if(as2(n+1)>as2(fpico2*na/fa)/sqrt(2))
            wa2=n*fa/na+(as2(fpico2*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as2(n)>as2(fpico2*na/fa)/sqrt(2))
        if(as2(n+1)<as2(fpico2*na/fa)/sqrt(2))
            wb2=n*fa/na+(as2(fpico2*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as3(n)<as3(fpico2*na/fa)/sqrt(2))
        if(as3(n+1)>as3(fpico2*na/fa)/sqrt(2))
            wa3=n*fa/na+(as3(fpico2*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
    if(as3(n)>as3(fpico2*na/fa)/sqrt(2))
        if(as3(n+1)<as3(fpico2*na/fa)/sqrt(2))
            wb3=n*fa/na+(as3(fpico2*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
    if(as4(n)<as4(fpico2*na/fa)/sqrt(2))
        if(as4(n+1)>as4(fpico2*na/fa)/sqrt(2))
            wa4=n*fa/na+(as4(fpico2*na/fa)/sqrt(2)-as4(n))*fa/(na*(as4(n+1)-
as4(n)));

```

```

        end
    end
    if (as4(n) > as4(fpico2*na/fa)/sqrt(2))
        if (as4(n+1) < as4(fpico2*na/fa)/sqrt(2))
            wb4 = n*fa/na + (as4(fpico2*na/fa)/sqrt(2) - as4(n)) * fa / (na * (as4(n+1) -
as4(n)));
        end
    end
    if (as5(n) < as5(fpico2*na/fa)/sqrt(2))
        if (as5(n+1) > as5(fpico2*na/fa)/sqrt(2))
            wa5 = n*fa/na + (as5(fpico2*na/fa)/sqrt(2) - as5(n)) * fa / (na * (as5(n+1) -
as5(n)));
        end
    end
    if (as5(n) > as5(fpico2*na/fa)/sqrt(2))
        if (as5(n+1) < as5(fpico2*na/fa)/sqrt(2))
            wb5 = n*fa/na + (as5(fpico2*na/fa)/sqrt(2) - as5(n)) * fa / (na * (as5(n+1) -
as5(n)));
        end
    end
    if (as6(n) < as6(fpico2*na/fa)/sqrt(2))
        if (as6(n+1) > as6(fpico2*na/fa)/sqrt(2))
            wa6 = n*fa/na + (as6(fpico2*na/fa)/sqrt(2) - as6(n)) * fa / (na * (as6(n+1) -
as6(n)));
        end
    end
    if (as6(n) > as6(fpico2*na/fa)/sqrt(2))
        if (as6(n+1) < as6(fpico2*na/fa)/sqrt(2))
            wb6 = n*fa/na + (as6(fpico2*na/fa)/sqrt(2) - as6(n)) * fa / (na * (as6(n+1) -
as6(n)));
        end
    end
    if (as7(n) < as7(fpico2*na/fa)/sqrt(2))
        if (as7(n+1) > as7(fpico2*na/fa)/sqrt(2))
            wa7 = n*fa/na + (as7(fpico2*na/fa)/sqrt(2) - as7(n)) * fa / (na * (as7(n+1) -
as7(n)));
        end
    end
    if (as7(n) > as7(fpico2*na/fa)/sqrt(2))
        if (as7(n+1) < as7(fpico2*na/fa)/sqrt(2))
            wb7 = n*fa/na + (as7(fpico2*na/fa)/sqrt(2) - as7(n)) * fa / (na * (as7(n+1) -
as7(n)));
        end
    end
    if (as8(n) < as8(fpico2*na/fa)/sqrt(2))
        if (as8(n+1) > as8(fpico2*na/fa)/sqrt(2))
            wa8 = n*fa/na + (as8(fpico2*na/fa)/sqrt(2) - as8(n)) * fa / (na * (as8(n+1) -
as8(n)));
        end
    end
    if (as8(n) > as8(fpico2*na/fa)/sqrt(2))
        if (as8(n+1) < as8(fpico2*na/fa)/sqrt(2))
            wb8 = n*fa/na + (as8(fpico2*na/fa)/sqrt(2) - as8(n)) * fa / (na * (as8(n+1) -
as8(n)));
        end
    end
    if (as9(n) < as9(fpico2*na/fa)/sqrt(2))

```

```

        if(as9(n+1)>as9(fpico2*na/fa)/sqrt(2))
            wa9=n*fa/na+(as9(fpico2*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
        end
    end
    if(as9(n)>as9(fpico2*na/fa)/sqrt(2))
        if(as9(n+1)<as9(fpico2*na/fa)/sqrt(2))
            wb9=n*fa/na+(as9(fpico2*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
        end
    end
    if(as10(n)<as10(fpico2*na/fa)/sqrt(2))
        if(as10(n+1)>as10(fpico2*na/fa)/sqrt(2))
            wal0=n*fa/na+(as10(fpico2*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
        end
    end
    if(as10(n)>as10(fpico2*na/fa)/sqrt(2))
        if(as10(n+1)<as10(fpico2*na/fa)/sqrt(2))
            wbl0=n*fa/na+(as10(fpico2*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
        end
    end
end
csi2=[wbl-wal wb2-wa2 wb3-wa3 wb4-wa4 wb5-wa5 wb6-wa6 wb7-wa7 wb8-wa8 wb9-wa9
wbl0-wal0]./(2*fpico2);
for n=fpico3*na/fa-10:fpico3*na/fa+10
    if(as1(n)<as1(fpico3*na/fa)/sqrt(2))
        if(as1(n+1)>as1(fpico3*na/fa)/sqrt(2))
            wal=n*fa/na+(as1(fpico3*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as1(n)>as1(fpico3*na/fa)/sqrt(2))
        if(as1(n+1)<as1(fpico3*na/fa)/sqrt(2))
            wbl=n*fa/na+(as1(fpico3*na/fa)/sqrt(2)-as1(n))*fa/(na*(as1(n+1)-
as1(n)));
        end
    end
    if(as2(n)<as2(fpico3*na/fa)/sqrt(2))
        if(as2(n+1)>as2(fpico3*na/fa)/sqrt(2))
            wa2=n*fa/na+(as2(fpico3*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as2(n)>as2(fpico3*na/fa)/sqrt(2))
        if(as2(n+1)<as2(fpico3*na/fa)/sqrt(2))
            wb2=n*fa/na+(as2(fpico3*na/fa)/sqrt(2)-as2(n))*fa/(na*(as2(n+1)-
as2(n)));
        end
    end
    if(as3(n)<as3(fpico3*na/fa)/sqrt(2))
        if(as3(n+1)>as3(fpico3*na/fa)/sqrt(2))
            wa3=n*fa/na+(as3(fpico3*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
end
end

```

```

    if(as3(n)>as3(fpico3*na/fa)/sqrt(2))
        if(as3(n+1)<as3(fpico3*na/fa)/sqrt(2))
            wb3=n*fa/na+(as3(fpico3*na/fa)/sqrt(2)-as3(n))*fa/(na*(as3(n+1)-
as3(n)));
        end
    end
    if(as4(n)<as4(fpico3*na/fa)/sqrt(2))
        if(as4(n+1)>as4(fpico3*na/fa)/sqrt(2))
            wa4=n*fa/na+(as4(fpico3*na/fa)/sqrt(2)-as4(n))*fa/(na*(as4(n+1)-
as4(n)));
        end
    end
    if(as4(n)>as4(fpico3*na/fa)/sqrt(2))
        if(as4(n+1)<as4(fpico3*na/fa)/sqrt(2))
            wb4=n*fa/na+(as4(fpico3*na/fa)/sqrt(2)-as4(n))*fa/(na*(as4(n+1)-
as4(n)));
        end
    end
    if(as5(n)<as5(fpico3*na/fa)/sqrt(2))
        if(as5(n+1)>as5(fpico3*na/fa)/sqrt(2))
            wa5=n*fa/na+(as5(fpico3*na/fa)/sqrt(2)-as5(n))*fa/(na*(as5(n+1)-
as5(n)));
        end
    end
    if(as5(n)>as5(fpico3*na/fa)/sqrt(2))
        if(as5(n+1)<as5(fpico3*na/fa)/sqrt(2))
            wb5=n*fa/na+(as5(fpico3*na/fa)/sqrt(2)-as5(n))*fa/(na*(as5(n+1)-
as5(n)));
        end
    end
    if(as6(n)<as6(fpico3*na/fa)/sqrt(2))
        if(as6(n+1)>as6(fpico3*na/fa)/sqrt(2))
            wa6=n*fa/na+(as6(fpico3*na/fa)/sqrt(2)-as6(n))*fa/(na*(as6(n+1)-
as6(n)));
        end
    end
    if(as6(n)>as6(fpico3*na/fa)/sqrt(2))
        if(as6(n+1)<as6(fpico3*na/fa)/sqrt(2))
            wb6=n*fa/na+(as6(fpico3*na/fa)/sqrt(2)-as6(n))*fa/(na*(as6(n+1)-
as6(n)));
        end
    end
    if(as7(n)<as7(fpico3*na/fa)/sqrt(2))
        if(as7(n+1)>as7(fpico3*na/fa)/sqrt(2))
            wa7=n*fa/na+(as7(fpico3*na/fa)/sqrt(2)-as7(n))*fa/(na*(as7(n+1)-
as7(n)));
        end
    end
    if(as7(n)>as7(fpico3*na/fa)/sqrt(2))
        if(as7(n+1)<as7(fpico3*na/fa)/sqrt(2))
            wb7=n*fa/na+(as7(fpico3*na/fa)/sqrt(2)-as7(n))*fa/(na*(as7(n+1)-
as7(n)));
        end
    end
    if(as8(n)<as8(fpico3*na/fa)/sqrt(2))
        if(as8(n+1)>as8(fpico3*na/fa)/sqrt(2))

```

```

        wa8=n*fa/na+(as8(fpico3*na/fa)/sqrt(2)-as8(n))*fa/(na*(as8(n+1)-
as8(n)));
    end
end
if(as8(n)>as8(fpico3*na/fa)/sqrt(2))
    if(as8(n+1)<as8(fpico3*na/fa)/sqrt(2))
        wb8=n*fa/na+(as8(fpico3*na/fa)/sqrt(2)-as8(n))*fa/(na*(as8(n+1)-
as8(n)));
    end
end
if(as9(n)<as9(fpico3*na/fa)/sqrt(2))
    if(as9(n+1)>as9(fpico3*na/fa)/sqrt(2))
        wa9=n*fa/na+(as9(fpico3*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
    end
end
if(as9(n)>as9(fpico3*na/fa)/sqrt(2))
    if(as9(n+1)<as9(fpico3*na/fa)/sqrt(2))
        wb9=n*fa/na+(as9(fpico3*na/fa)/sqrt(2)-as9(n))*fa/(na*(as9(n+1)-
as9(n)));
    end
end
if(as10(n)<as10(fpico3*na/fa)/sqrt(2))
    if(as10(n+1)>as10(fpico3*na/fa)/sqrt(2))
        wa10=n*fa/na+(as10(fpico3*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
    end
end
if(as10(n)>as10(fpico3*na/fa)/sqrt(2))
    if(as10(n+1)<as10(fpico3*na/fa)/sqrt(2))
        wb10=n*fa/na+(as10(fpico3*na/fa)/sqrt(2)-as10(n))*fa/(na*(as10(n+1)-
as10(n)));
    end
end
end
csi3=[wb1-wa1 wb2-wa2 wb3-wa3 wb4-wa4 wb5-wa5 wb6-wa6 wb7-wa7 wb8-wa8 wb9-wa9
wb10-wa10]./(2*fpico3);

```

% Modulo 8: Determinacao das amplitudes dos modos de vibrar (ui) da viga:

```

l11=20*[log10(as1(fpico1*na/fa)) log10(as1(fpico2*na/fa))
log10(as1(fpico3*na/fa))];
l21=20*[log10(as2(fpico1*na/fa)) log10(as2(fpico2*na/fa))
log10(as2(fpico3*na/fa))];
l31=20*[log10(as3(fpico1*na/fa)) log10(as3(fpico2*na/fa))
log10(as3(fpico3*na/fa))];
l41=20*[log10(as4(fpico1*na/fa)) log10(as4(fpico2*na/fa))
log10(as4(fpico3*na/fa))];
l51=20*[log10(as5(fpico1*na/fa)) log10(as5(fpico2*na/fa))
log10(as5(fpico3*na/fa))];
l61=20*[log10(as6(fpico1*na/fa)) log10(as6(fpico2*na/fa))
log10(as6(fpico3*na/fa))];
l71=20*[log10(as7(fpico1*na/fa)) log10(as7(fpico2*na/fa))
log10(as7(fpico3*na/fa))];
l81=20*[log10(as8(fpico1*na/fa)) log10(as8(fpico2*na/fa))
log10(as8(fpico3*na/fa))];

```

```

H91=20*[log10(as9(fpico1*na/fa)) log10(as9(fpico2*na/fa))
log10(as9(fpico3*na/fa))];
H101=20*[log10(as10(fpico1*na/fa)) log10(as10(fpico2*na/fa))
log10(as10(fpico3*na/fa))];
u1ult11=2*ksi1(1)*wn(1)^2*H11(1);
u1ult21=2*ksi1(2)*wn(1)^2*H21(1);
u1ult31=2*ksi1(3)*wn(1)^2*H31(1);
u1ult41=2*ksi1(4)*wn(1)^2*H41(1);
u1ult51=2*ksi1(5)*wn(1)^2*H51(1);
u1ult61=2*ksi1(6)*wn(1)^2*H61(1);
u1ult71=2*ksi1(7)*wn(1)^2*H71(1);
u1ult81=2*ksi1(8)*wn(1)^2*H81(1);
u1ult91=2*ksi1(9)*wn(1)^2*H91(1);
u1ult101=2*ksi1(10)*wn(1)^2*H101(1);
i2u2t11=2*ksi2(1)*wn(2)^2*H11(2);
i2u2t21=2*ksi2(2)*wn(2)^2*H21(2);
i2u2t31=2*ksi2(3)*wn(2)^2*H31(2);
i2u2t41=2*ksi2(4)*wn(2)^2*H41(2);
i2u2t51=2*ksi2(5)*wn(2)^2*H51(2);
i2u2t61=2*ksi2(6)*wn(2)^2*H61(2);
i2u2t71=2*ksi2(7)*wn(2)^2*H71(2);
i2u2t81=2*ksi2(8)*wn(2)^2*H81(2);
i2u2t91=2*ksi2(9)*wn(2)^2*H91(2);
i2u2t101=2*ksi2(10)*wn(2)^2*H101(2);
i3u3t11=2*ksi3(1)*wn(3)^2*H11(3);
i3u3t21=2*ksi3(2)*wn(3)^2*H21(3);
i3u3t31=2*ksi3(3)*wn(3)^2*H31(3);
i3u3t41=2*ksi3(4)*wn(3)^2*H41(3);
i3u3t51=2*ksi3(5)*wn(3)^2*H51(3);
i3u3t61=2*ksi3(6)*wn(3)^2*H61(3);
i3u3t71=2*ksi3(7)*wn(3)^2*H71(3);
i3u3t81=2*ksi3(8)*wn(3)^2*H81(3);
i3u3t91=2*ksi3(9)*wn(3)^2*H91(3);
i3u3t101=2*ksi3(10)*wn(3)^2*H101(3);

a1=sqrt(u1ult11);
i2=u1ult21/a1;
i3=u1ult31/a1;
i4=u1ult41/a1;
i5=u1ult51/a1;
i6=u1ult61/a1;
i7=u1ult71/a1;
i8=u1ult81/a1;
i9=u1ult91/a1;
i10=u1ult101/a1;
>1=sqrt(u2u2t11);
>2=u2u2t21/b1;
>3=u2u2t31/b1;
>4=u2u2t41/b1;
>5=u2u2t51/b1;
>6=u2u2t61/b1;
>7=u2u2t71/b1;
>8=u2u2t81/b1;
>9=u2u2t91/b1;
>10=u2u2t101/b1;
>1=sqrt(u3u3t11);
>2=u3u3t21/c1;

```

```

c3=u3u3t31/c1;
c4=u3u3t41/c1;
c5=u3u3t51/c1;
c6=u3u3t61/c1;
c7=u3u3t71/c1;
c8=u3u3t81/c1;
c9=u3u3t91/c1;
c10=u3u3t101/c1;
i1=[0 a1 a2 a3 a4 a5 a6 a7 a8 a9 a10]./max([a1 a2 a3 a4 a5 a6 a7 a8 a9 a10]);
i2=[0 b1 b2 b3 b4 b5 b6 b7 b8 -b9 -b10]./max([b1 b2 b3 b4 b5 b6 b7 b8 b9 b10]);
i3=[0 c1 c2 c3 c4 c5 -c6 -c7 -c8 c9 c10]./max([c1 c2 c3 c4 c5 c6 c7 c8 c9 c10]);

f4=[sum(xbatida.^4) sum(xbatida.^5) sum(xbatida.^6) sum(xbatida.^7);
sum(xbatida.^5) sum(xbatida.^6) sum(xbatida.^7) sum(xbatida.^8); sum(xbatida.^6)
sum(xbatida.^7) sum(xbatida.^8) sum(xbatida.^9); sum(xbatida.^7) sum(xbatida.^8)
sum(xbatida.^9) sum(xbatida.^10)];
R1=[sum(xbatida.^2.*u1) sum(xbatida.^3.*u1) sum(xbatida.^4.*u1)
sum(xbatida.^5.*u1)];
R2=[sum(xbatida.^2.*u2) sum(xbatida.^3.*u2) sum(xbatida.^4.*u2)
sum(xbatida.^5.*u2)];
R3=[sum(xbatida.^2.*u3) sum(xbatida.^3.*u3) sum(xbatida.^4.*u3)
sum(xbatida.^5.*u3)];
K1=R1/M;
K2=R2/M;
K3=R3/M;
K1=K1(1)*x.^2+K1(2)*x.^3+K1(3)*x.^4+K1(4)*x.^5;
K2=K2(1)*x.^2+K2(2)*x.^3+K2(3)*x.^4+K2(4)*x.^5;
K3=K3(1)*x.^2+K3(2)*x.^3+K3(3)*x.^4+K3(4)*x.^5;

% Modulo 9: Determinacao do angulo de fase em funcao da frequencia:

fase1=360*atan(rs1soma./is1soma)./pi;
fase2=360*atan(rs2soma./is2soma)./pi;
fase3=360*atan(rs3soma./is3soma)./pi;
fase4=360*atan(rs4soma./is4soma)./pi;
fase5=360*atan(rs5soma./is5soma)./pi;
fase6=360*atan(rs6soma./is6soma)./pi;
fase7=360*atan(rs7soma./is7soma)./pi;
fase8=360*atan(rs8soma./is8soma)./pi;
fase9=360*atan(rs9soma./is9soma)./pi;
fase10=360*atan(rs10soma./is10soma)./pi;

% Modulo 10: Saida grafica da fase em funcao da freq e dos modos de vibrar:

figure(20)
plot(w,fase3)
title(['Fase do sinal obtido no ponto 3']);
xlabel('frequencia (Hz)');
ylabel('fase (em graus)');
figure(21)
plot(w,fase6)
title(['Fase do sinal obtido no ponto 6']);
xlabel('frequencia (Hz)');
ylabel('fase (em graus)');
figure(22)
plot(w,fase10)
title(['Fase do sinal obtido no ponto 10']);

```

Programa para análise modal da placa biapoada estudada (Placa.m):

```
% Modulo 1: Aquisicao de dados:

format compact
clear all
clc
global x0 y0 raio rq % variaveis globais usadas na funcao nyquist
E=210000000000; % modulo de Young em Pa
ro=7860; % densidade da placa, em kg/m^3
ni=0.3; % coeficiente de Poisson
a=.3; % Comprimento da placa, em metros
b=.2; % Largura da placa, em metros
h=.002; % Espessura da placa, em metros
D=E*h^3/(12*(1-ni^2));
x=[0:.02:a]; % Vetor posicao x na placa - comprimento
y=[0:.02:b]; % Vetor posicao y na placa - largura
xc=[0:.001:a];
yc=[0:.001:b];
fa=4400; % Frequencia de aquisicao de dados, em Hz
na=4400; % Numero de dados (pontos) aquisitados
t=[1:1:na]; % Vetor tempo de aquisicao de dados
w=[fa/na:fa/na:fa]; % Vetor frequencia varrida pelo sinal
alpha=[pi/a:pi/a:3*pi/a];
beta0=[10.305433164820414 20.742175557961834 31.202948053539938 41.6676890569522];
beta1=[15.5981 27.0127 41.2031 56.2723];
beta2=[24.6893 33.9861 46.1196 59.8559];
beta3=[34.1828 42.0535 52.6843 65.1033];
w1=sqrt(D/(ro*h))*beta1.^2; % Frequencias naturais analiticas w1_
w2=sqrt(D/(ro*h))*beta2.^2; % Frequencias naturais analiticas w2_
w3=sqrt(D/(ro*h))*beta3.^2; % Frequencias naturais analiticas w3_

% modos de vibrar analitico Oi - componente Y
for n=1:4
    if(n==1)
        bt=10.305433164820414;
        al=pi/a;
    end
    if(n==2)
        bt=20.742175557961834;
        al=2*pi/a;
    end
    if(n==3)
        bt=31.202948053539938;
        al=3*pi/a;
    end
    if(n==4)
        bt=41.6676890569522;
        al=4*pi/a;
    end
    p=sqrt(al^2-bt^2);
    r=sqrt(al^2+bt^2);
    e12=p^2-ni*al^2;
    e14=r^2-ni*al^2;
    e21=p^3-(2-ni)*al^2*p;
    e23=r^3-(2-ni)*al^2*r;
```

```

e31=e21*cosh(p*b);
e32=e21*sinh(p*b);
e33=e23*cosh(r*b);
e34=e23*sinh(r*b);
e41=e12*sinh(p*b);
e42=e12*cosh(p*b);
e43=e14*sinh(r*b);
e44=e14*cosh(r*b);
EE=e12*sinh(p*b)-e14*e21/e23*sinh(r*b)+e21*(cosh(p*b)-cosh(r*b));
FF=e21*sinh(p*b)-e23*e12/e14*sinh(r*b)+e12*(cosh(p*b)-cosh(r*b));
AA=1;
BB=-EE/FF;
CC=-e21/e23;
DD=e12*EE/e14/FF;
if(n==1)
    % modo de vibrar analitico 01 - componente Y
    Y01=AA*sinh(p*yc)+BB*cosh(p*yc)+CC*sinh(r*yc)+DD*cosh(r*yc);
    Y01=-Y01/max(abs(Y01));
end
if(n==2)
    % modo de vibrar analitico 02 - componente Y
    Y02=AA*sinh(p*yc)+BB*cosh(p*yc)+CC*sinh(r*yc)+DD*cosh(r*yc);
    Y02=-Y02/max(abs(Y02));
end
if(n==3)
    % modo de vibrar analitico 03 - componente Y
    Y03=AA*sinh(p*yc)+BB*cosh(p*yc)+CC*sinh(r*yc)+DD*cosh(r*yc);
    Y03=-Y03/max(abs(Y03));
end
if(n==4)
    % modo de vibrar analitico 04 - componente Y
    Y04=AA*sinh(p*yc)+BB*cosh(p*yc)+CC*sinh(r*yc)+DD*cosh(r*yc);
    Y04=-Y04/max(abs(Y04));
end
end

% modo de vibrar analitico 11 - componente Y
a=sqrt(betal(1)^2+alpha(1)^2);
s=sqrt(betal(1)^2-alpha(1)^2);
B1=r*(cos(s*b)-cosh(r*b))*(betal(1)^4-alpha(1)^4*(1-ni)^2)*(betal(1)^2-alpha(1)^2*(1-ni));
B2=(betal(1)^2+alpha(1)^2*(1-ni))*(s*(betal(1)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
*(betal(1)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(betal(1)^4-alpha(1)^4*(1-ni)^2)*(betal(1)^2+alpha(1)^2*(1-ni));
B4=(betal(1)^2-alpha(1)^2*(1-ni))*(s*(betal(1)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
*(betal(1)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
U11=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 12 - componente Y
a=sqrt(betal(2)^2+alpha(1)^2);
s=sqrt(betal(2)^2-alpha(1)^2);
B1=r*(cos(s*b)-cosh(r*b))*(betal(2)^4-alpha(1)^4*(1-ni)^2)*(betal(2)^2-alpha(1)^2*(1-ni));
B2=(betal(2)^2+alpha(1)^2*(1-ni))*(s*(betal(2)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
*(betal(2)^2-alpha(1)^2*(1-ni))^2*sin(s*b));

```

```

xlabel('frequencia (Hz)');
ylabel('fase (em graus)');
figure(23)
plot(xbatida,u1,'*',x,X1./max(X1),x,x1)
title(['Modo de vibrar - 1a frequencia natural=',num2str(fpicos(2)),'Hz']);
legend('Ptos. obtidos','modelo analitico','interpolado');
xlabel('distancia a partir da extremidade engastada');
ylabel('amplitude modulo 1');
figure(24)
plot(xbatida,u2,'*',x,X2./max(X2),x,x2)
title(['Modo de vibrar - 2a frequencia natural=',num2str(fpico2),'Hz']);
legend('Ptos. obtidos','modelo analitico','interpolado');
xlabel('distancia a partir da extremidade engastada');
ylabel('amplitude modulo 1');
figure(25)
plot(xbatida,u3,'*',x,X3./max(X3),x,x3)
title(['Modo de vibrar - 3a frequencia natural=',num2str(fpico3),'Hz']);
legend('Ptos. obtidos','modelo analitico','interpolado');
xlabel('distancia a partir da extremidade engastada');
ylabel('amplitude modulo 1');

```

```

B3=s*(cos(s*b)-cosh(r*b))*(beta1(2)^4-alpha(1)^4*(1-ni)^2)*(beta1(2)^2+alpha(1)^2*(1-
ni));
B4=(beta1(2)^2-alpha(1)^2*(1-ni))*(s*(beta1(2)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
r*(beta1(2)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
Y12=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

```

modo de vibrar analitico 13 - componente Y

```

r=sqrt(beta1(3)^2+alpha(1)^2);
s=sqrt(beta1(3)^2-alpha(1)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta1(3)^4-alpha(1)^4*(1-ni)^2)*(beta1(3)^2-alpha(1)^2*(1-
ni));
B2=(beta1(3)^2+alpha(1)^2*(1-ni))*(s*(beta1(3)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
r*(beta1(3)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta1(3)^4-alpha(1)^4*(1-ni)^2)*(beta1(3)^2+alpha(1)^2*(1-
ni));
B4=(beta1(3)^2-alpha(1)^2*(1-ni))*(s*(beta1(3)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
r*(beta1(3)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
Y13=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

```

modo de vibrar analitico 14 - componente Y

```

r=sqrt(beta1(4)^2+alpha(1)^2);
s=sqrt(beta1(4)^2-alpha(1)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta1(4)^4-alpha(1)^4*(1-ni)^2)*(beta1(4)^2-alpha(1)^2*(1-
ni));
B2=(beta1(4)^2+alpha(1)^2*(1-ni))*(s*(beta1(4)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
r*(beta1(4)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta1(4)^4-alpha(1)^4*(1-ni)^2)*(beta1(4)^2+alpha(1)^2*(1-
ni));
B4=(beta1(4)^2-alpha(1)^2*(1-ni))*(s*(beta1(4)^2+alpha(1)^2*(1-ni))^2*sinh(r*b)-
r*(beta1(4)^2-alpha(1)^2*(1-ni))^2*sin(s*b));
Y14=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

```

modo de vibrar analitico 21 - componente Y

```

r=sqrt(beta2(1)^2+alpha(2)^2);
s=sqrt(beta2(1)^2-alpha(2)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta2(1)^4-alpha(2)^4*(1-ni)^2)*(beta2(1)^2-alpha(2)^2*(1-
ni));
B2=(beta2(1)^2+alpha(2)^2*(1-ni))*(s*(beta2(1)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(1)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta2(1)^4-alpha(2)^4*(1-ni)^2)*(beta2(1)^2+alpha(2)^2*(1-
ni));
B4=(beta2(1)^2-alpha(2)^2*(1-ni))*(s*(beta2(1)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(1)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
Y21=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

```

modo de vibrar analitico 22 - componente Y

```

r=sqrt(beta2(2)^2+alpha(2)^2);
s=sqrt(beta2(2)^2-alpha(2)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta2(2)^4-alpha(2)^4*(1-ni)^2)*(beta2(2)^2-alpha(2)^2*(1-
ni));
B2=(beta2(2)^2+alpha(2)^2*(1-ni))*(s*(beta2(2)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(2)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta2(2)^4-alpha(2)^4*(1-ni)^2)*(beta2(2)^2+alpha(2)^2*(1-
ni));
B4=(beta2(2)^2-alpha(2)^2*(1-ni))*(s*(beta2(2)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(2)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
Y22=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

```

```

% modo de vibrar analitico 23 - componente Y
r=sqrt(beta2(3)^2+alpha(2)^2);
s=sqrt(beta2(3)^2-alpha(2)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta2(3)^4-alpha(2)^4*(1-ni)^2)*(beta2(3)^2-alpha(2)^2*(1-
ni));
B2=(beta2(3)^2+alpha(2)^2*(1-ni))*(s*(beta2(3)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(3)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta2(3)^4-alpha(2)^4*(1-ni)^2)*(beta2(3)^2+alpha(2)^2*(1-
ni));
B4=(beta2(3)^2-alpha(2)^2*(1-ni))*(s*(beta2(3)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(3)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
T23=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 24 - componente Y
r=sqrt(beta2(4)^2+alpha(2)^2);
s=sqrt(beta2(4)^2-alpha(2)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta2(4)^4-alpha(2)^4*(1-ni)^2)*(beta2(4)^2-alpha(2)^2*(1-
ni));
B2=(beta2(4)^2+alpha(2)^2*(1-ni))*(s*(beta2(4)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(4)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta2(4)^4-alpha(2)^4*(1-ni)^2)*(beta2(4)^2+alpha(2)^2*(1-
ni));
B4=(beta2(4)^2-alpha(2)^2*(1-ni))*(s*(beta2(4)^2+alpha(2)^2*(1-ni))^2*sinh(r*b)-
r*(beta2(4)^2-alpha(2)^2*(1-ni))^2*sin(s*b));
T24=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 31 - componente Y
r=sqrt(beta3(1)^2+alpha(3)^2);
s=sqrt(beta3(1)^2-alpha(3)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta3(1)^4-alpha(3)^4*(1-ni)^2)*(beta3(1)^2-alpha(3)^2*(1-
ni));
B2=(beta3(1)^2+alpha(3)^2*(1-ni))*(s*(beta3(1)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
r*(beta3(1)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta3(1)^4-alpha(3)^4*(1-ni)^2)*(beta3(1)^2+alpha(3)^2*(1-
ni));
B4=(beta3(1)^2-alpha(3)^2*(1-ni))*(s*(beta3(1)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
r*(beta3(1)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
T31=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 32 - componente Y
r=sqrt(beta3(2)^2+alpha(3)^2);
s=sqrt(beta3(2)^2-alpha(3)^2);
B1=r*(cos(s*b)-cosh(r*b))*(beta3(2)^4-alpha(3)^4*(1-ni)^2)*(beta3(2)^2-alpha(3)^2*(1-
ni));
B2=(beta3(2)^2+alpha(3)^2*(1-ni))*(s*(beta3(2)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
r*(beta3(2)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
B3=s*(cos(s*b)-cosh(r*b))*(beta3(2)^4-alpha(3)^4*(1-ni)^2)*(beta3(2)^2+alpha(3)^2*(1-
ni));
B4=(beta3(2)^2-alpha(3)^2*(1-ni))*(s*(beta3(2)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
r*(beta3(2)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
T32=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 33 - componente Y
r=sqrt(beta3(3)^2+alpha(3)^2);
s=sqrt(beta3(3)^2-alpha(3)^2);

```

```

31=r*(cos(s*b)-cosh(r*b))*(beta3(3)^4-alpha(3)^4*(1-ni)^2)*(beta3(3)^2-alpha(3)^2*(1-
ni));
32=(beta3(3)^2+alpha(3)^2*(1-ni))*(s*(beta3(3)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
c*(beta3(3)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
33=s*(cos(s*b)-cosh(r*b))*(beta3(3)^4-alpha(3)^4*(1-ni)^2)*(beta3(3)^2+alpha(3)^2*(1-
ni));
34=(beta3(3)^2-alpha(3)^2*(1-ni))*(s*(beta3(3)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
c*(beta3(3)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
f33=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

% modo de vibrar analitico 34 - componente Y
r=sqrt(beta3(4)^2+alpha(3)^2);
s=sqrt(beta3(4)^2-alpha(3)^2);
31=r*(cos(s*b)-cosh(r*b))*(beta3(4)^4-alpha(3)^4*(1-ni)^2)*(beta3(4)^2-alpha(3)^2*(1-
ni));
32=(beta3(4)^2+alpha(3)^2*(1-ni))*(s*(beta3(4)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
c*(beta3(4)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
33=s*(cos(s*b)-cosh(r*b))*(beta3(4)^4-alpha(3)^4*(1-ni)^2)*(beta3(4)^2+alpha(3)^2*(1-
ni));
34=(beta3(4)^2-alpha(3)^2*(1-ni))*(s*(beta3(4)^2+alpha(3)^2*(1-ni))^2*sinh(r*b)-
c*(beta3(4)^2-alpha(3)^2*(1-ni))^2*sin(s*b));
f34=(B1*sin(s*yc)+B2*cos(s*yc)+B3*sinh(r*yc)+B4*cosh(r*yc));

f1=sin((pi/a).*xc); % modo de vibrar analitico 1_ - componente X
f2=sin((2*pi/a).*xc); % modo de vibrar analitico 2_ - componente X
f3=sin((3*pi/a).*xc); % modo de vibrar analitico 3_ - componente X

% aquisicao dos dados do acelerometro em cada um dos pontos
load c:\progra~1\daq\applications\matlab\mart1_1;
for n=1:na
    ex1y1(n)=A(n,2);
    sx1y1(n)=A(n,1);
end
load c:\progra~1\daq\applications\matlab\mart2_1;
for n=1:na
    ex2y1(n)=A(n,2);
    sx2y1(n)=A(n,1);
end
load c:\progra~1\daq\applications\matlab\mart3_1;
for n=1:na
    ex3y1(n)=A(n,2);
    sx3y1(n)=A(n,1);
end
load c:\progra~1\daq\applications\matlab\mart4_1;
for n=1:na
    ex4y1(n)=A(n,2);
    sx4y1(n)=A(n,1);
end
load c:\progra~1\daq\applications\matlab\mart5_1;
for n=1:na
    ex5y1(n)=A(n,2);
    sx5y1(n)=A(n,1);
end
load c:\progra~1\daq\applications\matlab\mart6_1;
for n=1:na
    ex6y1(n)=A(n,2);
    sx6y1(n)=A(n,1);
end

```

```

end
load c:\progra~1\daqx\applications\matlab\mart7_1;
for n=1:na
    ex7y1(n)=A(n,2);
    sx7y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart8_1;
for n=1:na
    ex8y1(n)=A(n,2);
    sx8y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart9_1;
for n=1:na
    ex9y1(n)=A(n,2);
    sx9y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart10_1;
for n=1:na
    ex10y1(n)=A(n,2);
    sx10y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart11_1;
for n=1:na
    ex11y1(n)=A(n,2);
    sx11y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart12_1;
for n=1:na
    ex12y1(n)=A(n,2);
    sx12y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart13_1;
for n=1:na
    ex13y1(n)=A(n,2);
    sx13y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart14_1;
for n=1:na
    ex14y1(n)=A(n,2);
    sx14y1(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_0;
for n=1:na
    ex3y0(n)=A(n,2);
    sx3y0(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_2;
for n=1:na
    ex3y2(n)=A(n,2);
    sx3y2(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_3;
for n=1:na
    ex3y3(n)=A(n,2);
    sx3y3(n)=A(n,1);
end

```

```

load c:\progra~1\daqx\applications\matlab\mart3_4;
for n=1:na
    ex3y4(n)=A(n,2);
    sx3y4(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_5;
for n=1:na
    ex3y5(n)=A(n,2);
    sx3y5(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_6;
for n=1:na
    ex3y6(n)=A(n,2);
    sx3y6(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_7;
for n=1:na
    ex3y7(n)=A(n,2);
    sx3y7(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_8;
for n=1:na
    ex3y8(n)=A(n,2);
    sx3y8(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_9;
for n=1:na
    ex3y9(n)=A(n,2);
    sx3y9(n)=A(n,1);
end
load c:\progra~1\daqx\applications\matlab\mart3_10;
for n=1:na
    ex3y10(n)=A(n,2);
    sx3y10(n)=A(n,1);
end
clear A; % Apagar a variavel de auxilio A

% Modulo 2: Tratamento de dados:

% Transformada de Fourier: passa a amplitude do sinal para o espectro de frequencia

% Sinal de saida:

xs1=fft(sx1y1); % amplitude em funcao da frequencia: xs1
xs1=real(xs1); % parte real de xs1
xs1=imag(xs1); % parte imaginaria de xs1
xs2=fft(sx2y1); % amplitude em funcao da frequencia: xs2
xs2=real(xs2); % parte real de xs2
xs2=imag(xs2); % parte imaginaria de xs2
xs3=fft(sx3y1); % amplitude em funcao da frequencia: xs3
xs3=real(xs3); % parte real de xs3
xs3=imag(xs3); % parte imaginaria de xs3
xs4=fft(sx4y1); % amplitude em funcao da frequencia: xs4
xs4=real(xs4); % parte real de xs4
xs4=imag(xs4); % parte imaginaria de xs4
xs5=fft(sx5y1); % amplitude em funcao da frequencia: xs5
xs5=real(xs5); % parte real de xs5

```

```

ixs5=imag(xs5); % parte imaginaria de xs5
xs6=fft(sx6y1); % amplitude em funcao da frequencia: xs6
rxs6=real(xs6); % parte real de xs6
ixs6=imag(xs6); % parte imaginaria de xs6
xs7=fft(sx7y1); % amplitude em funcao da frequencia: xs7
rxs7=real(xs7); % parte real de xs7
ixs7=imag(xs7); % parte imaginaria de xs7
xs8=fft(sx8y1); % amplitude em funcao da frequencia: xs8
rxs8=real(xs8); % parte real de xs8
ixs8=imag(xs8); % parte imaginaria de xs8
xs9=fft(sx9y1); % amplitude em funcao da frequencia: xs9
rxs9=real(xs9); % parte real de xs9
ixs9=imag(xs9); % parte imaginaria de xs9
xs10=fft(sx10y1); % amplitude em funcao da frequencia: xs10
rxs10=real(xs10); % parte real de xs10
ixs10=imag(xs10); % parte imaginaria de xs10
xs11=fft(sx11y1); % amplitude em funcao da frequencia: xs11
rxs11=real(xs11); % parte real de xs11
ixs11=imag(xs11); % parte imaginaria de xs11
xs12=fft(sx12y1); % amplitude em funcao da frequencia: xs12
rxs12=real(xs12); % parte real de xs12
ixs12=imag(xs12); % parte imaginaria de xs12
xs13=fft(sx13y1); % amplitude em funcao da frequencia: xs13
rxs13=real(xs13); % parte real de xs13
ixs13=imag(xs13); % parte imaginaria de xs13
xs14=fft(sx14y1); % amplitude em funcao da frequencia: xs14
rxs14=real(xs14); % parte real de xs14
ixs14=imag(xs14); % parte imaginaria de xs14

ys0=fft(sx3y0); % amplitude em funcao da frequencia: ys0
rys0=real(ys0); % parte real de ys0
iys0=imag(ys0); % parte imaginaria de ys0
ys1=fft(sx3y1); % amplitude em funcao da frequencia: ys1
rys1=real(ys1); % parte real de ys1
iys1=imag(ys1); % parte imaginaria de ys1
ys2=fft(sx3y2); % amplitude em funcao da frequencia: ys2
rys2=real(ys2); % parte real de ys2
iys2=imag(ys2); % parte imaginaria de ys2
ys3=fft(sx3y3); % amplitude em funcao da frequencia: ys3
rys3=real(ys3); % parte real de ys3
iys3=imag(ys3); % parte imaginaria de ys3
ys4=fft(sx3y4); % amplitude em funcao da frequencia: ys4
rys4=real(ys4); % parte real de ys4
iys4=imag(ys4); % parte imaginaria de ys4
ys5=fft(sx3y5); % amplitude em funcao da frequencia: ys5
rys5=real(ys5); % parte real de ys5
iys5=imag(ys5); % parte imaginaria de ys5
ys6=fft(sx3y6); % amplitude em funcao da frequencia: ys6
rys6=real(ys6); % parte real de ys6
iys6=imag(ys6); % parte imaginaria de ys6
ys7=fft(sx3y7); % amplitude em funcao da frequencia: ys7
rys7=real(ys7); % parte real de ys7
iys7=imag(ys7); % parte imaginaria de ys7
ys8=fft(sx3y8); % amplitude em funcao da frequencia: ys8
rys8=real(ys8); % parte real de ys8
iys8=imag(ys8); % parte imaginaria de ys8
ys9=fft(sx3y9); % amplitude em funcao da frequencia: ys9

```

```

cys9=real(ys9); % parte real de ys9
lys9=imag(ys9); % parte imaginaria de ys9
ys10=fft(sx3y10); % amplitude em funcao da frequencia: ys10
cys10=real(ys10); % parte real de ys10
lys10=imag(ys10); % parte imaginaria de ys10

% Modulo 3: Saida grafica da amplitude do sinal em funcao da frequencia:

figure(1)
plot(w,abs(xs1).*w,w,-abs(xs14).*w)
axis([0 fa/2 -max(abs(xs14(1:na/2)).*w(1:na/2)) max(abs(xs1(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (1,1) e (14,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(2)
plot(w,abs(xs2).*w,w,-abs(xs13).*w)
axis([0 fa/2 -max(abs(xs13(1:na/2)).*w(1:na/2)) max(abs(xs2(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (2,1) e (13,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(3)
plot(w,abs(xs3).*w,w,-abs(xs12).*w)
axis([0 fa/2 -max(abs(xs12(1:na/2)).*w(1:na/2)) max(abs(xs3(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,1) e (12,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(4)
plot(w,abs(xs4).*w,w,-abs(xs11).*w)
axis([0 fa/2 -max(abs(xs11(1:na/2)).*w(1:na/2)) max(abs(xs4(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (4,1) e (11,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(5)
plot(w,abs(xs5).*w,w,-abs(xs10).*w)
axis([0 fa/2 -max(abs(xs10(1:na/2)).*w(1:na/2)) max(abs(xs5(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (5,1) e (10,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(6)
plot(w,abs(xs6).*w,w,-abs(xs9).*w)
axis([0 fa/2 -max(abs(xs9(1:na/2)).*w(1:na/2)) max(abs(xs6(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (6,1) e (9,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(7)
plot(w,abs(xs7).*w,w,-abs(xs8).*w)
axis([0 fa/2 -max(abs(xs8(1:na/2)).*w(1:na/2)) max(abs(xs7(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (7,1) e (8,1)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ax')
figure(8)
plot(w,abs(ys0).*w,w,-abs(ys10).*w)
axis([0 fa/2 -max(abs(ys10(1:na/2)).*w(1:na/2)) max(abs(ys0(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,0) e (3,10)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')

```

```

figure(9)
plot(w,abs(ys1).*w,w,-abs(ys9).*w)
axis([0 fa/2 -max(abs(ys9(1:na/2)).*w(1:na/2)) max(abs(ys1(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,1) e (3,9)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')
figure(10)
plot(w,abs(ys2).*w,w,-abs(ys8).*w)
axis([0 fa/2 -max(abs(ys8(1:na/2)).*w(1:na/2)) max(abs(ys2(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,2) e (3,8)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')
figure(11)
plot(w,abs(ys3).*w,w,-abs(ys7).*w)
axis([0 fa/2 -max(abs(ys7(1:na/2)).*w(1:na/2)) max(abs(ys3(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,3) e (3,7)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')
figure(12)
plot(w,abs(ys4).*w,w,-abs(ys6).*w)
axis([0 fa/2 -max(abs(ys6(1:na/2)).*w(1:na/2)) max(abs(ys4(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Pontos (3,4) e (3,6)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')
figure(13)
plot(w,abs(ys5).*w)
axis([0 fa/2 0 max(abs(ys5(1:na/2)).*w(1:na/2))]);
title(['FFT do sinal - Ponto (3,5)']);
xlabel('frequencia (Hz)');
ylabel('amplitude Ay')

```

Modulo 4: Pre-selecao das frequencias candidatas a frequencia natural:

```

for n=1:na/2+2
    as1(n)=abs(xs1(n))*w(n);
    as2(n)=abs(xs2(n))*w(n);
    as3(n)=abs(xs3(n))*w(n);
    as4(n)=abs(xs4(n))*w(n);
    as5(n)=abs(xs5(n))*w(n);
    as6(n)=abs(xs6(n))*w(n);
    as7(n)=abs(xs7(n))*w(n);
    as8(n)=abs(xs8(n))*w(n);
    as9(n)=abs(xs9(n))*w(n);
    as10(n)=abs(xs10(n))*w(n);
    as11(n)=abs(xs11(n))*w(n);
    as12(n)=abs(xs12(n))*w(n);
    as13(n)=abs(xs13(n))*w(n);
    as14(n)=abs(xs14(n))*w(n);
    bs0(n)=abs(ys0(n))*w(n);
    bs1(n)=abs(ys1(n))*w(n);
    bs2(n)=abs(ys2(n))*w(n);
    bs3(n)=abs(ys3(n))*w(n);
    bs4(n)=abs(ys4(n))*w(n);
    bs5(n)=abs(ys5(n))*w(n);
    bs6(n)=abs(ys6(n))*w(n);
    bs7(n)=abs(ys7(n))*w(n);
    bs8(n)=abs(ys8(n))*w(n);

```

```

bs9(n)=abs(ys9(n))*w(n);
bs10(n)=abs(ys10(n))*w(n);
end
for n=1:na/2+2
    if (as1(n)<max(as1)/20)
        as1(n)=0;
    end
    if (as2(n)<max(as2)/20)
        as2(n)=0;
    end
    if (as3(n)<max(as3)/20)
        as3(n)=0;
    end
    if (as4(n)<max(as4)/20)
        as4(n)=0;
    end
    if (as5(n)<max(as5)/20)
        as5(n)=0;
    end
    if (as6(n)<max(as6)/20)
        as6(n)=0;
    end
    if (as7(n)<max(as7)/20)
        as7(n)=0;
    end
    if (as8(n)<max(as8)/20)
        as8(n)=0;
    end
    if (as9(n)<max(as9)/20)
        as9(n)=0;
    end
    if (as10(n)<max(as10)/20)
        as10(n)=0;
    end
    if (as11(n)<max(as11)/20)
        as11(n)=0;
    end
    if (as12(n)<max(as12)/20)
        as12(n)=0;
    end
    if (as13(n)<max(as13)/20)
        as13(n)=0;
    end
    if (as14(n)<max(as14)/20)
        as14(n)=0;
    end
    if (bs0(n)<max(bs0)/20)
        bs0(n)=0;
    end
    if (bs1(n)<max(bs1)/20)
        bs1(n)=0;
    end
    if (bs2(n)<max(bs2)/20)
        bs2(n)=0;
    end
    if (bs3(n)<max(bs3)/20)
        bs3(n)=0;
    end
end

```

```

end
if (bs4(n)<max(bs4)/20)
    bs4(n)=0;
end
if (bs5(n)<max(bs5)/20)
    bs5(n)=0;
end
if (bs6(n)<max(bs6)/20)
    bs6(n)=0;
end
if (bs7(n)<max(bs7)/20)
    bs7(n)=0;
end
if (bs8(n)<max(bs8)/20)
    bs8(n)=0;
end
if (bs9(n)<max(bs9)/20)
    bs9(n)=0;
end
if (bs10(n)<max(bs10)/20)
    bs10(n)=0;
end
end

; detector de frecuencias de pico
fa1=[]; fa2=[]; fa3=[]; fa4=[]; fa5=[]; fa6=[]; fa7=[]; fa8=[]; fa9=[]; fa10=[];
fa11=[]; fa12=[]; fa13=[]; fa14=[]; fb0=[]; fb1=[]; fb2=[]; fb3=[]; fb4=[]; fb5=[];
fb6=[]; fb7=[]; fb8=[]; fb9=[]; fb10=[]; fb11=[];
for n=2:(na/2)+1
    if (as1(n)>as1(n-1))
        if (as1(n)>as1(n+1))
            fa1=[fa1 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if (as2(n)>as2(n-1))
        if (as2(n)>as2(n+1))
            fa2=[fa2 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if (as3(n)>as3(n-1))
        if (as3(n)>as3(n+1))
            fa3=[fa3 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if (as4(n)>as4(n-1))
        if (as4(n)>as4(n+1))
            fa4=[fa4 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if (as5(n)>as5(n-1))
        if (as5(n)>as5(n+1))
            fa5=[fa5 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if (as6(n)>as6(n-1))
        if (as6(n)>as6(n+1))
            fa6=[fa6 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
end

```

```

    end
end
if(as7(n)>as7(n-1))
    if(as7(n)>as7(n+1))
        fa7=[fa7 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as8(n)>as8(n-1))
    if(as8(n)>as8(n+1))
        fa8=[fa8 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as9(n)>as9(n-1))
    if(as9(n)>as9(n+1))
        fa9=[fa9 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as10(n)>as10(n-1))
    if(as10(n)>as10(n+1))
        fa10=[fa10 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as11(n)>as11(n-1))
    if(as11(n)>as11(n+1))
        fa11=[fa11 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as12(n)>as12(n-1))
    if(as12(n)>as12(n+1))
        fa12=[fa12 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as13(n)>as13(n-1))
    if(as13(n)>as13(n+1))
        fa13=[fa13 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(as14(n)>as14(n-1))
    if(as14(n)>as14(n+1))
        fa14=[fa14 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(bs0(n)>bs0(n-1))
    if(bs0(n)>bs0(n+1))
        fb0=[fb0 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(bs1(n)>bs1(n-1))
    if(bs1(n)>bs1(n+1))
        fb1=[fb1 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
if(bs2(n)>bs2(n-1))
    if(bs2(n)>bs2(n+1))
        fb2=[fb2 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
    end
end
end

```

```

    if(bs3(n)>bs3(n-1))
        if(bs3(n)>bs3(n+1))
            fb3=[fb3 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs4(n)>bs4(n-1))
        if(bs4(n)>bs4(n+1))
            fb4=[fb4 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs5(n)>bs5(n-1))
        if(bs5(n)>bs5(n+1))
            fb5=[fb5 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs6(n)>bs6(n-1))
        if(bs6(n)>bs6(n+1))
            fb6=[fb6 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs7(n)>bs7(n-1))
        if(bs7(n)>bs7(n+1))
            fb7=[fb7 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs8(n)>bs8(n-1))
        if(bs8(n)>bs8(n+1))
            fb8=[fb8 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs9(n)>bs9(n-1))
        if(bs9(n)>bs9(n+1))
            fb9=[fb9 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
    if(bs10(n)>bs10(n-1))
        if(bs10(n)>bs10(n+1))
            fb10=[fb10 (n-2)*fa/na (n-1)*fa/na n*fa/na (n+1)*fa/na (n+2)*fa/na];
        end
    end
end
nd
picos_x=[fa1 fa2 fa3 fa4 fa5 fa6 fa7 fa8 fa9 fa10 fa11 fa12 fa13 fa14];
picos_y=[fb0 fb1 fb2 fb3 fb4 fb5 fb6 fb7 fb8 fb9 fb10];
picos_x=sort(fpicos_x);
picos_y=sort(fpicos_y);
x=[];
y=[];
or n=1:na/2
    ax=find(fpicos_x==n);
    ay=find(fpicos_y==n);
    if(length(ax)>6)
        fx=[fx n];
    end
    if(length(ay)>4)
        fy=[fy n];
    end
end
nd

```

```

as1=abs(xs1);
as2=abs(xs2);
as3=abs(xs3);
as4=abs(xs4);
as5=abs(xs5);
as6=abs(xs6);
as7=abs(xs7);
as8=abs(xs8);
as9=abs(xs9);
as10=abs(xs10);
as11=abs(xs11);
as12=abs(xs12);
as13=abs(xs13);
as14=abs(xs14);
as0=abs(ys0);
as1=abs(ys1);
as2=abs(ys2);
as3=abs(ys3);
as4=abs(ys4);
as5=abs(ys5);
as6=abs(ys6);
as7=abs(ys7);
as8=abs(ys8);
as9=abs(ys9);
as10=abs(ys10);

```

Modulo 5: Selecao das frequencias naturais atraves de interface com usuario:

```

stop=1;
while(stop~=0)
    disp(' Candidatas a frequencias naturais: ');
    disp(' ')
    display(fx);
    disp(' ')
    display(fy);
    disp(' ')
    disp(' Escolha os modos que deseja ver: ');
    disp(' 0 - modos de placa 01, 02 e 03 ');
    disp(' 1 - modos de placa 11, 21 e 31 ');
    disp(' 2 - modos de placa 12, 22 e 32 ');
    resp=3;
    while(resp~=0 & resp~=1 & resp~=2)
        resp=input(' 0, 1 ou 2: ');
    end
    if(resp==0)
        fpico1=input(' Frequencia natural 1 - Analitica: 52,9 Hz; Experimental: ');
        fpico2=input(' Frequencia natural 2 - Analitica: 214,2 Hz; Experimental: ');
        fpico3=input(' Frequencia natural 3 - Analitica: 484,8 Hz; Experimental: ');
        f1=52.9;
        f2=214.2;
        f3=484.8;
    end
    if(resp==1)
        fpico1=input(' Frequencia natural 11 - Analitica: 121,1 Hz; Experimental: ');
        fpico2=input(' Frequencia natural 21 - Analitica: 303,5 Hz; Experimental: ');
        fpico3=input(' Frequencia natural 31 - Analitica: 581,8 Hz; Experimental: ');
    end
end

```

[illegible]

```

rys4_11(i)=rys4(fn(1)*na/fa-8+i);
iys4_11(i)=iys4(fn(1)*na/fa-8+i);
rys5_11(i)=rys5(fn(1)*na/fa-8+i);
iys5_11(i)=iys5(fn(1)*na/fa-8+i);
rys6_11(i)=rys6(fn(1)*na/fa-8+i);
iys6_11(i)=iys6(fn(1)*na/fa-8+i);
rys7_11(i)=rys7(fn(1)*na/fa-8+i);
iys7_11(i)=iys7(fn(1)*na/fa-8+i);
rys8_11(i)=rys8(fn(1)*na/fa-8+i);
iys8_11(i)=iys8(fn(1)*na/fa-8+i);
rys9_11(i)=rys9(fn(1)*na/fa-8+i);
iys9_11(i)=iys9(fn(1)*na/fa-8+i);
rys10_11(i)=rys10(fn(1)*na/fa-8+i);
iys10_11(i)=iys10(fn(1)*na/fa-8+i);
% 2a. frecuencia natural
rxs1_21(i)=rxs1(fn(2)*na/fa-8+i);
ixs1_21(i)=ixs1(fn(2)*na/fa-8+i);
rxs2_21(i)=rxs2(fn(2)*na/fa-8+i);
ixs2_21(i)=ixs2(fn(2)*na/fa-8+i);
rxs3_21(i)=rxs3(fn(2)*na/fa-8+i);
ixs3_21(i)=ixs3(fn(2)*na/fa-8+i);
rxs4_21(i)=rxs4(fn(2)*na/fa-8+i);
ixs4_21(i)=ixs4(fn(2)*na/fa-8+i);
rxs5_21(i)=rxs5(fn(2)*na/fa-8+i);
ixs5_21(i)=ixs5(fn(2)*na/fa-8+i);
rxs6_21(i)=rxs6(fn(2)*na/fa-8+i);
ixs6_21(i)=ixs6(fn(2)*na/fa-8+i);
rxs7_21(i)=rxs7(fn(2)*na/fa-8+i);
ixs7_21(i)=ixs7(fn(2)*na/fa-8+i);
rxs8_21(i)=rxs8(fn(2)*na/fa-8+i);
ixs8_21(i)=ixs8(fn(2)*na/fa-8+i);
rxs9_21(i)=rxs9(fn(2)*na/fa-8+i);
ixs9_21(i)=ixs9(fn(2)*na/fa-8+i);
rxs10_21(i)=rxs10(fn(2)*na/fa-8+i);
ixs10_21(i)=ixs10(fn(2)*na/fa-8+i);
rxs11_21(i)=rxs11(fn(2)*na/fa-8+i);
ixs11_21(i)=ixs11(fn(2)*na/fa-8+i);
rxs12_21(i)=rxs12(fn(2)*na/fa-8+i);
ixs12_21(i)=ixs12(fn(2)*na/fa-8+i);
rxs13_21(i)=rxs13(fn(2)*na/fa-8+i);
ixs13_21(i)=ixs13(fn(2)*na/fa-8+i);
rxs14_21(i)=rxs14(fn(2)*na/fa-8+i);
ixs14_21(i)=ixs14(fn(2)*na/fa-8+i);
rys0_21(i)=rys0(fn(2)*na/fa-8+i);
iys0_21(i)=iys0(fn(2)*na/fa-8+i);
rys1_21(i)=rys1(fn(2)*na/fa-8+i);
iys1_21(i)=iys1(fn(2)*na/fa-8+i);
rys2_21(i)=rys2(fn(2)*na/fa-8+i);
iys2_21(i)=iys2(fn(2)*na/fa-8+i);
rys3_21(i)=rys3(fn(2)*na/fa-8+i);
iys3_21(i)=iys3(fn(2)*na/fa-8+i);
rys4_21(i)=rys4(fn(2)*na/fa-8+i);
iys4_21(i)=iys4(fn(2)*na/fa-8+i);
rys5_21(i)=rys5(fn(2)*na/fa-8+i);
iys5_21(i)=iys5(fn(2)*na/fa-8+i);
rys6_21(i)=rys6(fn(2)*na/fa-8+i);
iys6_21(i)=iys6(fn(2)*na/fa-8+i);

```

```

rys7_21(i)=rys7(fn(2)*na/fa-8+i);
iys7_21(i)=iys7(fn(2)*na/fa-8+i);
rys8_21(i)=rys8(fn(2)*na/fa-8+i);
iys8_21(i)=iys8(fn(2)*na/fa-8+i);
rys9_21(i)=rys9(fn(2)*na/fa-8+i);
iys9_21(i)=iys9(fn(2)*na/fa-8+i);
rys10_21(i)=rys10(fn(2)*na/fa-8+i);
iys10_21(i)=iys10(fn(2)*na/fa-8+i);
% 3a. frecuencia natural
rxs1_31(i)=rxs1(fn(3)*na/fa-8+i);
ixs1_31(i)=ixs1(fn(3)*na/fa-8+i);
rxs2_31(i)=rxs2(fn(3)*na/fa-8+i);
ixs2_31(i)=ixs2(fn(3)*na/fa-8+i);
rxs3_31(i)=rxs3(fn(3)*na/fa-8+i);
ixs3_31(i)=ixs3(fn(3)*na/fa-8+i);
rxs4_31(i)=rxs4(fn(3)*na/fa-8+i);
ixs4_31(i)=ixs4(fn(3)*na/fa-8+i);
rxs5_31(i)=rxs5(fn(3)*na/fa-8+i);
ixs5_31(i)=ixs5(fn(3)*na/fa-8+i);
rxs6_31(i)=rxs6(fn(3)*na/fa-8+i);
ixs6_31(i)=ixs6(fn(3)*na/fa-8+i);
rxs7_31(i)=rxs7(fn(3)*na/fa-8+i);
ixs7_31(i)=ixs7(fn(3)*na/fa-8+i);
rxs8_31(i)=rxs8(fn(3)*na/fa-8+i);
ixs8_31(i)=ixs8(fn(3)*na/fa-8+i);
rxs9_31(i)=rxs9(fn(3)*na/fa-8+i);
ixs9_31(i)=ixs9(fn(3)*na/fa-8+i);
rxs10_31(i)=rxs10(fn(3)*na/fa-8+i);
ixs10_31(i)=ixs10(fn(3)*na/fa-8+i);
rxs11_31(i)=rxs11(fn(3)*na/fa-8+i);
ixs11_31(i)=ixs11(fn(3)*na/fa-8+i);
rxs12_31(i)=rxs12(fn(3)*na/fa-8+i);
ixs12_31(i)=ixs12(fn(3)*na/fa-8+i);
rxs13_31(i)=rxs13(fn(3)*na/fa-8+i);
ixs13_31(i)=ixs13(fn(3)*na/fa-8+i);
rxs14_31(i)=rxs14(fn(3)*na/fa-8+i);
ixs14_31(i)=ixs14(fn(3)*na/fa-8+i);
rys0_31(i)=rys0(fn(3)*na/fa-8+i);
iys0_31(i)=iys0(fn(3)*na/fa-8+i);
rys1_31(i)=rys1(fn(3)*na/fa-8+i);
iys1_31(i)=iys1(fn(3)*na/fa-8+i);
rys2_31(i)=rys2(fn(3)*na/fa-8+i);
iys2_31(i)=iys2(fn(3)*na/fa-8+i);
rys3_31(i)=rys3(fn(3)*na/fa-8+i);
iys3_31(i)=iys3(fn(3)*na/fa-8+i);
rys4_31(i)=rys4(fn(3)*na/fa-8+i);
iys4_31(i)=iys4(fn(3)*na/fa-8+i);
rys5_31(i)=rys5(fn(3)*na/fa-8+i);
iys5_31(i)=iys5(fn(3)*na/fa-8+i);
rys6_31(i)=rys6(fn(3)*na/fa-8+i);
iys6_31(i)=iys6(fn(3)*na/fa-8+i);
rys7_31(i)=rys7(fn(3)*na/fa-8+i);
iys7_31(i)=iys7(fn(3)*na/fa-8+i);
rys8_31(i)=rys8(fn(3)*na/fa-8+i);
iys8_31(i)=iys8(fn(3)*na/fa-8+i);
rys9_31(i)=rys9(fn(3)*na/fa-8+i);
iys9_31(i)=iys9(fn(3)*na/fa-8+i);

```

```

    rys10_31(i)=rys10(fn(3)*na/fa-8+i);
    iys10_31(i)=iys10(fn(3)*na/fa-8+i);
end
% escolha dos pontos dos circulos de Nyquist para visualizacao grafica
ptox=15; ptoy=15;
while(ptox<1|ptox>14)
    disp(' Escolha a ordenada (de 1 a 14) do ponto da placa que deseja ver o ');
    disp('Circulo de Nyquist (Exemplo: pto (5,1), digite 5):');
    ptox=input(' Ordenada (x): ');
end
while(ptoy<0|ptoy>10)
    disp(' Escolha a abscissa (de 0 a 10) do ponto da placa que deseja ver o ');
    disp('Circulo de Nyquist (Exemplo: pto (3,5), digite 5):');
    ptoy=input(' Abscissa (y): ');
    disp(' ')
end
if(ptox==1)
    rr1=rxs1_11;
    ii1=ixs1_11;
    rr2=rxs1_21;
    ii2=ixs1_21;
    rr3=rxs1_31;
    ii3=ixs1_31;
end
if(ptox==2)
    rr1=rxs2_11;
    ii1=ixs2_11;
    rr2=rxs2_21;
    ii2=ixs2_21;
    rr3=rxs2_31;
    ii3=ixs2_31;
end
if(ptox==3)
    rr1=rxs3_11;
    ii1=ixs3_11;
    rr2=rxs3_21;
    ii2=ixs3_21;
    rr3=rxs3_31;
    ii3=ixs3_31;
end
if(ptox==4)
    rr1=rxs4_11;
    ii1=ixs4_11;
    rr2=rxs4_21;
    ii2=ixs4_21;
    rr3=rxs4_31;
    ii3=ixs4_31;
end
if(ptox==5)
    rr1=rxs5_11;
    ii1=ixs5_11;
    rr2=rxs5_21;
    ii2=ixs5_21;
    rr3=rxs5_31;
    ii3=ixs5_31;
end
if(ptox==6)

```

```

        rr1=rxs6_11;
        ii1=ixs6_11;
        rr2=rxs6_21;
        ii2=ixs6_21;
        rr3=rxs6_31;
        ii3=ixs6_31;
    end
    if(ptox==7)
        rr1=rxs7_11;
        ii1=ixs7_11;
        rr2=rxs7_21;
        ii2=ixs7_21;
        rr3=rxs7_31;
        ii3=ixs7_31;
    end
    if(ptox==8)
        rr1=rxs8_11;
        ii1=ixs8_11;
        rr2=rxs8_21;
        ii2=ixs8_21;
        rr3=rxs8_31;
        ii3=ixs8_31;
    end
    if(ptox==9)
        rr1=rxs9_11;
        ii1=ixs9_11;
        rr2=rxs9_21;
        ii2=ixs9_21;
        rr3=rxs9_31;
        ii3=ixs9_31;
    end
    if(ptox==10)
        rr1=rxs10_11;
        ii1=ixs10_11;
        rr2=rxs10_21;
        ii2=ixs10_21;
        rr3=rxs10_31;
        ii3=ixs10_31;
    end
    if(ptox==11)
        rr1=rxs11_11;
        ii1=ixs11_11;
        rr2=rxs11_21;
        ii2=ixs11_21;
        rr3=rxs11_31;
        ii3=ixs11_31;
    end
    if(ptox==12)
        rr1=rxs12_11;
        ii1=ixs12_11;
        rr2=rxs12_21;
        ii2=ixs12_21;
        rr3=rxs12_31;
        ii3=ixs12_31;
    end
    if(ptox==13)
        rr1=rxs13_11;

```

```

    ii1=ixs13_11;
    rr2=rxs13_21;
    ii2=ixs13_21;
    rr3=rxs13_31;
    ii3=ixs13_31;
end
if(ptox==14)
    rr1=rxs14_11;
    ii1=ixs14_11;
    rr2=rxs14_21;
    ii2=ixs14_21;
    rr3=rxs14_31;
    ii3=ixs14_31;
end
nyquist(rr1,ii1);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr1(1+2*(N-A-1):16-2*(1+A)),ii1(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr1(1+2*(N-A):16-2*A),ii1(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr1(1+2*(N-A-1):16-2*(1+A)),ii1(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
end

```

Modulo 7: Saída grafica dos circulos de Nyquist em torno das freqs naturais:

```

figure(14)

lot(rr1(1),ii1(1),'r.',rr1(2),ii1(2),'r+',rr1(3),ii1(3),'rx',rr1(4),ii1(4),'r^',rr1(5),
ii1(5),'rs',rr1(6),ii1(6),'rp',rr1(7),ii1(7),'rh',rr1(9),ii1(9),'gh',rr1(10),ii1(10),'
p',rr1(11),ii1(11),'gs',rr1(12),ii1(12),'g^',rr1(13),ii1(13),'gx',rr1(14),ii1(14),'g+'
rr1(15),ii1(15),'g.',rr1(16),ii1(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na
,rr1(8),ii1(8),'k*'))
title(['Circulo de Nyquist - frecuencia natural 11 =',num2str(fpicol1),' Hz - ponto
',num2str(ptox),'1). Correlacao: ',num2str(rq)]);

legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');
nyquist(rr2,ii2);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr2(1+2*(N-A-1):16-2*(1+A)),ii2(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr2(1+2*(N-A):16-2*A),ii2(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr2(1+2*(N-A-1):16-2*(1+A)),ii2(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
end

```

```

figure(15)

plot(rr2(1),ii2(1),'r.',rr2(2),ii2(2),'r+',rr2(3),ii2(3),'rx',rr2(4),ii2(4),'r^',rr2(5),
ii2(5),'rs',rr2(6),ii2(6),'rp',rr2(7),ii2(7),'rh',rr2(8),ii2(8),'gh',rr2(9),ii2(9),'gh',rr2(10),ii2(10),'
gp',rr2(11),ii2(11),'gs',rr2(12),ii2(12),'g^',rr2(13),ii2(13),'gx',rr2(14),ii2(14),'g+',
rr2(15),ii2(15),'g.',rr2(16),ii2(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),
rr2(8),ii2(8),'k*')
title(['Circulo de Nyquist - frequencia natural 21 =',num2str(fpico21),' Hz - ponto
',num2str(ptox),'1). Correlacao: ',num2str(rq)]);

legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
.. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');
nyquist(rr3,ii3);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr3(1+2*(N-A-1):16-2*(1+A)),ii3(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr3(1+2*(N-A):16-2*A),ii3(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr3(1+2*(N-A-1):16-2*(1+A)),ii3(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
end
figure(16)

plot(rr3(1),ii3(1),'r.',rr3(2),ii3(2),'r+',rr3(3),ii3(3),'rx',rr3(4),ii3(4),'r^',rr3(5),
ii3(5),'rs',rr3(6),ii3(6),'rp',rr3(7),ii3(7),'rh',rr3(8),ii3(8),'gh',rr3(9),ii3(9),'gh',rr3(10),ii3(10),'
p',rr3(11),ii3(11),'gs',rr3(12),ii3(12),'g^',rr3(13),ii3(13),'gx',rr3(14),ii3(14),'g+',
rr3(15),ii3(15),'g.',rr3(16),ii3(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na),
rr3(8),ii3(8),'k*')
title(['Circulo de Nyquist - frequencia natural 31 =',num2str(fpico31),' Hz - ponto
',num2str(ptox),'1). Correlacao: ',num2str(rq)]);

legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
.. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');

if(ptoy==0)
    rr1=rys0_11;
    ii1=iys0_11;
    rr2=rys0_21;
    ii2=iys0_21;
    rr3=rys0_31;
    ii3=iys0_31;
end
if(ptoy==1)
    rr1=rys1_11;
    ii1=iys1_11;
    rr2=rys1_21;
    ii2=iys1_21;
    rr3=rys1_31;
    ii3=iys1_31;
end

```

```

end
if(ptoy==2)
    rr1=rys2_11;
    ii1=iys2_11;
    rr2=rys2_21;
    ii2=iys2_21;
    rr3=rys2_31;
    ii3=iys2_31;
end
if(ptoy==3)
    rr1=rys3_11;
    ii1=iys3_11;
    rr2=rys3_21;
    ii2=iys3_21;
    rr3=rys3_31;
    ii3=iys3_31;
end
if(ptoy==4)
    rr1=rys4_11;
    ii1=iys4_11;
    rr2=rys4_21;
    ii2=iys4_21;
    rr3=rys4_31;
    ii3=iys4_31;
end
if(ptoy==5)
    rr1=rys5_11;
    ii1=iys5_11;
    rr2=rys5_21;
    ii2=iys5_21;
    rr3=rys5_31;
    ii3=iys5_31;
end
if(ptoy==6)
    rr1=rys6_11;
    ii1=iys6_11;
    rr2=rys6_21;
    ii2=iys6_21;
    rr3=rys6_31;
    ii3=iys6_31;
end
if(ptoy==7)
    rr1=rys7_11;
    ii1=iys7_11;
    rr2=rys7_21;
    ii2=iys7_21;
    rr3=rys7_31;
    ii3=iys7_31;
end
if(ptoy==8)
    rr1=rys8_11;
    ii1=iys8_11;
    rr2=rys8_21;
    ii2=iys8_21;
    rr3=rys8_31;
    ii3=iys8_31;
end

```

```

if(ptoy==9)
    rr1=rys9_11;
    ii1=iys9_11;
    rr2=rys9_21;
    ii2=iys9_21;
    rr3=rys9_31;
    ii3=iys9_31;
end
if(ptoy==10)
    rr1=rys10_11;
    ii1=iys10_11;
    rr2=rys10_21;
    ii2=iys10_21;
    rr3=rys10_31;
    ii3=iys10_31;
end
nyquist(rr1,ii1);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr1(1+2*(N-A-1):16-2*(1+A)),ii1(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr1(1+2*(N-A):16-2*A),ii1(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr1(1+2*(N-A-1):16-2*(1+A)),ii1(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
figure(17)

lot(rr1(1),ii1(1),'r.',rr1(2),ii1(2),'r+',rr1(3),ii1(3),'rx',rr1(4),ii1(4),'r^',rr1(5),
ii1(5),'rs',rr1(6),ii1(6),'rp',rr1(7),ii1(7),'rh',rr1(9),ii1(9),'gh',rr1(10),ii1(10),'
p',rr1(11),ii1(11),'gs',rr1(12),ii1(12),'g^',rr1(13),ii1(13),'gx',rr1(14),ii1(14),'g+'
rr1(15),ii1(15),'g.',rr1(16),ii1(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na
,rr1(8),ii1(8),'k*')
title(['Circulo de Nyquist - frequencia natural 11 =',num2str(fpicoll),' Hz - ponto
3,',num2str(ptoy),'). Correlacao: ',num2str(rq)]);

legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');
nyquist(rr2,ii2);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr2(1+2*(N-A-1):16-2*(1+A)),ii2(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr2(1+2*(N-A):16-2*A),ii2(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr2(1+2*(N-A-1):16-2*(1+A)),ii2(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
figure(18)

```

```
plot(rr2(1),ii2(1),'r.',rr2(2),ii2(2),'r+',rr2(3),ii2(3),'rx',rr2(4),ii2(4),'r^',rr2(5),
ii2(5),'rs',rr2(6),ii2(6),'rp',rr2(7),ii2(7),'rh',rr2(9),ii2(9),'gh',rr2(10),ii2(10),'
p',rr2(11),ii2(11),'gs',rr2(12),ii2(12),'g^',rr2(13),ii2(13),'gx',rr2(14),ii2(14),'g+'
rr2(15),ii2(15),'g.',rr2(16),ii2(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na
,rr2(8),ii2(8),'k*')
title(['Circulo de Nyquist - frequencia natural 21 =',num2str(fpico21),' Hz - ponto
3,',num2str(ptoy),'). Correlacao: ',num2str(rq)]);
```

```
legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');
nyquist(rr3,ii3);
N=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rr3(1+2*(N-A-1):16-2*(1+A)),ii3(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rr3(1+2*(N-A):16-2*A),ii3(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rr3(1+2*(N-A-1):16-2*(1+A)),ii3(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
figure(19)
```

```
plot(rr3(1),ii3(1),'r.',rr3(2),ii3(2),'r+',rr3(3),ii3(3),'rx',rr3(4),ii3(4),'r^',rr3(5),
ii3(5),'rs',rr3(6),ii3(6),'rp',rr3(7),ii3(7),'rh',rr3(9),ii3(9),'gh',rr3(10),ii3(10),'
p',rr3(11),ii3(11),'gs',rr3(12),ii3(12),'g^',rr3(13),ii3(13),'gx',rr3(14),ii3(14),'g+'
rr3(15),ii3(15),'g.',rr3(16),ii3(16),'go',x0+raio*sin(6.3*t./na),y0+raio*cos(6.3*t./na
,rr3(8),ii3(8),'k*')
title(['Circulo de Nyquist - frequencia natural 31 =',num2str(fpico31),' Hz - ponto
3,',num2str(ptoy),'). Correlacao: ',num2str(rq)]);
```

```
legend('1','2','3','4','5','6','7','9','10','11','12','13','14','15','16','interp.','re
. a fn',0);
xlabel('parte real da amplitude');
ylabel('parte imaginaria da amplitude');
```

Modulo 8: Determinacao das constantes de amortecimento (damping ratio: ksi)

```
rr01=[]; alfa0=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[];
lfa7=[]; alfa8=[]; alfa9=[]; alfa10=[];
nyquist(rys0_11,iys0_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys0_11(1+2*(N-A-1):16-2*(1+A)),iys0_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys0_11(1+2*(N-A):16-2*A),iys0_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys0_11(1+2*(N-A-1):16-2*(1+A)),iys0_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
```

```

end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys0_11(n+1))^2+(y0-iys0_11(n+1))^2);
    bb=sqrt((x0-rys0_11(n))^2+(y0-iys0_11(n))^2);
    cc=sqrt((rys0_11(n+1)-rys0_11(n))^2+(iys0_11(n+1)-iys0_11(n))^2);
    alfa0(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrrol=[Errol rq];
nyquist(rys1_11,iys1_11)
l=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys1_11(1+2*(N-A-1):16-2*(1+A)),iys1_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys1_11(1+2*(N-A):16-2*A),iys1_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys1_11(1+2*(N-A-1):16-2*(1+A)),iys1_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys1_11(n+1))^2+(y0-iys1_11(n+1))^2);
    bb=sqrt((x0-rys1_11(n))^2+(y0-iys1_11(n))^2);
    cc=sqrt((rys1_11(n+1)-rys1_11(n))^2+(iys1_11(n+1)-iys1_11(n))^2);
    alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrrol=[Errol rq];
nyquist(rys2_11,iys2_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys2_11(1+2*(N-A-1):16-2*(1+A)),iys2_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys2_11(1+2*(N-A):16-2*A),iys2_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys2_11(1+2*(N-A-1):16-2*(1+A)),iys2_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys2_11(n+1))^2+(y0-iys2_11(n+1))^2);
    bb=sqrt((x0-rys2_11(n))^2+(y0-iys2_11(n))^2);
    cc=sqrt((rys2_11(n+1)-rys2_11(n))^2+(iys2_11(n+1)-iys2_11(n))^2);
    alfa2(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrrol=[Errol rq];
nyquist(rys3_11,iys3_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys3_11(1+2*(N-A-1):16-2*(1+A)),iys3_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys3_11(1+2*(N-A):16-2*A),iys3_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)

```

```

nyquist(rys3_11(1+2*(N-A-1):16-2*(1+A)),iys3_11(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
end
for n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys3_11(n+1))^2+(y0-iys3_11(n+1))^2);
bb=sqrt((x0-rys3_11(n))^2+(y0-iys3_11(n))^2);
cc=sqrt((rys3_11(n+1)-rys3_11(n))^2+(iys3_11(n+1)-iys3_11(n))^2);
alfa3(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrol=[Errol rq];
nyquist(rys4_11,iys4_11)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rys4_11(1+2*(N-A-1):16-2*(1+A)),iys4_11(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rys4_11(1+2*(N-A):16-2*A),iys4_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rys4_11(1+2*(N-A-1):16-2*(1+A)),iys4_11(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
end
for n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys4_11(n+1))^2+(y0-iys4_11(n+1))^2);
bb=sqrt((x0-rys4_11(n))^2+(y0-iys4_11(n))^2);
cc=sqrt((rys4_11(n+1)-rys4_11(n))^2+(iys4_11(n+1)-iys4_11(n))^2);
alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrol=[Errol rq];
nyquist(rys5_11,iys5_11)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rys5_11(1+2*(N-A-1):16-2*(1+A)),iys5_11(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rys5_11(1+2*(N-A):16-2*A),iys5_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rys5_11(1+2*(N-A-1):16-2*(1+A)),iys5_11(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
end
for n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys5_11(n+1))^2+(y0-iys5_11(n+1))^2);
bb=sqrt((x0-rys5_11(n))^2+(y0-iys5_11(n))^2);
cc=sqrt((rys5_11(n+1)-rys5_11(n))^2+(iys5_11(n+1)-iys5_11(n))^2);
alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rrol=[Errol rq];
nyquist(rys6_11,iys6_11)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rys6_11(1+2*(N-A-1):16-2*(1+A)),iys6_11(1+2*(N-A-1):16-2*(1+A)));
a=rq;

```

```

nyquist(rys6_11(1+2*(N-A):16-2*A),iys6_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rys6_11(1+2*(N-A-1):16-2*(1+A)),iys6_11(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys6_11(n+1))^2+(y0-iys6_11(n+1))^2);
bb=sqrt((x0-rys6_11(n))^2+(y0-iys6_11(n))^2);
cc=sqrt((rys6_11(n+1)-rys6_11(n))^2+(iys6_11(n+1)-iys6_11(n))^2);
alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[Error1 rq];
nyquist(rys7_11,iys7_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys7_11(1+2*(N-A-1):16-2*(1+A)),iys7_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys7_11(1+2*(N-A):16-2*A),iys7_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys7_11(1+2*(N-A-1):16-2*(1+A)),iys7_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys7_11(n+1))^2+(y0-iys7_11(n+1))^2);
bb=sqrt((x0-rys7_11(n))^2+(y0-iys7_11(n))^2);
cc=sqrt((rys7_11(n+1)-rys7_11(n))^2+(iys7_11(n+1)-iys7_11(n))^2);
alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[Error1 rq];
nyquist(rys8_11,iys8_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys8_11(1+2*(N-A-1):16-2*(1+A)),iys8_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys8_11(1+2*(N-A):16-2*A),iys8_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys8_11(1+2*(N-A-1):16-2*(1+A)),iys8_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys8_11(n+1))^2+(y0-iys8_11(n+1))^2);
bb=sqrt((x0-rys8_11(n))^2+(y0-iys8_11(n))^2);
cc=sqrt((rys8_11(n+1)-rys8_11(n))^2+(iys8_11(n+1)-iys8_11(n))^2);
alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[Error1 rq];
nyquist(rys9_11,iys9_11)
=0; A=0;
while(rq<.7&N<3)

```

```

N=N+1;
nyquist(rys9_11(1+2*(N-A-1):16-2*(1+A)),iys9_11(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rys9_11(1+2*(N-A):16-2*A),iys9_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rys9_11(1+2*(N-A-1):16-2*(1+A)),iys9_11(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys9_11(n+1))^2+(y0-iys9_11(n+1))^2);
    bb=sqrt((x0-rys9_11(n))^2+(y0-iys9_11(n))^2);
    cc=sqrt((rys9_11(n+1)-rys9_11(n))^2+(iys9_11(n+1)-iys9_11(n))^2);
    alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro1=[Erro1 rq];
nyquist(rys10_11,iys10_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys10_11(1+2*(N-A-1):16-2*(1+A)),iys10_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys10_11(1+2*(N-A):16-2*A),iys10_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys10_11(1+2*(N-A-1):16-2*(1+A)),iys10_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys10_11(n+1))^2+(y0-iys10_11(n+1))^2);
    bb=sqrt((x0-rys10_11(n))^2+(y0-iys10_11(n))^2);
    cc=sqrt((rys10_11(n+1)-rys10_11(n))^2+(iys10_11(n+1)-iys10_11(n))^2);
    alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro1=[Erro1 rq];
sil=(fpicoll^2-(fpicoll-1)^2)./(2*fpicoll*(2*fpicoll-1)*tan([max(alfa0/2) max(alfa1/2)
max(alfa2/2) max(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2)
max(alfa8/2) max(alfa9/2) max(alfa10/2)]))
1=[find(alfa0==max(alfa0)) find(alfa1==max(alfa1)) find(alfa2==max(alfa2))
find(alfa3==max(alfa3)) find(alfa4==max(alfa4)) find(alfa5==max(alfa5))
find(alfa6==max(alfa6)) find(alfa7==max(alfa7)) find(alfa8==max(alfa8))
find(alfa9==max(alfa9)) find(alfa10==max(alfa10))];
erro2=[]; alfa0=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[];
alfa7=[]; alfa8=[]; alfa9=[]; alfa10=[];
nyquist(rys0_21,iys0_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys0_21(1+2*(N-A-1):16-2*(1+A)),iys0_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys0_21(1+2*(N-A):16-2*A),iys0_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys0_21(1+2*(N-A-1):16-2*(1+A)),iys0_21(1+2*(N-A-1):16-2*(1+A)));

```

```

        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys0_21(n+1))^2+(y0-iys0_21(n+1))^2);
bb=sqrt((x0-rys0_21(n))^2+(y0-iys0_21(n))^2);
cc=sqrt((rys0_21(n+1)-rys0_21(n))^2+(iys0_21(n+1)-iys0_21(n))^2);
alfa0(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys1_21,iys1_21)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys1_21(1+2*(N-A-1):16-2*(1+A)),iys1_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys1_21(1+2*(N-A):16-2*A),iys1_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys1_21(1+2*(N-A-1):16-2*(1+A)),iys1_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys1_21(n+1))^2+(y0-iys1_21(n+1))^2);
bb=sqrt((x0-rys1_21(n))^2+(y0-iys1_21(n))^2);
cc=sqrt((rys1_21(n+1)-rys1_21(n))^2+(iys1_21(n+1)-iys1_21(n))^2);
alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys2_21,iys2_21)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys2_21(1+2*(N-A-1):16-2*(1+A)),iys2_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys2_21(1+2*(N-A):16-2*A),iys2_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys2_21(1+2*(N-A-1):16-2*(1+A)),iys2_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys2_21(n+1))^2+(y0-iys2_21(n+1))^2);
bb=sqrt((x0-rys2_21(n))^2+(y0-iys2_21(n))^2);
cc=sqrt((rys2_21(n+1)-rys2_21(n))^2+(iys2_21(n+1)-iys2_21(n))^2);
alfa2(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys3_21,iys3_21)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys3_21(1+2*(N-A-1):16-2*(1+A)),iys3_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys3_21(1+2*(N-A):16-2*A),iys3_21(1+2*(N-A):16-2*A));

```

```

b=rq;
if(a>b)
    nyquist(rys3_21(1+2*(N-A-1):16-2*(1+A)),iys3_21(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys3_21(n+1))^2+(y0-iys3_21(n+1))^2);
    bb=sqrt((x0-rys3_21(n))^2+(y0-iys3_21(n))^2);
    cc=sqrt((rys3_21(n+1)-rys3_21(n))^2+(iys3_21(n+1)-iys3_21(n))^2);
    alfa3(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
Erro2=[Erro2 rq];
nyquist(rys4_21,iys4_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys4_21(1+2*(N-A-1):16-2*(1+A)),iys4_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys4_21(1+2*(N-A):16-2*A),iys4_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys4_21(1+2*(N-A-1):16-2*(1+A)),iys4_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys4_21(n+1))^2+(y0-iys4_21(n+1))^2);
    bb=sqrt((x0-rys4_21(n))^2+(y0-iys4_21(n))^2);
    cc=sqrt((rys4_21(n+1)-rys4_21(n))^2+(iys4_21(n+1)-iys4_21(n))^2);
    alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
nd
Erro2=[Erro2 rq];
nyquist(rys5_21,iys5_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys5_21(1+2*(N-A-1):16-2*(1+A)),iys5_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys5_21(1+2*(N-A):16-2*A),iys5_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys5_21(1+2*(N-A-1):16-2*(1+A)),iys5_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys5_21(n+1))^2+(y0-iys5_21(n+1))^2);
    bb=sqrt((x0-rys5_21(n))^2+(y0-iys5_21(n))^2);
    cc=sqrt((rys5_21(n+1)-rys5_21(n))^2+(iys5_21(n+1)-iys5_21(n))^2);
    alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
nd
Erro2=[Erro2 rq];
nyquist(rys6_21,iys6_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;

```

```

nyquist(rys6_21(1+2*(N-A-1):16-2*(1+A)),iys6_21(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rys6_21(1+2*(N-A):16-2*A),iys6_21(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rys6_21(1+2*(N-A-1):16-2*(1+A)),iys6_21(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys6_21(n+1))^2+(y0-iys6_21(n+1))^2);
bb=sqrt((x0-rys6_21(n))^2+(y0-iys6_21(n))^2);
cc=sqrt((rys6_21(n+1)-rys6_21(n))^2+(iys6_21(n+1)-iys6_21(n))^2);
alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys7_21,iys7_21)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys7_21(1+2*(N-A-1):16-2*(1+A)),iys7_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys7_21(1+2*(N-A):16-2*A),iys7_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys7_21(1+2*(N-A-1):16-2*(1+A)),iys7_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys7_21(n+1))^2+(y0-iys7_21(n+1))^2);
bb=sqrt((x0-rys7_21(n))^2+(y0-iys7_21(n))^2);
cc=sqrt((rys7_21(n+1)-rys7_21(n))^2+(iys7_21(n+1)-iys7_21(n))^2);
alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys8_21,iys8_21)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys8_21(1+2*(N-A-1):16-2*(1+A)),iys8_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys8_21(1+2*(N-A):16-2*A),iys8_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys8_21(1+2*(N-A-1):16-2*(1+A)),iys8_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys8_21(n+1))^2+(y0-iys8_21(n+1))^2);
bb=sqrt((x0-rys8_21(n))^2+(y0-iys8_21(n))^2);
cc=sqrt((rys8_21(n+1)-rys8_21(n))^2+(iys8_21(n+1)-iys8_21(n))^2);
alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[Erro2 rq];
yquist(rys9_21,iys9_21)

```

```

=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys9_21(1+2*(N-A-1):16-2*(1+A)),iys9_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys9_21(1+2*(N-A):16-2*A),iys9_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys9_21(1+2*(N-A-1):16-2*(1+A)),iys9_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys9_21(n+1))^2+(y0-iys9_21(n+1))^2);
bb=sqrt((x0-rys9_21(n))^2+(y0-iys9_21(n))^2);
cc=sqrt((rys9_21(n+1)-rys9_21(n))^2+(iys9_21(n+1)-iys9_21(n))^2);
alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rro2=[Erro2 rq];
yquist(rys10_21,iys10_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys10_21(1+2*(N-A-1):16-2*(1+A)),iys10_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys10_21(1+2*(N-A):16-2*A),iys10_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys10_21(1+2*(N-A-1):16-2*(1+A)),iys10_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys10_21(n+1))^2+(y0-iys10_21(n+1))^2);
bb=sqrt((x0-rys10_21(n))^2+(y0-iys10_21(n))^2);
cc=sqrt((rys10_21(n+1)-rys10_21(n))^2+(iys10_21(n+1)-iys10_21(n))^2);
alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
rro2=[Erro2 rq];
si2=(fpico21^2-(fpico21-1)^2)./(2*fpico21*(2*fpico21-1)*tan([max(alfa0/2) max(alfa1/2)
ax(alfa2/2) max(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2)
ax(alfa8/2) max(alfa9/2) max(alfa10/2)]))
2=[find(alfa0==max(alfa0)) find(alfa1==max(alfa1)) find(alfa2==max(alfa2))
ind(alfa3==max(alfa3)) find(alfa4==max(alfa4)) find(alfa5==max(alfa5))
ind(alfa6==max(alfa6)) find(alfa7==max(alfa7)) find(alfa8==max(alfa8))
ind(alfa9==max(alfa9)) find(alfa10==max(alfa10))];
rro3=[]; alfa0=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[];
lfa7=[]; alfa8=[]; alfa9=[]; alfa10=[];
yquist(rys0_31,iys0_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys0_31(1+2*(N-A-1):16-2*(1+A)),iys0_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys0_31(1+2*(N-A):16-2*A),iys0_31(1+2*(N-A):16-2*A));
    b=rq;

```

```

    if(a>b)
        nyquist(rys0_31(1+2*(N-A-1):16-2*(1+A)),iys0_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys0_31(n+1))^2+(y0-iys0_31(n+1))^2);
bb=sqrt((x0-rys0_31(n))^2+(y0-iys0_31(n))^2);
cc=sqrt((rys0_31(n+1)-rys0_31(n))^2+(iys0_31(n+1)-iys0_31(n))^2);
alfa0(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[Erro3 rq];
yquist(rys1_31,iys1_31)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys1_31(1+2*(N-A-1):16-2*(1+A)),iys1_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys1_31(1+2*(N-A):16-2*A),iys1_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys1_31(1+2*(N-A-1):16-2*(1+A)),iys1_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys1_31(n+1))^2+(y0-iys1_31(n+1))^2);
bb=sqrt((x0-rys1_31(n))^2+(y0-iys1_31(n))^2);
cc=sqrt((rys1_31(n+1)-rys1_31(n))^2+(iys1_31(n+1)-iys1_31(n))^2);
alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[Erro3 rq];
yquist(rys2_31,iys2_31)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys2_31(1+2*(N-A-1):16-2*(1+A)),iys2_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys2_31(1+2*(N-A):16-2*A),iys2_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys2_31(1+2*(N-A-1):16-2*(1+A)),iys2_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys2_31(n+1))^2+(y0-iys2_31(n+1))^2);
bb=sqrt((x0-rys2_31(n))^2+(y0-iys2_31(n))^2);
cc=sqrt((rys2_31(n+1)-rys2_31(n))^2+(iys2_31(n+1)-iys2_31(n))^2);
alfa2(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[Erro3 rq];
yquist(rys3_31,iys3_31)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rys3_31(1+2*(N-A-1):16-2*(1+A)),iys3_31(1+2*(N-A-1):16-2*(1+A)));

```

```

a=rq;
nyquist(rys3_31(1+2*(N-A):16-2*A),iys3_31(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rys3_31(1+2*(N-A-1):16-2*(1+A)),iys3_31(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys3_31(n+1))^2+(y0-iys3_31(n+1))^2);
    bb=sqrt((x0-rys3_31(n))^2+(y0-iys3_31(n))^2);
    cc=sqrt((rys3_31(n+1)-rys3_31(n))^2+(iys3_31(n+1)-iys3_31(n))^2);
    alfa3(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[Erro3 rq];
nyquist(rys4_31,iys4_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys4_31(1+2*(N-A-1):16-2*(1+A)),iys4_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys4_31(1+2*(N-A):16-2*A),iys4_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys4_31(1+2*(N-A-1):16-2*(1+A)),iys4_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys4_31(n+1))^2+(y0-iys4_31(n+1))^2);
    bb=sqrt((x0-rys4_31(n))^2+(y0-iys4_31(n))^2);
    cc=sqrt((rys4_31(n+1)-rys4_31(n))^2+(iys4_31(n+1)-iys4_31(n))^2);
    alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[Erro3 rq];
nyquist(rys5_31,iys5_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys5_31(1+2*(N-A-1):16-2*(1+A)),iys5_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys5_31(1+2*(N-A):16-2*A),iys5_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys5_31(1+2*(N-A-1):16-2*(1+A)),iys5_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rys5_31(n+1))^2+(y0-iys5_31(n+1))^2);
    bb=sqrt((x0-rys5_31(n))^2+(y0-iys5_31(n))^2);
    cc=sqrt((rys5_31(n+1)-rys5_31(n))^2+(iys5_31(n+1)-iys5_31(n))^2);
    alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[Erro3 rq];
nyquist(rys6_31,iys6_31)
=0; A=0;

```

```

while(rq<.7&N<3)
    N=N+1;
    nyquist(rys6_31(1+2*(N-A-1):16-2*(1+A)),iys6_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys6_31(1+2*(N-A):16-2*A),iys6_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys6_31(1+2*(N-A-1):16-2*(1+A)),iys6_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys6_31(n+1))^2+(y0-iys6_31(n+1))^2);
bb=sqrt((x0-rys6_31(n))^2+(y0-iys6_31(n))^2);
cc=sqrt((rys6_31(n+1)-rys6_31(n))^2+(iys6_31(n+1)-iys6_31(n))^2);
alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[Erro3 rq];
yquist(rys7_31,iys7_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys7_31(1+2*(N-A-1):16-2*(1+A)),iys7_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys7_31(1+2*(N-A):16-2*A),iys7_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys7_31(1+2*(N-A-1):16-2*(1+A)),iys7_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys7_31(n+1))^2+(y0-iys7_31(n+1))^2);
bb=sqrt((x0-rys7_31(n))^2+(y0-iys7_31(n))^2);
cc=sqrt((rys7_31(n+1)-rys7_31(n))^2+(iys7_31(n+1)-iys7_31(n))^2);
alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[Erro3 rq];
yquist(rys8_31,iys8_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys8_31(1+2*(N-A-1):16-2*(1+A)),iys8_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys8_31(1+2*(N-A):16-2*A),iys8_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys8_31(1+2*(N-A-1):16-2*(1+A)),iys8_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys8_31(n+1))^2+(y0-iys8_31(n+1))^2);
bb=sqrt((x0-rys8_31(n))^2+(y0-iys8_31(n))^2);
cc=sqrt((rys8_31(n+1)-rys8_31(n))^2+(iys8_31(n+1)-iys8_31(n))^2);
alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd

```

```

erro3=[Erro3 rq];
nyquist(rys9_31,iys9_31)
l=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys9_31(1+2*(N-A-1):16-2*(1+A)),iys9_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys9_31(1+2*(N-A):16-2*A),iys9_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys9_31(1+2*(N-A-1):16-2*(1+A)),iys9_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys9_31(n+1))^2+(y0-iys9_31(n+1))^2);
bb=sqrt((x0-rys9_31(n))^2+(y0-iys9_31(n))^2);
cc=sqrt((rys9_31(n+1)-rys9_31(n))^2+(iys9_31(n+1)-iys9_31(n))^2);
alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[Erro3 rq];
nyquist(rys10_31,iys10_31)
l=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rys10_31(1+2*(N-A-1):16-2*(1+A)),iys10_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rys10_31(1+2*(N-A):16-2*A),iys10_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rys10_31(1+2*(N-A-1):16-2*(1+A)),iys10_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rys10_31(n+1))^2+(y0-iys10_31(n+1))^2);
bb=sqrt((x0-rys10_31(n))^2+(y0-iys10_31(n))^2);
cc=sqrt((rys10_31(n+1)-rys10_31(n))^2+(iys10_31(n+1)-iys10_31(n))^2);
alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[Erro3 rq];
si3=(fpico31^2-(fpico31-1)^2)./(2*fpico31*(2*fpico31-1)*tan([max(alfa0/2) max(alfa1/2)
ax(alfa2/2) max(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2)
ax(alfa8/2) max(alfa9/2) max(alfa10/2)]))
3=[find(alfa0==max(alfa0)) find(alfa1==max(alfa1)) find(alfa2==max(alfa2))
ind(alfa3==max(alfa3)) find(alfa4==max(alfa4)) find(alfa5==max(alfa5))
ind(alfa6==max(alfa6)) find(alfa7==max(alfa7)) find(alfa8==max(alfa8))
ind(alfa9==max(alfa9)) find(alfa10==max(alfa10))];
erro1=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[]; alfa7=[];
alfa8=[]; alfa9=[]; alfa10=[]; alfa11=[]; alfa12=[]; alfa13=[]; alfa14=[];
nyquist(rxs1_11,ixs1_11)
l=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs1_11(1+2*(N-A-1):16-2*(1+A)),ixs1_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;

```

```

nyquist(rxs1_11(1+2*(N-A):16-2*A),ixs1_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rxs1_11(1+2*(N-A-1):16-2*(1+A)),ixs1_11(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
end
for n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs1_11(n+1))^2+(y0-ixs1_11(n+1))^2);
    bb=sqrt((x0-rxs1_11(n))^2+(y0-ixs1_11(n))^2);
    cc=sqrt((rxs1_11(n+1)-rxs1_11(n))^2+(ixs1_11(n+1)-ixs1_11(n))^2);
    alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro1=[erro1 rq];
nyquist(rxs2_11,ixs2_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs2_11(1+2*(N-A-1):16-2*(1+A)),ixs2_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs2_11(1+2*(N-A):16-2*A),ixs2_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs2_11(1+2*(N-A-1):16-2*(1+A)),ixs2_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
erro1=[erro1 rq];
nyquist(rxs3_11,ixs3_11)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs3_11(1+2*(N-A-1):16-2*(1+A)),ixs3_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs3_11(1+2*(N-A):16-2*A),ixs3_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs3_11(1+2*(N-A-1):16-2*(1+A)),ixs3_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
erro1=[erro1 rq];
nyquist(rxs4_11,ixs4_11)
=0; A=0;
while(rq<.7&N<3)

```

```

N=N+1;
nyquist(rxs4_11(1+2*(N-A-1):16-2*(1+A)),ixs4_11(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs4_11(1+2*(N-A):16-2*A),ixs4_11(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rxs4_11(1+2*(N-A-1):16-2*(1+A)),ixs4_11(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs4_11(n+1))^2+(y0-ixs4_11(n+1))^2);
bb=sqrt((x0-rxs4_11(n))^2+(y0-ixs4_11(n))^2);
cc=sqrt((rxs4_11(n+1)-rxs4_11(n))^2+(ixs4_11(n+1)-ixs4_11(n))^2);
alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];
yquist(rxs5_11,ixs5_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs5_11(1+2*(N-A-1):16-2*(1+A)),ixs5_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs5_11(1+2*(N-A):16-2*A),ixs5_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs5_11(1+2*(N-A-1):16-2*(1+A)),ixs5_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs5_11(n+1))^2+(y0-ixs5_11(n+1))^2);
bb=sqrt((x0-rxs5_11(n))^2+(y0-ixs5_11(n))^2);
cc=sqrt((rxs5_11(n+1)-rxs5_11(n))^2+(ixs5_11(n+1)-ixs5_11(n))^2);
alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];
yquist(rxs6_11,ixs6_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs6_11(1+2*(N-A-1):16-2*(1+A)),ixs6_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs6_11(1+2*(N-A):16-2*A),ixs6_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs6_11(1+2*(N-A-1):16-2*(1+A)),ixs6_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs6_11(n+1))^2+(y0-ixs6_11(n+1))^2);
bb=sqrt((x0-rxs6_11(n))^2+(y0-ixs6_11(n))^2);
cc=sqrt((rxs6_11(n+1)-rxs6_11(n))^2+(ixs6_11(n+1)-ixs6_11(n))^2);
alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];

```

```

yquist(rxs7_11,ixs7_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs7_11(1+2*(N-A-1):16-2*(1+A)),ixs7_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs7_11(1+2*(N-A):16-2*A),ixs7_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs7_11(1+2*(N-A-1):16-2*(1+A)),ixs7_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs7_11(n+1))^2+(y0-ixs7_11(n+1))^2);
    bb=sqrt((x0-rxs7_11(n))^2+(y0-ixs7_11(n))^2);
    cc=sqrt((rxs7_11(n+1)-rxs7_11(n))^2+(ixs7_11(n+1)-ixs7_11(n))^2);
    alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rr01=[errol rq];
yquist(rxs8_11,ixs8_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs8_11(1+2*(N-A-1):16-2*(1+A)),ixs8_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs8_11(1+2*(N-A):16-2*A),ixs8_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs8_11(1+2*(N-A-1):16-2*(1+A)),ixs8_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs8_11(n+1))^2+(y0-ixs8_11(n+1))^2);
    bb=sqrt((x0-rxs8_11(n))^2+(y0-ixs8_11(n))^2);
    cc=sqrt((rxs8_11(n+1)-rxs8_11(n))^2+(ixs8_11(n+1)-ixs8_11(n))^2);
    alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rr01=[errol rq];
yquist(rxs9_11,ixs9_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs9_11(1+2*(N-A-1):16-2*(1+A)),ixs9_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs9_11(1+2*(N-A):16-2*A),ixs9_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs9_11(1+2*(N-A-1):16-2*(1+A)),ixs9_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs9_11(n+1))^2+(y0-ixs9_11(n+1))^2);
    bb=sqrt((x0-rxs9_11(n))^2+(y0-ixs9_11(n))^2);
    cc=sqrt((rxs9_11(n+1)-rxs9_11(n))^2+(ixs9_11(n+1)-ixs9_11(n))^2);

```

```

    alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];
yquist(rxs10_11,ixs10_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs10_11(1+2*(N-A-1):16-2*(1+A)),ixs10_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs10_11(1+2*(N-A):16-2*A),ixs10_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs10_11(1+2*(N-A-1):16-2*(1+A)),ixs10_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs10_11(n+1))^2+(y0-ixs10_11(n+1))^2);
    bb=sqrt((x0-rxs10_11(n))^2+(y0-ixs10_11(n))^2);
    cc=sqrt((rxs10_11(n+1)-rxs10_11(n))^2+(ixs10_11(n+1)-ixs10_11(n))^2);
    alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];
yquist(rxs11_11,ixs11_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs11_11(1+2*(N-A-1):16-2*(1+A)),ixs11_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs11_11(1+2*(N-A):16-2*A),ixs11_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs11_11(1+2*(N-A-1):16-2*(1+A)),ixs11_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs11_11(n+1))^2+(y0-ixs11_11(n+1))^2);
    bb=sqrt((x0-rxs11_11(n))^2+(y0-ixs11_11(n))^2);
    cc=sqrt((rxs11_11(n+1)-rxs11_11(n))^2+(ixs11_11(n+1)-ixs11_11(n))^2);
    alfa11(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rrol=[errol rq];
yquist(rxs12_11,ixs12_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs12_11(1+2*(N-A-1):16-2*(1+A)),ixs12_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs12_11(1+2*(N-A):16-2*A),ixs12_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs12_11(1+2*(N-A-1):16-2*(1+A)),ixs12_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A

```

```

aa=sqrt((x0-rxs12_11(n+1))^2+(y0-ixs12_11(n+1))^2);
bb=sqrt((x0-rxs12_11(n))^2+(y0-ixs12_11(n))^2);
cc=sqrt((rxs12_11(n+1)-rxs12_11(n))^2+(ixs12_11(n+1)-ixs12_11(n))^2);
alfa12(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];

nd
rrol=[errol rq];
yquist(rxs13_11,ixs13_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs13_11(1+2*(N-A-1):16-2*(1+A)),ixs13_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs13_11(1+2*(N-A):16-2*A),ixs13_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs13_11(1+2*(N-A-1):16-2*(1+A)),ixs13_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs13_11(n+1))^2+(y0-ixs13_11(n+1))^2);
bb=sqrt((x0-rxs13_11(n))^2+(y0-ixs13_11(n))^2);
cc=sqrt((rxs13_11(n+1)-rxs13_11(n))^2+(ixs13_11(n+1)-ixs13_11(n))^2);
alfa13(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];

nd
rrol=[errol rq];
yquist(rxs14_11,ixs14_11)
=0; A=0;
hile(rq<.7&N<3)
    N=N+1;
    nyquist(rxs14_11(1+2*(N-A-1):16-2*(1+A)),ixs14_11(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs14_11(1+2*(N-A):16-2*A),ixs14_11(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs14_11(1+2*(N-A-1):16-2*(1+A)),ixs14_11(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs14_11(n+1))^2+(y0-ixs14_11(n+1))^2);
bb=sqrt((x0-rxs14_11(n))^2+(y0-ixs14_11(n))^2);
cc=sqrt((rxs14_11(n+1)-rxs14_11(n))^2+(ixs14_11(n+1)-ixs14_11(n))^2);
alfa14(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];

nd
rrol=[errol rq];
s11=(fpicoll^2-(fpicoll-1)^2)./(2*fpicoll*(2*fpicoll-1)*tan([max(alfa1/2) max(alfa2/2)
ax(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2) max(alfa8/2)
ax(alfa9/2) max(alfa10/2) max(alfa11/2) max(alfa12/2) max(alfa13/2) max(alfa14/2)]))
l=[find(alfa1==max(alfa1)) find(alfa2==max(alfa2)) find(alfa3==max(alfa3))
ind(alfa4==max(alfa4)) find(alfa5==max(alfa5)) find(alfa6==max(alfa6))
ind(alfa7==max(alfa7)) find(alfa8==max(alfa8)) find(alfa9==max(alfa9))
ind(alfa10==max(alfa10)) find(alfa11==max(alfa11)) find(alfa12==max(alfa12))
ind(alfa13==max(alfa13)) find(alfa14==max(alfa14))];

cro2=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[]; alfa7=[];
lfa8=[]; alfa9=[]; alfa10=[]; alfa11=[]; alfa12=[]; alfa13=[]; alfa14=[];

```

```

nyquist(rxs1_21,ixs1_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs1_21(1+2*(N-A-1):16-2*(1+A)),ixs1_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs1_21(1+2*(N-A):16-2*A),ixs1_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs1_21(1+2*(N-A-1):16-2*(1+A)),ixs1_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs1_21(n+1))^2+(y0-ixs1_21(n+1))^2);
bb=sqrt((x0-rxs1_21(n))^2+(y0-ixs1_21(n))^2);
cc=sqrt((rxs1_21(n+1)-rxs1_21(n))^2+(ixs1_21(n+1)-ixs1_21(n))^2);
alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro2=[erro2 rq];
nyquist(rxs2_21,ixs2_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs2_21(1+2*(N-A-1):16-2*(1+A)),ixs2_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs2_21(1+2*(N-A):16-2*A),ixs2_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs2_21(1+2*(N-A-1):16-2*(1+A)),ixs2_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs2_21(n+1))^2+(y0-ixs2_21(n+1))^2);
bb=sqrt((x0-rxs2_21(n))^2+(y0-ixs2_21(n))^2);
cc=sqrt((rxs2_21(n+1)-rxs2_21(n))^2+(ixs2_21(n+1)-ixs2_21(n))^2);
alfa2(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro2=[erro2 rq];
nyquist(rxs3_21,ixs3_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs3_21(1+2*(N-A-1):16-2*(1+A)),ixs3_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs3_21(1+2*(N-A):16-2*A),ixs3_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs3_21(1+2*(N-A-1):16-2*(1+A)),ixs3_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs3_21(n+1))^2+(y0-ixs3_21(n+1))^2);
bb=sqrt((x0-rxs3_21(n))^2+(y0-ixs3_21(n))^2);
cc=sqrt((rxs3_21(n+1)-rxs3_21(n))^2+(ixs3_21(n+1)-ixs3_21(n))^2);

```

```

    alfa3(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro2=[erro2 rq];
nyquist(rxs4_21,ixs4_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs4_21(1+2*(N-A-1):16-2*(1+A)),ixs4_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs4_21(1+2*(N-A):16-2*A),ixs4_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs4_21(1+2*(N-A-1):16-2*(1+A)),ixs4_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs4_21(n+1))^2+(y0-ixs4_21(n+1))^2);
bb=sqrt((x0-rxs4_21(n))^2+(y0-ixs4_21(n))^2);
cc=sqrt((rxs4_21(n+1)-rxs4_21(n))^2+(ixs4_21(n+1)-ixs4_21(n))^2);
alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro2=[erro2 rq];
nyquist(rxs5_21,ixs5_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs5_21(1+2*(N-A-1):16-2*(1+A)),ixs5_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs5_21(1+2*(N-A):16-2*A),ixs5_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs5_21(1+2*(N-A-1):16-2*(1+A)),ixs5_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs5_21(n+1))^2+(y0-ixs5_21(n+1))^2);
bb=sqrt((x0-rxs5_21(n))^2+(y0-ixs5_21(n))^2);
cc=sqrt((rxs5_21(n+1)-rxs5_21(n))^2+(ixs5_21(n+1)-ixs5_21(n))^2);
alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro2=[erro2 rq];
nyquist(rxs6_21,ixs6_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs6_21(1+2*(N-A-1):16-2*(1+A)),ixs6_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs6_21(1+2*(N-A):16-2*A),ixs6_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs6_21(1+2*(N-A-1):16-2*(1+A)),ixs6_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A

```

```

aa=sqrt((x0-rxs6_21(n+1))^2+(y0-ixs6_21(n+1))^2);
bb=sqrt((x0-rxs6_21(n))^2+(y0-ixs6_21(n))^2);
cc=sqrt((rxs6_21(n+1)-rxs6_21(n))^2+(ixs6_21(n+1)-ixs6_21(n))^2);
alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[erro2 rq];
yquist(rxs7_21,ixs7_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs7_21(1+2*(N-A-1):16-2*(1+A)),ixs7_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs7_21(1+2*(N-A):16-2*A),ixs7_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs7_21(1+2*(N-A-1):16-2*(1+A)),ixs7_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs7_21(n+1))^2+(y0-ixs7_21(n+1))^2);
bb=sqrt((x0-rxs7_21(n))^2+(y0-ixs7_21(n))^2);
cc=sqrt((rxs7_21(n+1)-rxs7_21(n))^2+(ixs7_21(n+1)-ixs7_21(n))^2);
alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[erro2 rq];
yquist(rxs8_21,ixs8_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs8_21(1+2*(N-A-1):16-2*(1+A)),ixs8_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs8_21(1+2*(N-A):16-2*A),ixs8_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs8_21(1+2*(N-A-1):16-2*(1+A)),ixs8_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs8_21(n+1))^2+(y0-ixs8_21(n+1))^2);
bb=sqrt((x0-rxs8_21(n))^2+(y0-ixs8_21(n))^2);
cc=sqrt((rxs8_21(n+1)-rxs8_21(n))^2+(ixs8_21(n+1)-ixs8_21(n))^2);
alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro2=[erro2 rq];
yquist(rxs9_21,ixs9_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs9_21(1+2*(N-A-1):16-2*(1+A)),ixs9_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs9_21(1+2*(N-A):16-2*A),ixs9_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs9_21(1+2*(N-A-1):16-2*(1+A)),ixs9_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end

```

```

end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs9_21(n+1))^2+(y0-ixs9_21(n+1))^2);
bb=sqrt((x0-rxs9_21(n))^2+(y0-ixs9_21(n))^2);
cc=sqrt((rxs9_21(n+1)-rxs9_21(n))^2+(ixs9_21(n+1)-ixs9_21(n))^2);
alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
nyquist(rxs10_21,ixs10_21)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs10_21(1+2*(N-A-1):16-2*(1+A)),ixs10_21(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs10_21(1+2*(N-A):16-2*A),ixs10_21(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rxs10_21(1+2*(N-A-1):16-2*(1+A)),ixs10_21(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs10_21(n+1))^2+(y0-ixs10_21(n+1))^2);
bb=sqrt((x0-rxs10_21(n))^2+(y0-ixs10_21(n))^2);
cc=sqrt((rxs10_21(n+1)-rxs10_21(n))^2+(ixs10_21(n+1)-ixs10_21(n))^2);
alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
nyquist(rxs11_21,ixs11_21)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs11_21(1+2*(N-A-1):16-2*(1+A)),ixs11_21(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs11_21(1+2*(N-A):16-2*A),ixs11_21(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rxs11_21(1+2*(N-A-1):16-2*(1+A)),ixs11_21(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs11_21(n+1))^2+(y0-ixs11_21(n+1))^2);
bb=sqrt((x0-rxs11_21(n))^2+(y0-ixs11_21(n))^2);
cc=sqrt((rxs11_21(n+1)-rxs11_21(n))^2+(ixs11_21(n+1)-ixs11_21(n))^2);
alfa11(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
nyquist(rxs12_21,ixs12_21)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs12_21(1+2*(N-A-1):16-2*(1+A)),ixs12_21(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs12_21(1+2*(N-A):16-2*A),ixs12_21(1+2*(N-A):16-2*A));
b=rq;

```

```

    if(a>b)
        nyquist(rxs12_21(1+2*(N-A-1):16-2*(1+A)),ixs12_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs12_21(n+1))^2+(y0-ixs12_21(n+1))^2);
    bb=sqrt((x0-rxs12_21(n))^2+(y0-ixs12_21(n))^2);
    cc=sqrt((rxs12_21(n+1)-rxs12_21(n))^2+(ixs12_21(n+1)-ixs12_21(n))^2);
    alfa12(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
yquist(rxs13_21,ixs13_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs13_21(1+2*(N-A-1):16-2*(1+A)),ixs13_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs13_21(1+2*(N-A):16-2*A),ixs13_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs13_21(1+2*(N-A-1):16-2*(1+A)),ixs13_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs13_21(n+1))^2+(y0-ixs13_21(n+1))^2);
    bb=sqrt((x0-rxs13_21(n))^2+(y0-ixs13_21(n))^2);
    cc=sqrt((rxs13_21(n+1)-rxs13_21(n))^2+(ixs13_21(n+1)-ixs13_21(n))^2);
    alfa13(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
yquist(rxs14_21,ixs14_21)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs14_21(1+2*(N-A-1):16-2*(1+A)),ixs14_21(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs14_21(1+2*(N-A):16-2*A),ixs14_21(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs14_21(1+2*(N-A-1):16-2*(1+A)),ixs14_21(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs14_21(n+1))^2+(y0-ixs14_21(n+1))^2);
    bb=sqrt((x0-rxs14_21(n))^2+(y0-ixs14_21(n))^2);
    cc=sqrt((rxs14_21(n+1)-rxs14_21(n))^2+(ixs14_21(n+1)-ixs14_21(n))^2);
    alfa14(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro2=[erro2 rq];
si2=(fpico21^2-(fpico21-1)^2)./(2*fpico21*(2*fpico21-1)*tan([max(alfa1/2) max(alfa2/2)
max(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2) max(alfa8/2)
max(alfa9/2) max(alfa10/2) max(alfa11/2) max(alfa12/2) max(alfa13/2) max(alfa14/2)]))
=[find(alfa1==max(alfa1)) find(alfa2==max(alfa2)) find(alfa3==max(alfa3))
find(alfa4==max(alfa4)) find(alfa5==max(alfa5)) find(alfa6==max(alfa6))

```

```

ind(alfa7==max(alfa7)) find(alfa8==max(alfa8)) find(alfa9==max(alfa9))
ind(alfa10==max(alfa10)) find(alfa11==max(alfa11)) find(alfa12==max(alfa12))
ind(alfa13==max(alfa13)) find(alfa14==max(alfa14));

erro3=[]; alfa1=[]; alfa2=[]; alfa3=[]; alfa4=[]; alfa5=[]; alfa6=[]; alfa7=[];
alfa8=[]; alfa9=[]; alfa10=[]; alfa11=[]; alfa12=[]; alfa13=[]; alfa14=[];
nyquist(rxs1_31,ixs1_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs1_31(1+2*(N-A-1):16-2*(1+A)),ixs1_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs1_31(1+2*(N-A):16-2*A),ixs1_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs1_31(1+2*(N-A-1):16-2*(1+A)),ixs1_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs1_31(n+1))^2+(y0-ixs1_31(n+1))^2);
bb=sqrt((x0-rxs1_31(n))^2+(y0-ixs1_31(n))^2);
cc=sqrt((rxs1_31(n+1)-rxs1_31(n))^2+(ixs1_31(n+1)-ixs1_31(n))^2);
alfa1(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[erro3 rq];
nyquist(rxs2_31,ixs2_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs2_31(1+2*(N-A-1):16-2*(1+A)),ixs2_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs2_31(1+2*(N-A):16-2*A),ixs2_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs2_31(1+2*(N-A-1):16-2*(1+A)),ixs2_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs2_31(n+1))^2+(y0-ixs2_31(n+1))^2);
bb=sqrt((x0-rxs2_31(n))^2+(y0-ixs2_31(n))^2);
cc=sqrt((rxs2_31(n+1)-rxs2_31(n))^2+(ixs2_31(n+1)-ixs2_31(n))^2);
alfa2(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[erro3 rq];
nyquist(rxs3_31,ixs3_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs3_31(1+2*(N-A-1):16-2*(1+A)),ixs3_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs3_31(1+2*(N-A):16-2*A),ixs3_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs3_31(1+2*(N-A-1):16-2*(1+A)),ixs3_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end

```

```

end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs3_31(n+1))^2+(y0-ixs3_31(n+1))^2);
bb=sqrt((x0-rxs3_31(n))^2+(y0-ixs3_31(n))^2);
cc=sqrt((rxs3_31(n+1)-rxs3_31(n))^2+(ixs3_31(n+1)-ixs3_31(n))^2);
alfa3(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[erro3 rq];
yquist(rxs4_31,ixs4_31)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs4_31(1+2*(N-A-1):16-2*(1+A)),ixs4_31(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs4_31(1+2*(N-A):16-2*A),ixs4_31(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rxs4_31(1+2*(N-A-1):16-2*(1+A)),ixs4_31(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs4_31(n+1))^2+(y0-ixs4_31(n+1))^2);
bb=sqrt((x0-rxs4_31(n))^2+(y0-ixs4_31(n))^2);
cc=sqrt((rxs4_31(n+1)-rxs4_31(n))^2+(ixs4_31(n+1)-ixs4_31(n))^2);
alfa4(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[erro3 rq];
yquist(rxs5_31,ixs5_31)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs5_31(1+2*(N-A-1):16-2*(1+A)),ixs5_31(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs5_31(1+2*(N-A):16-2*A),ixs5_31(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
nyquist(rxs5_31(1+2*(N-A-1):16-2*(1+A)),ixs5_31(1+2*(N-A-1):16-2*(1+A)));
A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs5_31(n+1))^2+(y0-ixs5_31(n+1))^2);
bb=sqrt((x0-rxs5_31(n))^2+(y0-ixs5_31(n))^2);
cc=sqrt((rxs5_31(n+1)-rxs5_31(n))^2+(ixs5_31(n+1)-ixs5_31(n))^2);
alfa5(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
rro3=[erro3 rq];
yquist(rxs6_31,ixs6_31)
=0; A=0;
while(rq<.7&N<3)
N=N+1;
nyquist(rxs6_31(1+2*(N-A-1):16-2*(1+A)),ixs6_31(1+2*(N-A-1):16-2*(1+A)));
a=rq;
nyquist(rxs6_31(1+2*(N-A):16-2*A),ixs6_31(1+2*(N-A):16-2*A));
b=rq;

```

```

    if(a>b)
        nyquist(rxs6_31(1+2*(N-A-1):16-2*(1+A)),ixs6_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs6_31(n+1))^2+(y0-ixs6_31(n+1))^2);
    bb=sqrt((x0-rxs6_31(n))^2+(y0-ixs6_31(n))^2);
    cc=sqrt((rxs6_31(n+1)-rxs6_31(n))^2+(ixs6_31(n+1)-ixs6_31(n))^2);
    alfa6(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs7_31,ixs7_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs7_31(1+2*(N-A-1):16-2*(1+A)),ixs7_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs7_31(1+2*(N-A):16-2*A),ixs7_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs7_31(1+2*(N-A-1):16-2*(1+A)),ixs7_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs7_31(n+1))^2+(y0-ixs7_31(n+1))^2);
    bb=sqrt((x0-rxs7_31(n))^2+(y0-ixs7_31(n))^2);
    cc=sqrt((rxs7_31(n+1)-rxs7_31(n))^2+(ixs7_31(n+1)-ixs7_31(n))^2);
    alfa7(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs8_31,ixs8_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs8_31(1+2*(N-A-1):16-2*(1+A)),ixs8_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs8_31(1+2*(N-A):16-2*A),ixs8_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs8_31(1+2*(N-A-1):16-2*(1+A)),ixs8_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
nd
or n=1+2*(N-A):15-2*A
    aa=sqrt((x0-rxs8_31(n+1))^2+(y0-ixs8_31(n+1))^2);
    bb=sqrt((x0-rxs8_31(n))^2+(y0-ixs8_31(n))^2);
    cc=sqrt((rxs8_31(n+1)-rxs8_31(n))^2+(ixs8_31(n+1)-ixs8_31(n))^2);
    alfa8(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs9_31,ixs9_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs9_31(1+2*(N-A-1):16-2*(1+A)),ixs9_31(1+2*(N-A-1):16-2*(1+A)));

```

```

a=rq;
nyquist(rxs9_31(1+2*(N-A):16-2*A),ixs9_31(1+2*(N-A):16-2*A));
b=rq;
if(a>b)
    nyquist(rxs9_31(1+2*(N-A-1):16-2*(1+A)),ixs9_31(1+2*(N-A-1):16-2*(1+A)));
    A=A+1;
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs9_31(n+1))^2+(y0-ixs9_31(n+1))^2);
bb=sqrt((x0-rxs9_31(n))^2+(y0-ixs9_31(n))^2);
cc=sqrt((rxs9_31(n+1)-rxs9_31(n))^2+(ixs9_31(n+1)-ixs9_31(n))^2);
alfa9(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs10_31,ixs10_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs10_31(1+2*(N-A-1):16-2*(1+A)),ixs10_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs10_31(1+2*(N-A):16-2*A),ixs10_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs10_31(1+2*(N-A-1):16-2*(1+A)),ixs10_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs10_31(n+1))^2+(y0-ixs10_31(n+1))^2);
bb=sqrt((x0-rxs10_31(n))^2+(y0-ixs10_31(n))^2);
cc=sqrt((rxs10_31(n+1)-rxs10_31(n))^2+(ixs10_31(n+1)-ixs10_31(n))^2);
alfa10(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs11_31,ixs11_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs11_31(1+2*(N-A-1):16-2*(1+A)),ixs11_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs11_31(1+2*(N-A):16-2*A),ixs11_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs11_31(1+2*(N-A-1):16-2*(1+A)),ixs11_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
nd
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs11_31(n+1))^2+(y0-ixs11_31(n+1))^2);
bb=sqrt((x0-rxs11_31(n))^2+(y0-ixs11_31(n))^2);
cc=sqrt((rxs11_31(n+1)-rxs11_31(n))^2+(ixs11_31(n+1)-ixs11_31(n))^2);
alfa11(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
nd
erro3=[erro3 rq];
nyquist(rxs12_31,ixs12_31)
=0; A=0;

```

```

while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs12_31(1+2*(N-A-1):16-2*(1+A)),ixs12_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs12_31(1+2*(N-A):16-2*A),ixs12_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs12_31(1+2*(N-A-1):16-2*(1+A)),ixs12_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs12_31(n+1))^2+(y0-ixs12_31(n+1))^2);
bb=sqrt((x0-rxs12_31(n))^2+(y0-ixs12_31(n))^2);
cc=sqrt((rxs12_31(n+1)-rxs12_31(n))^2+(ixs12_31(n+1)-ixs12_31(n))^2);
alfa12(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[erro3 rq];
nyquist(rxs13_31,ixs13_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs13_31(1+2*(N-A-1):16-2*(1+A)),ixs13_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs13_31(1+2*(N-A):16-2*A),ixs13_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs13_31(1+2*(N-A-1):16-2*(1+A)),ixs13_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs13_31(n+1))^2+(y0-ixs13_31(n+1))^2);
bb=sqrt((x0-rxs13_31(n))^2+(y0-ixs13_31(n))^2);
cc=sqrt((rxs13_31(n+1)-rxs13_31(n))^2+(ixs13_31(n+1)-ixs13_31(n))^2);
alfa13(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end
erro3=[erro3 rq];
nyquist(rxs14_31,ixs14_31)
=0; A=0;
while(rq<.7&N<3)
    N=N+1;
    nyquist(rxs14_31(1+2*(N-A-1):16-2*(1+A)),ixs14_31(1+2*(N-A-1):16-2*(1+A)));
    a=rq;
    nyquist(rxs14_31(1+2*(N-A):16-2*A),ixs14_31(1+2*(N-A):16-2*A));
    b=rq;
    if(a>b)
        nyquist(rxs14_31(1+2*(N-A-1):16-2*(1+A)),ixs14_31(1+2*(N-A-1):16-2*(1+A)));
        A=A+1;
    end
end
or n=1+2*(N-A):15-2*A
aa=sqrt((x0-rxs14_31(n+1))^2+(y0-ixs14_31(n+1))^2);
bb=sqrt((x0-rxs14_31(n))^2+(y0-ixs14_31(n))^2);
cc=sqrt((rxs14_31(n+1)-rxs14_31(n))^2+(ixs14_31(n+1)-ixs14_31(n))^2);
alfa14(n)=[acos((aa^2+bb^2-cc^2)/(2*aa*bb))];
end

```

```

erro3=[erro3 rq];
si3=(fpico31^2-(fpico31-1)^2)./(2*fpico31*(2*fpico31-1)*tan([max(alfa1/2) max(alfa2/2)
max(alfa3/2) max(alfa4/2) max(alfa5/2) max(alfa6/2) max(alfa7/2) max(alfa8/2)
max(alfa9/2) max(alfa10/2) max(alfa11/2) max(alfa12/2) max(alfa13/2) max(alfa14/2)]))
3=[find(alfa1==max(alfa1)) find(alfa2==max(alfa2)) find(alfa3==max(alfa3))
find(alfa4==max(alfa4)) find(alfa5==max(alfa5)) find(alfa6==max(alfa6))
find(alfa7==max(alfa7)) find(alfa8==max(alfa8)) find(alfa9==max(alfa9))
find(alfa10==max(alfa10)) find(alfa11==max(alfa11)) find(alfa12==max(alfa12))
find(alfa13==max(alfa13)) find(alfa14==max(alfa14))];

```

```

cro1=str2num(num2str(100*Erro1,2));
cro2=str2num(num2str(100*Erro2,2));
cro3=str2num(num2str(100*Erro3,2));
cro1=str2num(num2str(100*erro1,2));
cro2=str2num(num2str(100*erro2,2));
cro3=str2num(num2str(100*erro3,2));

```

Exibe a precisão do circulo de Nyquist ajustado em cada ponto e a posicao da fn:

```

lsp(' ')
lsp(' Precisação 1 (%) :')
ats(Erro1,4)
lsp(' N1:')
ats(N1,4)
lsp(' ')
lsp(' Precisação 2 (%) :')
ats(Erro2,4)
lsp(' N2:')
ats(N2,4)
lsp(' ')
lsp(' Precisação 3 (%) :')
ats(Erro3,4)
lsp(' N3:')
ats(N3,4)
lsp(' ')
lsp(' precisao 1 (%) :')
ats(erro1,4)
lsp(' n1:')
ats(n1,4)
lsp(' ')
lsp(' precisao 2 (%) :')
ats(erro2,4)
lsp(' n2:')
ats(n2,4)
lsp(' ')
lsp(' precisao 3 (%) :')
ats(erro3,4)
lsp(' n3:')
ats(n3,4)
lsp(' ')

```

Modulo 9: Determinação das amplitudes dos modos de vibrar (ui) da placa:

```

H11=[(max(as1(fpico11*na/fa-8+n1(1)),as1(fpico11*na/fa-7+n1(1))))
max(as1(fpico21*na/fa-8+n2(1)),as1(fpico21*na/fa-7+n2(1))))
max(as1(fpico31*na/fa-8+n3(1)),as1(fpico31*na/fa-7+n3(1))))];

```

```

H21=[ (max(as2(fpicol1*na/fa-8+n1(2)),as2(fpicol1*na/fa-7+n1(2))))
max(as2(fpico21*na/fa-8+n2(2)),as2(fpico21*na/fa-7+n2(2))))
max(as2(fpico31*na/fa-8+n3(2)),as2(fpico31*na/fa-7+n3(2))))];
H31=[ (max(as3(fpicol1*na/fa-8+n1(3)),as3(fpicol1*na/fa-7+n1(3))))
max(as3(fpico21*na/fa-8+n2(3)),as3(fpico21*na/fa-7+n2(3))))
max(as3(fpico31*na/fa-8+n3(3)),as3(fpico31*na/fa-7+n3(3))))];
H41=[ (max(as4(fpicol1*na/fa-8+n1(4)),as4(fpicol1*na/fa-7+n1(4))))
max(as4(fpico21*na/fa-8+n2(4)),as4(fpico21*na/fa-7+n2(4))))
max(as4(fpico31*na/fa-8+n3(4)),as4(fpico31*na/fa-7+n3(4))))];
H51=[ (max(as5(fpicol1*na/fa-8+n1(5)),as5(fpicol1*na/fa-7+n1(5))))
max(as5(fpico21*na/fa-8+n2(5)),as5(fpico21*na/fa-7+n2(5))))
max(as5(fpico31*na/fa-8+n3(5)),as5(fpico31*na/fa-7+n3(5))))];
H61=[ (max(as5(fpicol1*na/fa-8+n1(5)),as5(fpicol1*na/fa-7+n1(5))))
max(as6(fpico21*na/fa-8+n2(6)),as6(fpico21*na/fa-7+n2(6))))
max(as6(fpico31*na/fa-8+n3(6)),as6(fpico31*na/fa-7+n3(6))))];
H71=[ (max(as7(fpicol1*na/fa-8+n1(7)),as7(fpicol1*na/fa-7+n1(7))))
max(as7(fpico21*na/fa-8+n2(7)),as7(fpico21*na/fa-7+n2(7))))
max(as7(fpico31*na/fa-8+n3(7)),as7(fpico31*na/fa-7+n3(7))))];
H81=[ (max(as8(fpicol1*na/fa-8+n1(8)),as8(fpicol1*na/fa-7+n1(8))))
max(as8(fpico21*na/fa-8+n2(8)),as8(fpico21*na/fa-7+n2(8))))
max(as8(fpico31*na/fa-8+n3(8)),as8(fpico31*na/fa-7+n3(8))))];
H91=[ (max(as9(fpicol1*na/fa-8+n1(9)),as9(fpicol1*na/fa-7+n1(9))))
max(as9(fpico21*na/fa-8+n2(9)),as9(fpico21*na/fa-7+n2(9))))
max(as9(fpico31*na/fa-8+n3(9)),as9(fpico31*na/fa-7+n3(9))))];
H101=[ (max(as10(fpicol1*na/fa-8+n1(10)),as10(fpicol1*na/fa-7+n1(10))))
max(as10(fpico21*na/fa-8+n2(10)),as10(fpico21*na/fa-7+n2(10))))
max(as10(fpico31*na/fa-8+n3(10)),as10(fpico31*na/fa-7+n3(10))))];
H111=[ (max(as11(fpicol1*na/fa-8+n1(11)),as11(fpicol1*na/fa-7+n1(11))))
max(as11(fpico21*na/fa-8+n2(11)),as11(fpico21*na/fa-7+n2(11))))
max(as11(fpico31*na/fa-8+n3(11)),as11(fpico31*na/fa-7+n3(11))))];
H121=[ (max(as12(fpicol1*na/fa-8+n1(12)),as12(fpicol1*na/fa-7+n1(12))))
max(as12(fpico21*na/fa-8+n2(12)),as12(fpico21*na/fa-7+n2(12))))
max(as12(fpico31*na/fa-8+n3(12)),as12(fpico31*na/fa-7+n3(12))))];
H131=[ (max(as13(fpicol1*na/fa-8+n1(13)),as13(fpicol1*na/fa-7+n1(13))))
max(as13(fpico21*na/fa-8+n2(13)),as13(fpico21*na/fa-7+n2(13))))
max(as13(fpico31*na/fa-8+n3(13)),as13(fpico31*na/fa-7+n3(13))))];
H141=[ (max(as14(fpicol1*na/fa-8+n1(14)),as14(fpicol1*na/fa-7+n1(14))))
max(as14(fpico21*na/fa-8+n2(14)),as14(fpico21*na/fa-7+n2(14))))
max(as14(fpico31*na/fa-8+n3(14)),as14(fpico31*na/fa-7+n3(14))))];

I01=[ (max(bs0(fpicol1*na/fa-8+N1(1)),bs0(fpicol1*na/fa-7+N1(1))))
max(bs0(fpico21*na/fa-8+N2(1)),bs0(fpico21*na/fa-7+N2(1))))
max(bs0(fpico31*na/fa-8+N3(1)),bs0(fpico31*na/fa-7+N3(1))))];
I11=[ (max(bs1(fpicol1*na/fa-8+N1(2)),bs1(fpicol1*na/fa-7+N1(2))))
max(bs1(fpico21*na/fa-8+N2(2)),bs1(fpico21*na/fa-7+N2(2))))
max(bs1(fpico31*na/fa-8+N3(2)),bs1(fpico31*na/fa-7+N3(2))))];
I21=[ (max(bs2(fpicol1*na/fa-8+N1(3)),bs2(fpicol1*na/fa-7+N1(3))))
max(bs2(fpico21*na/fa-8+N2(3)),bs2(fpico21*na/fa-7+N2(3))))
max(bs2(fpico31*na/fa-8+N3(3)),bs2(fpico31*na/fa-7+N3(3))))];
I31=[ (max(bs3(fpicol1*na/fa-8+N1(4)),bs3(fpicol1*na/fa-7+N1(4))))
max(bs3(fpico21*na/fa-8+N2(4)),bs3(fpico21*na/fa-7+N2(4))))
max(bs3(fpico31*na/fa-8+N3(4)),bs3(fpico31*na/fa-7+N3(4))))];
I41=[ (max(bs4(fpicol1*na/fa-8+N1(5)),bs4(fpicol1*na/fa-7+N1(5))))
max(bs4(fpico21*na/fa-8+N2(5)),bs4(fpico21*na/fa-7+N2(5))))
max(bs4(fpico31*na/fa-8+N3(5)),bs4(fpico31*na/fa-7+N3(5))))];

```

```

I51=[(max(bs5(fpico11*na/fa-8+N1(6)),bs5(fpico11*na/fa-7+N1(6))))
nax(bs5(fpico21*na/fa-8+N2(6)),bs5(fpico21*na/fa-7+N2(6))))
nax(bs5(fpico31*na/fa-8+N3(6)),bs5(fpico31*na/fa-7+N3(6))))];
I61=[(max(bs6(fpico11*na/fa-8+N1(7)),bs6(fpico11*na/fa-7+N1(7))))
nax(bs6(fpico21*na/fa-8+N2(7)),bs6(fpico21*na/fa-7+N2(7))))
nax(bs6(fpico31*na/fa-8+N3(7)),bs6(fpico31*na/fa-7+N3(7))))];
I71=[(max(bs7(fpico11*na/fa-8+N1(8)),bs7(fpico11*na/fa-7+N1(8))))
nax(bs7(fpico21*na/fa-8+N2(8)),bs7(fpico21*na/fa-7+N2(8))))
nax(bs7(fpico31*na/fa-8+N3(8)),bs7(fpico31*na/fa-7+N3(8))))];
I81=[(max(bs8(fpico11*na/fa-8+N1(9)),bs8(fpico11*na/fa-7+N1(9))))
nax(bs8(fpico21*na/fa-8+N2(9)),bs8(fpico21*na/fa-7+N2(9))))
nax(bs8(fpico31*na/fa-8+N3(9)),bs8(fpico31*na/fa-7+N3(9))))];
I91=[(max(bs9(fpico11*na/fa-8+N1(10)),bs9(fpico11*na/fa-7+N1(10))))
nax(bs9(fpico21*na/fa-8+N2(10)),bs9(fpico21*na/fa-7+N2(10))))
nax(bs9(fpico31*na/fa-8+N3(10)),bs9(fpico31*na/fa-7+N3(10))))];
I101=[(max(bs10(fpico11*na/fa-8+N1(11)),bs10(fpico11*na/fa-7+N1(11))))
nax(bs10(fpico21*na/fa-8+N2(11)),bs10(fpico21*na/fa-7+N2(11))))
nax(bs10(fpico31*na/fa-8+N3(11)),bs10(fpico31*na/fa-7+N3(11))))];

```

```

wn=2*pi*[fpico11 fpico21 fpico31];
Wn=wn;

```

```

u1u1t11=2*(sum(ksi1))*wn(1)^2*H11(1);
u1u1t21=2*(sum(ksi1))*wn(1)^2*H21(1);
u1u1t31=2*(sum(ksi1))*wn(1)^2*H31(1);
u1u1t41=2*(sum(ksi1))*wn(1)^2*H41(1);
u1u1t51=2*(sum(ksi1))*wn(1)^2*H51(1);
u1u1t61=2*(sum(ksi1))*wn(1)^2*H61(1);
u1u1t71=2*(sum(ksi1))*wn(1)^2*H71(1);
u1u1t81=2*(sum(ksi1))*wn(1)^2*H81(1);
u1u1t91=2*(sum(ksi1))*wn(1)^2*H91(1);
u1u1t101=2*(sum(ksi1))*wn(1)^2*H101(1);
u1u1t111=2*(sum(ksi1))*wn(1)^2*H111(1);
u1u1t121=2*(sum(ksi1))*wn(1)^2*H121(1);
u1u1t131=2*(sum(ksi1))*wn(1)^2*H131(1);
u1u1t141=2*(sum(ksi1))*wn(1)^2*H141(1);

```

```

u2u2t11=2*(sum(ksi2))*wn(2)^2*H11(2);
u2u2t21=2*(sum(ksi2))*wn(2)^2*H21(2);
u2u2t31=2*(sum(ksi2))*wn(2)^2*H31(2);
u2u2t41=2*(sum(ksi2))*wn(2)^2*H41(2);
u2u2t51=2*(sum(ksi2))*wn(2)^2*H51(2);
u2u2t61=2*(sum(ksi2))*wn(2)^2*H61(2);
u2u2t71=2*(sum(ksi2))*wn(2)^2*H71(2);
u2u2t81=2*(sum(ksi2))*wn(2)^2*H81(2);
u2u2t91=2*(sum(ksi2))*wn(2)^2*H91(2);
u2u2t101=2*(sum(ksi2))*wn(2)^2*H101(2);
u2u2t111=2*(sum(ksi2))*wn(2)^2*H111(2);
u2u2t121=2*(sum(ksi2))*wn(2)^2*H121(2);
u2u2t131=2*(sum(ksi2))*wn(2)^2*H131(2);
u2u2t141=2*(sum(ksi2))*wn(2)^2*H141(2);

```

```

u3u3t11=2*(sum(ksi3))*wn(3)^2*H11(3);
u3u3t21=2*(sum(ksi3))*wn(3)^2*H21(3);
u3u3t31=2*(sum(ksi3))*wn(3)^2*H31(3);
u3u3t41=2*(sum(ksi3))*wn(3)^2*H41(3);
u3u3t51=2*(sum(ksi3))*wn(3)^2*H51(3);

```

```

u3u3t61=2*(sum(ksi3))*wn(3)^2*H61(3);
u3u3t71=2*(sum(ksi3))*wn(3)^2*H71(3);
u3u3t81=2*(sum(ksi3))*wn(3)^2*H81(3);
u3u3t91=2*(sum(ksi3))*wn(3)^2*H91(3);
u3u3t101=2*(sum(ksi3))*wn(3)^2*H101(3);
u3u3t111=2*(sum(ksi3))*wn(3)^2*H111(3);
u3u3t121=2*(sum(ksi3))*wn(3)^2*H121(3);
u3u3t131=2*(sum(ksi3))*wn(3)^2*H131(3);
u3u3t141=2*(sum(ksi3))*wn(3)^2*H141(3);

```

```

ululT01=2*(sum(Ksi1))*Wn(1)^2*I01(1);
ululT11=2*(sum(Ksi1))*Wn(1)^2*I11(1);
ululT21=2*(sum(Ksi1))*Wn(1)^2*I21(1);
ululT31=2*(sum(Ksi1))*Wn(1)^2*I31(1);
ululT41=2*(sum(Ksi1))*Wn(1)^2*I41(1);
ululT51=2*(sum(Ksi1))*Wn(1)^2*I51(1);
ululT61=2*(sum(Ksi1))*Wn(1)^2*I61(1);
ululT71=2*(sum(Ksi1))*Wn(1)^2*I71(1);
ululT81=2*(sum(Ksi1))*Wn(1)^2*I81(1);
ululT91=2*(sum(Ksi1))*Wn(1)^2*I91(1);
ululT101=2*(sum(Ksi1))*Wn(1)^2*I101(1);

```

```

u2u2T01=2*(sum(Ksi2))*Wn(2)^2*I01(2);
u2u2T11=2*(sum(Ksi2))*Wn(2)^2*I11(2);
u2u2T21=2*(sum(Ksi2))*Wn(2)^2*I21(2);
u2u2T31=2*(sum(Ksi2))*Wn(2)^2*I31(2);
u2u2T41=2*(sum(Ksi2))*Wn(2)^2*I41(2);
u2u2T51=2*(sum(Ksi2))*Wn(2)^2*I51(2);
u2u2T61=2*(sum(Ksi2))*Wn(2)^2*I61(2);
u2u2T71=2*(sum(Ksi2))*Wn(2)^2*I71(2);
u2u2T81=2*(sum(Ksi2))*Wn(2)^2*I81(2);
u2u2T91=2*(sum(Ksi2))*Wn(2)^2*I91(2);
u2u2T101=2*(sum(Ksi2))*Wn(2)^2*I101(2);

```

```

u3u3T01=2*(sum(Ksi3))*Wn(3)^2*I01(3);
u3u3T11=2*(sum(Ksi3))*Wn(3)^2*I11(3);
u3u3T21=2*(sum(Ksi3))*Wn(3)^2*I21(3);
u3u3T31=2*(sum(Ksi3))*Wn(3)^2*I31(3);
u3u3T41=2*(sum(Ksi3))*Wn(3)^2*I41(3);
u3u3T51=2*(sum(Ksi3))*Wn(3)^2*I51(3);
u3u3T61=2*(sum(Ksi3))*Wn(3)^2*I61(3);
u3u3T71=2*(sum(Ksi3))*Wn(3)^2*I71(3);
u3u3T81=2*(sum(Ksi3))*Wn(3)^2*I81(3);
u3u3T91=2*(sum(Ksi3))*Wn(3)^2*I91(3);
u3u3T101=2*(sum(Ksi3))*Wn(3)^2*I101(3);

```

```

a1=sqrt(ulult11);
a2=ulult21/a1;
a3=ulult31/a1;
a4=ulult41/a1;
a5=ulult51/a1;
a6=ulult61/a1;
a7=ulult71/a1;
a8=ulult81/a1;
a9=ulult91/a1;
a10=ulult101/a1;
a11=ulult111/a1;

```

```

a12=u1u1t121/a1;
a13=u1u1t131/a1;
a14=u1u1t141/a1;
b1=sqrt(u2u2t11);
b2=u2u2t21/b1;
b3=u2u2t31/b1;
b4=u2u2t41/b1;
b5=u2u2t51/b1;
b6=u2u2t61/b1;
b7=u2u2t71/b1;
b8=u2u2t81/b1;
b9=u2u2t91/b1;
b10=u2u2t101/b1;
b11=u2u2t111/b1;
b12=u2u2t121/b1;
b13=u2u2t131/b1;
b14=u2u2t141/b1;
c1=sqrt(u3u3t11);
c2=u3u3t21/c1;
c3=u3u3t31/c1;
c4=u3u3t41/c1;
c5=u3u3t51/c1;
c6=u3u3t61/c1;
c7=u3u3t71/c1;
c8=u3u3t81/c1;
c9=u3u3t91/c1;
c10=u3u3t101/c1;
c11=u3u3t111/c1;
c12=u3u3t121/c1;
c13=u3u3t131/c1;
c14=u3u3t141/c1;

% Normalizacao das amplitudes dos modos de vibrar (ui):
u1=[0 a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 0]./max([a1 a2 a3 a4 a5 a6 a7
a9 a10 a11 a12 a13 a14]);
u2=[0 b1 b2 b3 b4 b5 b6 b7 -b8 -b9 -b10 -b11 -b12 -b13 -b14 0]./max([b1 b2 b3 b4 b5
b7 b8 b9 b10 b11 b12 b13 b14]);
u3=[0 c1 c2 c3 c4 -c5 -c6 -c7 -c8 -c9 -c10 c11 c12 c13 c14 0]./max([c1 c2 c3 c4 c5
c7 c8 c9 c10 c11 c12 c13 c14]);

A1=sqrt(u1u1T11);
A0=u1u1T01/A1;
A2=u1u1T21/A1;
A3=u1u1T31/A1;
A4=u1u1T41/A1;
A5=u1u1T51/A1;
A6=u1u1T61/A1;
A7=u1u1T71/A1;
A8=u1u1T81/A1;
A9=u1u1T91/A1;
A10=u1u1T101/A1;
B1=sqrt(u2u2T11);
B0=u2u2T01/B1;
B2=u2u2T21/B1;
B3=u2u2T31/B1;
B4=u2u2T41/B1;
B5=u2u2T51/B1;

```

```

B6=u2u2T61/B1;
B7=u2u2T71/B1;
B8=u2u2T81/B1;
B9=u2u2T91/B1;
B10=u2u2T101/B1;
C1=sqrt(u3u3T11);
C0=u3u3T01/C1;
C2=u3u3T21/C1;
C3=u3u3T31/C1;
C4=u3u3T41/C1;
C5=u3u3T51/C1;
C6=u3u3T61/C1;
C7=u3u3T71/C1;
C8=u3u3T81/C1;
C9=u3u3T91/C1;
C10=u3u3T101/C1;

% Normalizacao das amplitudes dos modos de vibrar (Ui):
if (resp==0)
    U1=[A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10]./max([A0 A1 A2 A3 A4 A5 A6 A7 A8 A9
10]);
    U2=[B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 B10]./max([B0 B1 B2 B3 B4 B5 B6 B7 B8 B9
10]);
    U3=[C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 C10]./max([C0 C1 C2 C3 C4 C5 C6 C7 C8 C9
10]);
    Y1i=Y01;
    Y2i=Y02;
    Y3i=Y03;
end
if (resp==1)
    U1=[A0 A1 A2 A3 A4 A5 -A6 -A7 -A8 -A9 -A10]./max([A0 A1 A2 A3 A4 A5 A6 A7 A8 A9
10]);
    U2=[B0 B1 B2 B3 B4 B5 -B6 -B7 -B8 -B9 -B10]./max([B0 B1 B2 B3 B4 B5 B6 B7 B8 B9
10]);
    U3=[C0 C1 C2 C3 C4 C5 -C6 -C7 -C8 -C9 -C10]./max([C0 C1 C2 C3 C4 C5 C6 C7 C8 C9
10]);
    Y1i=Y11;
    Y2i=Y21;
    Y3i=Y31;
end
if (resp==2)
    U1=[A0 A1 A2 A3 -A4 -A5 -A6 A7 A8 A9 A10]./max([A0 A1 A2 A3 A4 A5 A6 A7 A8 A9
.0]);
    U2=[B0 B1 B2 B3 -B4 -B5 -B6 B7 B8 B9 B10]./max([B0 B1 B2 B3 B4 B5 B6 B7 B8 B9
.0]);
    U3=[C0 C1 C2 C3 -C4 -C5 -C6 C7 C8 C9 C10]./max([C0 C1 C2 C3 C4 C5 C6 C7 C8 C9
.0]);
    Y1i=Y12;
    Y2i=Y22;
    Y3i=Y32;
end

% Ajuste dos pontos dos modos experimentais pelo metodo dos minimos quadrados:
m=[sum(x.^2) sum(x.^3) sum(x.^4) sum(x.^5) sum(x.^6); sum(x.^3) sum(x.^4) sum(x.^5)
sum(x.^6) sum(x.^7); sum(x.^4) sum(x.^5) sum(x.^6) sum(x.^7) sum(x.^8); sum(x.^5)
sum(x.^6) sum(x.^7) sum(x.^8) sum(x.^9); sum(x.^6) sum(x.^7) sum(x.^8) sum(x.^9)
sum(x.^10)];

```

```

r1=[sum(x.*u1) sum(x.^2.*u1) sum(x.^3.*u1) sum(x.^4.*u1) sum(x.^5.*u1)];
r2=[sum(x.*u2) sum(x.^2.*u2) sum(x.^3.*u2) sum(x.^4.*u2) sum(x.^5.*u2)];
r3=[sum(x.*u3) sum(x.^2.*u3) sum(x.^3.*u3) sum(x.^4.*u3) sum(x.^5.*u3)];
k1=r1/m;
k2=r2/m;
k3=r3/m;
x1=k1(1)*xc+k1(2)*xc.^2+k1(3)*xc.^3+k1(4)*xc.^4+k1(5)*xc.^5;
x2=k2(1)*xc+k2(2)*xc.^2+k2(3)*xc.^3+k2(4)*xc.^4+k2(5)*xc.^5;
x3=k3(1)*xc+k3(2)*xc.^2+k3(3)*xc.^3+k3(4)*xc.^4+k3(5)*xc.^5;

u1=u1./max(abs(x1));
x1=x1./max(abs(x1));
u2=u2./max(abs(x2));
x2=x2./max(abs(x2));
u3=u3./max(abs(x3));
x3=x3./max(abs(x3));

M=[length(y) sum(y) sum(y.^2) sum(y.^3); sum(y) sum(y.^2) sum(y.^3) sum(y.^4);
sum(y.^2) sum(y.^3) sum(y.^4) sum(y.^5); sum(y.^3) sum(y.^4) sum(y.^5) sum(y.^6)];
R1=[sum(U1) sum(y.*U1) sum(y.^2.*U1) sum(y.^3.*U1)];
R2=[sum(U2) sum(y.*U2) sum(y.^2.*U2) sum(y.^3.*U2)];
R3=[sum(U3) sum(y.*U3) sum(y.^2.*U3) sum(y.^3.*U3)];
K1=R1/M;
K2=R2/M;
K3=R3/M;
y1=K1(1)+K1(2)*yc+K1(3)*yc.^2+K1(4)*yc.^3;
y2=K2(1)+K2(2)*yc+K2(3)*yc.^2+K2(4)*yc.^3;
y3=K3(1)+K3(2)*yc+K3(3)*yc.^2+K3(4)*yc.^3;

U1=U1./max(abs(y1));
y1=y1./max(abs(y1));
U2=U2./max(abs(y2));
y2=y2./max(abs(y2));
U3=U3./max(abs(y3));
y3=y3./max(abs(y3));

% Modos de vibrar experimentais da placa:
for i=1:21
    for n=1:31
        modo11(i,n)=x1(-9+10*n)*y1(-9+10*i);
        modo21(i,n)=x2(-9+10*n)*y2(-9+10*i);
        modo31(i,n)=x3(-9+10*n)*y3(-9+10*i);
    end
end

% Modulo 10: Saida grafica dos modos de vibrar experimentais e analiticos da placa:

figure(20)
subplot(2,1,1)
plot(y,U1,'*',yc,y1,yc,Y1i./max(abs(Y1i)))
title(['1o Modo de vibrar - frequencia natural:
experimental=',num2str(fpicol1),'Hz, analitico=',num2str(f1),' Hz, diferenca:
num2str(100*(fpicol1-f1)/f1),'%']);
xlabel('Y - distancia a partir da borda livre');
ylabel('deflexao w');
legend('ptos. medidos','interpolado','analitico')
subplot(2,1,2)

```

```

plot(x,u1,'o',xc,x1,xc,X1)
xlabel('X - distancia a partir do apoio');
ylabel('deflexao w');
figure(21)
subplot(2,1,1)
plot(y,U2,'*',yc,y2,yc,Y2i./max(abs(Y2i)))
title(['2o Modo de vibrar - frequencia natural:
experimental=',num2str(fpico21),'Hz, analitico=',num2str(f2),' Hz, diferenca:
',num2str(100*(fpico21-f2)/f2),'%']);
xlabel('Y - distancia a partir da borda livre');
ylabel('deflexao w');
legend('ptos. medidos','interpolado','analitico')
subplot(2,1,2)
plot(x,u2,'o',xc,x2,xc,X2)
xlabel('X - distancia a partir do apoio');
ylabel('deflexao w');
figure(22)
subplot(2,1,1)
plot(y,U3,'*',yc,y3,yc,Y3i./max(abs(Y3i)))
title(['3o Modo de vibrar - frequencia natural:
experimental=',num2str(fpico31),'Hz, analitico=',num2str(f3),' Hz, diferenca:
',num2str(100*(fpico31-f3)/f3),'%']);
xlabel('Y - distancia a partir da borda livre');
ylabel('deflexao w');
legend('ptos. medidos','interpolado','analitico')
subplot(2,1,2)
plot(x,u3,'o',xc,x3,xc,X3)
xlabel('X - distancia a partir do apoio');
ylabel('deflexao w');
figure(23)
surf(mod011)
axis([0 30 0 20 -1 1]);
title(['1o Modo de vibrar - frequencia natural:
experimental=',num2str(fpico11),'Hz, analitico=',num2str(f1),' Hz, diferenca:
',num2str(100*(fpico11-f1)/f1),'%']);
xlabel('X - distancia a partir do apoio');
ylabel('Y - distancia a partir da borda livre');
zlabel('amplitude modulo 1');
figure(24)
surf(mod021)
axis([0 30 0 20 -1 1]);
title(['2o Modo de vibrar - frequencia natural:
experimental=',num2str(fpico21),'Hz, analitico=',num2str(f2),' Hz, diferenca:
',num2str(100*(fpico21-f2)/f2),'%']);
xlabel('X - distancia a partir do apoio');
ylabel('Y - distancia a partir da borda livre');
zlabel('amplitude modulo 1');
figure(25)
surf(mod031)
axis([0 30 0 20 -1 1]);
title(['3o Modo de vibrar - frequencia natural:
experimental=',num2str(fpico31),'Hz, analitico=',num2str(f3),' Hz, diferenca:
',num2str(100*(fpico31-f3)/f3),'%']);
xlabel('X - distancia a partir do apoio');
ylabel('Y - distancia a partir da borda livre');
zlabel('amplitude modulo 1');
disp(' Deseja escolher novamente as frequencias naturais?');

```

```
stop=input(' quelque numero = S; 0 = N :');  
disp(' ')  
nd
```

Programa para determinação dos modos e frequências naturais analíticas da placa (P_analitica.m):

```

Modos de Vibrar e Frequencias Naturais Analiticas da Placa Biapoiada:

clear all;
lc;
=0;
while (n~=1 & n~=2 & n~=3 & n~=4)
    n=input('Escolha o modo de vibrar em x (X=n*x*pi/a), n(1, 2, 3, ou 4)= ');
end
=.3;
=.2;
i=.3;
l=n*pi/a;
=[];
for i=0:.01:a
    x=[x; i];
end
=[0:.01:b];
l=sin(x*pi/a);
2=sin(2*x*pi/a);
3=sin(3*x*pi/a);
4=sin(4*x*pi/a);

caso 1: beta^4 < alpha^4; Y=A*sin(s*y)+B*cos(s*y)+C*sinh(r*y)+D*cosh(r*y)
=[];
for i=0:1200/(n+1)
    bt=al+i*al/100;
    s=sqrt(bt^2-al^2);
    r=sqrt(al^2+bt^2);
    d12=-s^2-ni*al^2;
    d14=r^2-ni*al^2;
    d21=-s^3-(2-ni)*al^2*s;
    d23=r^3-(2-ni)*al^2*r;
    d31=d12*sin(s*b);
    d32=d12*cos(s*b);
    d33=d14*sinh(r*b);
    d34=d14*cosh(r*b);
    d41=d21*cos(s*b);
    d42=-d21*sin(s*b);
    d43=d23*cosh(r*b);
    d44=d23*sinh(r*b);
    l=[1 det([0 d12 0 d14; d21 0 d23 0; d31 d32 d33 d34; d41 d42 d43 d44])];
end
pha=[pi/a:pi/a:4*pi/a];
ta0=[10.305433164820414 20.742175557961834 31.202948053539938
    .6676890569522];
ta1=[15.5981 27.0127 41.2031 56.2723];
ta2=[24.6893 33.9861 46.1196 59.8559];
ta3=[34.1828 42.0535 52.6843 65.1033];
ta4=[50.7225 97.4771 125.95795 155.6079];
=sqrt(beta1.^2-(pi/a)^2);
=sqrt(beta1.^2+(pi/a)^2);
=sqrt(beta2.^2-(2*pi/a)^2);
=sqrt(beta2.^2+(2*pi/a)^2);
=sqrt(beta3.^2-(3*pi/a)^2);
=sqrt(beta3.^2+(3*pi/a)^2);

```

```

4=sqrt(beta4.^2-(4*pi/a)^2);
4=sqrt(beta4.^2+(4*pi/a)^2);
1=r1.*(cos(s1.*b)-cosh(r1.*b)).*(beta1.^4-alpha(1)^4*(1-ni)^2).*(beta1.^2-
lpha(1)^2*(1-ni));
1=(beta1.^2+alpha(1)^2*(1-ni)).*(s1.*(beta1.^2+alpha(1)^2*(1-
i)).^2.*sinh(r1.*b)-r1.*(beta1.^2-alpha(1)^2*(1-ni)).^2.*sin(s1.*b));
1=s1.*(cos(s1.*b)-cosh(r1.*b)).*(beta1.^4-alpha(1)^4*(1-
i)^2).*(beta1.^2+alpha(1)^2*(1-ni));
1=(beta1.^2-alpha(1)^2*(1-ni)).*(s1.*(beta1.^2+alpha(1)^2*(1-
i)).^2.*sinh(r1.*b)-r1.*(beta1.^2-alpha(1)^2*(1-ni)).^2.*sin(s1.*b));
2=r2.*(cos(s2.*b)-cosh(r2.*b)).*(beta2.^4-alpha(2)^4*(1-ni)^2).*(beta2.^2-
lpha(2)^2*(1-ni));
2=(beta2.^2+alpha(2)^2*(1-ni)).*(s2.*(beta2.^2+alpha(2)^2*(1-
i)).^2.*sinh(r2.*b)-r2.*(beta2.^2-alpha(2)^2*(1-ni)).^2.*sin(s2.*b));
2=s2.*(cos(s2.*b)-cosh(r2.*b)).*(beta2.^4-alpha(2)^4*(1-
i)^2).*(beta2.^2+alpha(2)^2*(1-ni));
2=(beta2.^2-alpha(2)^2*(1-ni)).*(s2.*(beta2.^2+alpha(2)^2*(1-
i)).^2.*sinh(r2.*b)-r2.*(beta2.^2-alpha(2)^2*(1-ni)).^2.*sin(s2.*b));
3=r3.*(cos(s3.*b)-cosh(r3.*b)).*(beta3.^4-alpha(3)^4*(1-ni)^2).*(beta3.^2-
lpha(3)^2*(1-ni));
3=(beta3.^2+alpha(3)^2*(1-ni)).*(s3.*(beta3.^2+alpha(3)^2*(1-
i)).^2.*sinh(r3.*b)-r3.*(beta3.^2-alpha(3)^2*(1-ni)).^2.*sin(s3.*b));
3=s3.*(cos(s3.*b)-cosh(r3.*b)).*(beta3.^4-alpha(3)^4*(1-
i)^2).*(beta3.^2+alpha(3)^2*(1-ni));
3=(beta3.^2-alpha(3)^2*(1-ni)).*(s3.*(beta3.^2+alpha(3)^2*(1-
i)).^2.*sinh(r3.*b)-r3.*(beta3.^2-alpha(3)^2*(1-ni)).^2.*sin(s3.*b));
4=r4.*(cos(s4.*b)-cosh(r4.*b)).*(beta4.^4-alpha(4)^4*(1-ni)^2).*(beta4.^2-
lpha(4)^2*(1-ni));
4=(beta4.^2+alpha(4)^2*(1-ni)).*(s4.*(beta4.^2+alpha(4)^2*(1-
i)).^2.*sinh(r4.*b)-r4.*(beta4.^2-alpha(4)^2*(1-ni)).^2.*sin(s4.*b));
4=s4.*(cos(s4.*b)-cosh(r4.*b)).*(beta4.^4-alpha(4)^4*(1-
i)^2).*(beta4.^2+alpha(4)^2*(1-ni));
4=(beta4.^2-alpha(4)^2*(1-ni)).*(s4.*(beta4.^2+alpha(4)^2*(1-
i)).^2.*sinh(r4.*b)-r4.*(beta4.^2-alpha(4)^2*(1-ni)).^2.*sin(s4.*b));
1=A1(1)*sin(s1(1)*y)+B1(1)*cos(s1(1)*y)+C1(1)*sinh(r1(1)*y)+D1(1)*cosh(r1(1)*y
2=A1(2)*sin(s1(2)*y)+B1(2)*cos(s1(2)*y)+C1(2)*sinh(r1(2)*y)+D1(2)*cosh(r1(2)*y
3=A1(3)*sin(s1(3)*y)+B1(3)*cos(s1(3)*y)+C1(3)*sinh(r1(3)*y)+D1(3)*cosh(r1(3)*y
4=A1(4)*sin(s1(4)*y)+B1(4)*cos(s1(4)*y)+C1(4)*sinh(r1(4)*y)+D1(4)*cosh(r1(4)*y
1=A2(1)*sin(s2(1)*y)+B2(1)*cos(s2(1)*y)+C2(1)*sinh(r2(1)*y)+D2(1)*cosh(r2(1)*y
2=A2(2)*sin(s2(2)*y)+B2(2)*cos(s2(2)*y)+C2(2)*sinh(r2(2)*y)+D2(2)*cosh(r2(2)*y
3=A2(3)*sin(s2(3)*y)+B2(3)*cos(s2(3)*y)+C2(3)*sinh(r2(3)*y)+D2(3)*cosh(r2(3)*y
4=A2(4)*sin(s2(4)*y)+B2(4)*cos(s2(4)*y)+C2(4)*sinh(r2(4)*y)+D2(4)*cosh(r2(4)*y
1=A3(1)*sin(s3(1)*y)+B3(1)*cos(s3(1)*y)+C3(1)*sinh(r3(1)*y)+D3(1)*cosh(r3(1)*y
2=A3(2)*sin(s3(2)*y)+B3(2)*cos(s3(2)*y)+C3(2)*sinh(r3(2)*y)+D3(2)*cosh(r3(2)*y
3=A3(3)*sin(s3(3)*y)+B3(3)*cos(s3(3)*y)+C3(3)*sinh(r3(3)*y)+D3(3)*cosh(r3(3)*y

```

```

34=A3(4)*sin(s3(4)*y)+B3(4)*cos(s3(4)*y)+C3(4)*sinh(r3(4)*y)+D3(4)*cosh(r3(4)*y
;
41=A4(1)*sin(s4(1)*y)+B4(1)*cos(s4(1)*y)+C4(1)*sinh(r4(1)*y)+D4(1)*cosh(r4(1)*y
;
42=A4(2)*sin(s4(2)*y)+B4(2)*cos(s4(2)*y)+C4(2)*sinh(r4(2)*y)+D4(2)*cosh(r4(2)*y
;
43=A4(3)*sin(s4(3)*y)+B4(3)*cos(s4(3)*y)+C4(3)*sinh(r4(3)*y)+D4(3)*cosh(r4(3)*y
;
44=A4(4)*sin(s4(4)*y)+B4(4)*cos(s4(4)*y)+C4(4)*sinh(r4(4)*y)+D4(4)*cosh(r4(4)*y
;
11=Y11/max(abs(Y11)); Y12=Y12/max(abs(Y12)); Y13=Y13/max(abs(Y13));
14=Y14/max(abs(Y14));
21=Y21/max(abs(Y21)); Y22=Y22/max(abs(Y22)); Y23=Y23/max(abs(Y23));
24=Y24/max(abs(Y24));
31=Y31/max(abs(Y31)); Y32=Y32/max(abs(Y32)); Y33=Y33/max(abs(Y33));
34=Y34/max(abs(Y34));
41=Y41/max(abs(Y41)); Y42=Y42/max(abs(Y42)); Y43=Y43/max(abs(Y43));
44=Y44/max(abs(Y44));
isplay('Caso 1: beta^4 < alpha^4;
=A*sin(s*y)+B*cos(s*y)+C*sinh(r*y)+D*cosh(r*y)');
isplay('onde s=sqrt(-alpha^2+beta^2) e r=sqrt(alpha^2+beta^2)');
isplay(beta1);
isplay(beta2);
isplay(beta3);
isplay(beta4);
figure(1)
plot(al+al/100*(0:1:1200/(n+1)),1);
axis on;
label('Beta');
label('det(M)');
axis([al 11*al -1 1]);

caso 2: alpha^4 = beta^4; Y=A+B*y+C*exp(-sqrt(2)*al*y)+D*exp(sqrt(2)*al*y)
=[-ni*al^2 0 al^2*(2-ni) al^2*(2-ni) ; 0 -al^2*(2-ni) -sqrt(2)*al^3*ni
sqrt(2)*al^3*ni ; -ni*al^2 -ni*al^2*b al^2*(2-ni)*exp(-sqrt(2)*al*b) al^2*(2-
)*exp(sqrt(2)*al*b) ; 0 -al^2*(2-ni) -sqrt(2)*al^3*ni*exp(-sqrt(2)*al*b)
sqrt(2)*al^3*ni*exp(sqrt(2)*al*b)];
detMatriz=det(m);
isplay('Caso 2: alpha^4 = beta^4; Y=A+B*y+C*exp(-
sqrt(2)*alpha*y)+D*exp(sqrt(2)*alpha*y)');
isplay(detMatriz);
matriz triangularizada
isplay('Matriz triangularizada');
matriz=[ -ni*al^2 -ni*al^2*b al^2*(2-ni)*exp(-sqrt(2)*al*b) al^2*(2-
)*exp(sqrt(2)*al*b) ; 0 -al^2*(2-ni) -sqrt(2)*al^3*ni*exp(-sqrt(2)*al*b)
sqrt(2)*al^3*ni*exp(sqrt(2)*al*b) ; 0 0 sqrt(2)*al^3*ni*(exp(-sqrt(2)*al*b)-1) -
sqrt(2)*al^3*ni*(exp(sqrt(2)*al*b)-1) ; 0 0 0 -
sqrt(2)*al^5*ni*(exp(sqrt(2)*al*b)-1)*((2-ni)^2*(exp(-
sqrt(2)*al*b)+sqrt(2)*al*ni^2*b))-sqrt(2)*al^5*ni*(exp(-sqrt(2)*al*b)-1)*((2-
)^2*(exp(sqrt(2)*al*b)-sqrt(2)*al*ni^2*b))];
isplay(Matriz);
isplay('Como detMatriz /= 0 e a Matriz triangularizada nao apresenta zeros na
agonal principal,');
isplay('nao ha modos de vibrar neste caso.');
```

```

caso 3: beta^4 < alpha^4; Y=A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y)
[];
```

```

or i=0:1000
    bt=i*al/1000;
    p=sqrt(al^2-bt^2);
    r=sqrt(al^2+bt^2);
    e12=p^2-ni*al^2;
    e14=r^2-ni*al^2;
    e21=p^3-(2-ni)*al^2*p;
    e23=r^3-(2-ni)*al^2*r;
    e31=e21*cosh(p*b);
    e32=e21*sinh(p*b);
    e33=e23*cosh(r*b);
    e34=e23*sinh(r*b);
    e41=e12*sinh(p*b);
    e42=e12*cosh(p*b);
    e43=e14*sinh(r*b);
    e44=e14*cosh(r*b);
    o=[o det([0 e12 0 e14; e21 0 e23 0; e31 e32 e33 e34; e41 e42 e43 e44])];
end
figure(2)
plot(al/1000*(0:1:1000),o);
axis on;
xlabel('Beta');
ylabel('det(M)');
text(0.05, 0.05, 'Caso 3: beta^4 < alpha^4;');
text(0.05, 0.1, 'A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y)');
text(0.05, 0.15, 'onde p=sqrt(alpha^2-beta^2) e r=sqrt(alpha^2+beta^2)');
text(0.05, 0.2, 'Beta01=10.305433164820414; Beta02=20.742175557961834;');
text(0.05, 0.25, 'Beta03=31.202948053539938; Beta04=41.6676890569522;');
or n=1:4
    if(n==1)
        bt=10.305433164820414;
        al=pi/a;
    end
    if(n==2)
        bt=20.742175557961834;
        al=2*pi/a;
    end
    if(n==3)
        bt=31.202948053539938;
        al=3*pi/a;
    end
    if(n==4)
        bt=41.6676890569522;
        al=4*pi/a;
    end
    p=sqrt(al^2-bt^2)
    r=sqrt(al^2+bt^2)
    e12=p^2-ni*al^2;
    e14=r^2-ni*al^2;
    e21=p^3-(2-ni)*al^2*p;
    e23=r^3-(2-ni)*al^2*r;
    e31=e21*cosh(p*b);
    e32=e21*sinh(p*b);
    e33=e23*cosh(r*b);
    e34=e23*sinh(r*b);
    e41=e12*sinh(p*b);
    e42=e12*cosh(p*b);

```

```

e43=e14*sinh(r*b);
e44=e14*cosh(r*b);
detM=det([0 e12 0 e14; e21 0 e23 0; e31 e32 e33 e34; e41 e42 e43 e44]);
E=e12*sinh(p*b)-e14*e21/e23*sinh(r*b)+e21*(cosh(p*b)-cosh(r*b));
F=e21*sinh(p*b)-e23*e12/e14*sinh(r*b)+e12*(cosh(p*b)-cosh(r*b));
A=1
B=-E/F
C=-e21/e23
D=e12*E/e14/F
eq0=sinh(p*b)*sinh(r*b)*(r^2*(bt^2-(1-ni)*al^2)^4+p^2*(bt^2+(1-ni)*al^2)^4)-
*p*r*(cosh(p*b)*cosh(r*b)-1)*(bt^4-(1-ni)^2*al^4)^2
if(n==1)
    Y01=A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y);
    Y01=-Y01/max(abs(Y01));
end
if(n==2)
    Y02=A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y);
    Y02=-Y02/max(abs(Y02));
end
if(n==3)
    Y03=A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y);
    Y03=-Y03/max(abs(Y03));
end
if(n==4)
    Y04=A*sinh(p*y)+B*cosh(p*y)+C*sinh(r*y)+D*cosh(r*y);
    Y04=-Y04/max(abs(Y04));
end
end
modos de vibrar 0i, na direcao de y.
figure(3)
plot(0:1:100*b,Y01,0:1:100*b,Y02,0:1:100*b,Y03,0:1:100*b,Y04);
axis on;
xlabel('y (cm)');
ylabel('Amplitude');
legend('Y01','Y02','Y03','Y04');
modos de vibrar 01, 02, 03, 04, 11, 12, 13, 14, 21, 22, 23, 24, 31, 32, 33,
34, 41, 42, 43 e 44.
w01=X1*Y01; w02=X2*Y02; w03=X3*Y03; w04=X4*Y04;
w11=X1*Y11; w12=X1*Y12; w13=X1*Y13; w14=X1*Y14;
w21=X2*Y21; w22=X2*Y22; w23=X2*Y23; w24=X2*Y24;
w31=X3*Y31; w32=X3*Y32; w33=X3*Y33; w34=X3*Y34;
w41=X4*Y41; w42=X4*Y42; w43=X4*Y43; w44=X4*Y44;
b=(beta0).^2*3.12836235./(2*pi);
b=(beta1).^2*3.12836235./(2*pi);
b=(beta2).^2*3.12836235./(2*pi);
b=(beta3).^2*3.12836235./(2*pi);
f=[f0; f1; f2; f3];
w=[w01 w02 w03 w04; w11 w12 w13 w14; w21 w22 w23 w24; w31 w32 w33 w34; w41 w42
w43 w44; ];
filme ou figura dos modos de vibrar 01, 02, 03, 04, 11, 12, 13, 14, 21, 22,
23, 24, 31, 32, 33, 34, 41, 42, 43 e 44.
resp=input('Deseja ver animacoes dos modos de vibrar? (1=Sim, outro
numero=Nao)');
for i=1:4
    for j=1:4
        figure(3+4*(i-1)+j);
        surf(ww(31*(i-1)+1:31*i,21*(j-1)+1:21*j));
    end
end

```

```

        axis([0 20 0 30 -1 1]);
        title([' ', num2str(j), 'o. modo de vibrar analitico - frecuencia:
,num2str(f(i,j)), ' Hz']);
        ylabel('x(cm)');
        xlabel('y(cm)');
        zlabel('z(mm)');
        F=getframe;
modo(1)=getframe;
if resp==1
    for(n=2:18)
        WW=ww*cos((n-1)*pi/9);
        surf(WW);
        axis([0 20 0 30 -1 1]);
        modo(n)=getframe;
    end
    movie2avi(modo,'MODO',num2str(i,j),'.avi');
end
end
end
id

```