6
Referências Bibliográficas

ALIPOUR, M. M.; SHARIYAT, M. Semi-analytical bukling analysis of heterogeneus variable thickness viscoelastic circular plates on elastic foundations. **Mechanics research communications**, v. 38, p. 594-601, September 2011.

BATHE, K.-J. **Finite Element Procedure**. United Estates of America: Prentice Hall, 1996.

BAZANT, Z. P.; CEDOLIN, L. **Stability of Structures**. New Jersey: Word Scientific, 2010.

CHAKRAVERTY, S. **Vibration of Plates**. New York: Taylor & Francis Group, 2009.

COOK, R. D. et al. **Concepts And Applications of Finite Element Analysis**. Fourth. ed. United State: John Wiley & Sons, 2002.

HOSSEINI-HASHEMI, S. et al. Exact cloesd-form frequency equations for thick circular plates using a third-order shear deformation theory. **Journal of Sound and Vibration**, v. 329, p. 3382-3396, March 2010.

INMAN, D. J. **Engineering Vibrations**. Fourth. ed. United Estate of America: Pearson, 2012.

KOOHKAN, H. et al. **An analitical approach on the buckling analysis of circular, solid and annular functionally graded thin plates**. [S.l.]: Journal of Mechanical Engineering, v. 41, 2010.

LEISSA, A. W. **Vibration of Plates**. Washinton, D.C.: Office of Technology Utilization National Aeronautics and Space Administration, 1969.

LIEW, K. M.; YANG, B. **Elasticity solutions for free Vibrations of annular plates from three-dimensional analysis**. Singapure: international journal of Solid and Structures, 30 May 1998.

MAZHARI, E.; SHAHIDI, A. R. Analysis of post buckling behavior of circular plates with non-concentric hole using the Rayleigh-Ritz method. **Applied Mathenatical Modelling**, v. 35, p. 3136-3153, December 2011.

REDDY, J. N. **Mechanics of Lamineted Composite Plates and Shells Theory and Analysis**. Second. ed. Washington D. C.: CRC Press LLC, 2004.

REDDY, J. N. **Theory and Analysis of Elastic Plates and Shells**. Second. ed. United States of America: CRC Press, 2007.

SUANNO, R.; ROSAS E SILVA, R. Influence of Distributio, Shear Deformation, and Rotary Inertia on Flutter Loads. **The American Society of Mechanical Enginners**, v. 42, p. 249-252, june 2009.

TIMOSHENKO, S. P.; GERE, J. M. **Theory of Elastic Stability**. Second. ed. New York: Mc-Graw-Hill, 1963.

TIMOSHENKO, S. P.; WOINOWSKY-KRIEGER, S. **Theory of Plates and Shells**. Second. ed. New York: McGraw-Hill, 1959.

TIMOSHENKO, S.; GOODIER, J. N. **Theory of Elasticity**. New York: McGraw-Hill, 1934.

VENTSEL, E.; KRAUTHAMMER, T. **Thin Plates and Shells**. New York: Marcel Dekker, 2001.

WANG, C. M. et al. Axisymmetric Buckling of Circular Mindlin Plates With Ring Supports. **Journal of Structural Engineering**, v. 119, p. 782-793, 1993.

WEAVER, JR, W.; TIMOSHENKO, S. P.; YOUNG, D. H. **Vibration Problems in Engineering**. Fifth. ed. United Stated of America: John Wiley & Sons, 1990.

YOUNG, W. C. **Roark's formulas for Stress & Strain**. Sixth. ed. Singapore: McGraw-Hill, 1989.

ZIENKIEWICZ, O. C.; TAYLOR, R. L. **El Metodo de los Elementos Finitos**. Cuarta. ed. Madrid: McGraw-Hill, v. 1, 1994.

Anexo

A. Modelo Computacional para a obtenção de esforços e deslocamentos placa fina.

Calculo da placa circular com teoria de placa fina

```
> restart : with(plots) : with(LinearAlgebra) :
> : with(linalg) : with(ExcelTools) : with(plottools) :
```

Calculo do polinômio adicional

> $Eq1 := w(x) = A1 + B1\cdot\left(\frac{x}{L}\right) + C1\cdot\left(\frac{x}{L}\right)^2 + D1\cdot\left(\frac{x}{L}\right)^3 + \left(\frac{x}{L}\right)^{n+3}$:

> $Eq1a := subs(x = 0, rhs(Eq1) = 0)$:

> $Eq1b := subs(x = L, rhs(Eq1) = 0)$:

> $Eq2 := dw(x) = simplify\left(rhs\left(\frac{\partial}{\partial x} Eq1\right)\right)$:

> $Eq2a := subs(x = 0, rhs(Eq2) = 0)$:

> $Eq2b := subs(x = L, rhs(Eq2) = 0)$:

> $Solution := simplify(solve(\{Eq1a, Eq1b, Eq2a, Eq2b\}, \{A1, B1, C1, D1\}))$:

> $A1 := rhs(Solution_1) : B1 := rhs(Solution_2) : C1 := rhs(Solution_3) : D1 := rhs(Solution_4)$:

> $Eq3 := rhs(Eq1)$:

> $Eq4 := subs(L = a - b, x = r - b, w(r) = Eq3)$:

> #a :=3: n:=1:b:=9:

> #plot(rhs(Eq4), r = 3..9) :

Funções Polinomiais Cubicas Básicas $N_{pb} = 4$

Numero de funciones Basicas

> $N_{pb} := 4$:

Numero de funções adicionais

Numero de funções adicionais em r

> $N_r := 4$:

Numero de funções trigonométricas

> $N_\theta := 0$: Placa_Fina_es

Introdução de dados necessários para o calculo

Características geométricas do elemento (Dimensões da placa circular) [L]

> $b := 3 : a := 9 : h0 := 0.2 : h1 := 0.6$:

Densidade de massa [F/L^3]

> $\rho := 0$:

Função da variação de espessura "h"

> $h := \left(\frac{(h1 - h0)}{(a-b)} \cdot (r-b)\right) + h0 :$

Características do material

Coeficiente de Poisson

> $\nu := 0.3 :$

Modulo de Elasticidade Longitudinal do material [F/L^2]

> $E := 2 \cdot 10^{10} :$

Rigidez a flexão da placa

> $Do := \frac{E \cdot h^3}{12 \cdot (1 - \nu^2)} :$

Caracteristicas do Suelo [F/ L]

Coeficiente de rigidez

> $k := 0 :$

Condições das cargas externas

Função do carregamento

> $qz := qo + \frac{(q1 - qo)}{(a-b)} \cdot (r-b) + po \cdot \left(\frac{r}{a}\right) \cdot \sin(N_\theta \cdot \theta) :$

Valor da carga linear no bordo interno [F/L]

> $qo := -5000 :$

Valor da carga linear no bordo externo [F/L]

> $q1 := -5000 :$

Carregamento de variação linear [F/L]

> $po := 0 :$

Carregamento Puntual[F]

> $Pz := -\frac{0}{2 \cdot \pi \cdot b} :$

Condiciones de Apoyo

> $mola_1 := 0 \cdot 10^{15} :$

> $mola_2 := 1 \cdot 10^{15} :$

> $mola_3 := 1 \cdot 10^{15} :$

> $mola_4 := 1 \cdot 10^{15} :$

Armado do vetor Campo de deslocamentos

> $interface(rtablesize = 50) :$

> if $N_r = 0$ and $N_\theta = 0$ then

> $fp := Matrix(N_{pb}, 1, 0)$:

> $fp_{1,1} := 1 - \frac{3 \cdot x^2}{L^2} + \frac{2 \cdot x^3}{L^3}$: $fp_{2,1} := x - \frac{2 \cdot x^2}{L} + \frac{x^3}{L^2}$: $fp_{3,1} := \frac{3 \cdot x^2}{L^2} - 2 \cdot \frac{x^3}{L^3}$:
$fp_{4,1} := -\frac{x^2}{L} + \frac{x^3}{L^2}$:

> for i from 1 to N_{pb} do

> $fp_{i,1} := collect(subs(L = a - b, x = r - b, fp_{i,1}), \{r\})$:

> end do:

> $w := fp$

> $dw1r := Matrix(N_{pb}, 1, 0)$:

> for i from 1 to N_{pb} do

> $dw1r_{i,1} := \frac{\partial}{\partial r} w_{i,1}$

> end do:

> $dwr := dw1r$

> $ddwr := Matrix(N_{pb}, 1, 0)$:

> for i from 1 to N_{pb} do

> $ddwr_{i,1} := \frac{\partial}{\partial r} dwr_{i,1}$

> end do:

> $ddwr := ddwr$

> $dwr\theta := Matrix(N_{pb}, 1, 0)$:

> for i from 1 to N_{pb} do

> $dwr\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial r} w_{i,1}$

> end do:

> $dwr\theta := dwr\theta$

> $dw\theta := Matrix(N_{pb}, 1, 0)$:

> for i from 1 to N_{pb} do

> $dw\theta_{i,1} := \frac{\partial}{\partial \theta} w_{i,1}$

> end do:

> $dw\theta := dw\theta$

> $ddw\theta := Matrix(N_{pb}, 1, 0)$:

> for i from 1 to N_{pb} do

> $ddw\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial \theta} w_{i,1}$

> end do:

> $ddw\theta := ddw\theta$

Cambio de condicion

> elif $N_r > 0$ and $N_\theta = 0$ then

> $fp := Matrix(N_{pb} + N_r, 1, 0)$:

> $fp_{1,1} := 1 - \frac{3 \cdot x^2}{L^2} + \frac{2 \cdot x^3}{L^3}$: $fp_{2,1} := x - \frac{2 \cdot x^2}{L} + \frac{x^3}{L^2}$: $fp_{3,1} := \frac{3 \cdot x^2}{L^2} - 2 \cdot \frac{x^3}{L^3}$:
$fp_{4,1} := -\frac{x^2}{L} + \frac{x^3}{L^2}$:

$>$ **for** i **from** 1 **to** N_{pb} **do**

$>$ $fp_{i,1} := collect\left(subs\left(L = a - b, x = r - b, fp_{i,1}\right), \{r\}\right)$

$>$ **end do**:

$>$ **for** n **from** 1 **to** N_r **do**

$>$ $fp_{N_{pb}+n,1} := rhs(Eq4)$

$>$ **end do**:

$>$ $w := fp$

$>$ $dw1r := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ **for** i **from** 1 **to** $N_{pb} + N_r$ **do**

$>$ $dw1r_{i,1} := \frac{\partial}{\partial r} w_{i,1}$

$>$ **end do**:

$>$ $dwr := dw1r$:

$>$ $ddwr := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ **for** i **from** 1 **to** $N_{pb} + N_r$ **do**

$>$ $ddwr_{i,1} := \frac{\partial}{\partial r} dwr_{i,1}$

$>$ **end do**:

$>$ $ddwr := ddwr$:

$>$ $dwr\theta := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ **for** i **from** 1 **to** $N_{pb} + N_r$ **do**

$>$ $dwr\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial r} w_{i,1}$

$>$ **end do**:

$>$ $dwr\theta := dwr\theta$:

$>$ $dw\theta := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ **for** i **from** 1 **to** $N_{pb} + N_r$ **do**

$>$ $dw\theta_{i,1} := \frac{\partial}{\partial \theta} w_{i,1}$

$>$ **end do**:

$>$ $dw\theta := dw\theta$:

$>$ $ddw\theta := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ **for** i **from** 1 **to** $N_{pb} + N_r$ **do**

$>$ $ddw\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial \theta} w_{i,1}$

$>$ **end do**:

$>$ $ddw\theta := ddw\theta$:

Cambio de condicion

$>$ **elif** $N_r > 0$ **and** $N_\theta > 0$ **then**

$>$ $interface(rtablesize = 50)$

$>$ $fp := Matrix\left(N_{pb} + N_r, 1, 0\right)$:

$>$ $fp_{1,1} := 1 - \frac{3 \cdot x^2}{L^2} + \frac{2 \cdot x^3}{L^3} : fp_{2,1} := x - \frac{2 \cdot x^2}{L} + \frac{x^3}{L^2} : fp_{3,1} := \frac{3 \cdot x^2}{L^2} - 2 \cdot \frac{x^3}{L^3} :$
$fp_{4,1} := -\frac{x^2}{L} + \frac{x^3}{L^2} :$

$>$ **for** i **from** 1 **to** N_{pb} **do**

> $fp_{i,1} := collect(subs(L = a - b, x = r - b, fp_{i,1}), \{r\})$:
> **end do**:
> **for** n **from** 1 **to** N_r **do**
> $fp_{N_{pb}+n,1} := rhs(Eq4)$
> **end do**:
> $ft := Matrix(N_\theta \cdot (N_{pb} + N_r), 1, 0)$:
> $tot := 0$:
> **for** k **from** 1 **by** 1 **while** $k \le N_\theta$ **do**
> **for** i **from** 1 **to** $N_{pb} + N_r$ **do**
> $ft_{i+tot,1} := fp_{i,1} \cdot \sin(k \cdot \theta)$
> **end do**
> $tot := (N_{pb} + N_r) \cdot k$
> **end do**:
> $w := \langle fp, ft \rangle$
> $dw1r := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:
> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**
> $dw1r_{i,1} := \frac{\partial}{\partial r} w_{i,1}$
> **end do**:
> $dwr := dw1r$
> $ddwr := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:
> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**
> $ddwr_{i,1} := \frac{\partial}{\partial r} dwr_{i,1}$
> **end do**:
> $ddwr := ddwr$
> $dwr\theta := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:
> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**
> $dwr\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial r} w_{i,1}$
> **end do**:
> $dwr\theta := dwr\theta$
> $dw\theta := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:
> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**
> $dw\theta_{i,1} := \frac{\partial}{\partial \theta} w_{i,1}$
> **end do**:
> $dw\theta := dw\theta$
> $ddw\theta := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:
> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**
> $ddw\theta_{i,1} := \frac{\partial}{\partial \theta} \frac{\partial}{\partial \theta} w_{i,1}$
> **end do**:
> $ddw\theta := ddw\theta$

Cambio de condicion

> **elif** $N_r = 0$ **and** $N_\theta > 0$ **then**
> $interface(rtablesize = 50)$

```
> fp := Matrix(N_pb, 1, 0) :
> fp[1,1] := 1 - 3·x^2/L^2 + 2·x^3/L^3 : fp[2,1] := x - 2·x^2/L + x^3/L^2 : fp[3,1] := 3·x^2/L^2 - 2·x^3/L^3 :
      fp[4,1] := -x^2/L + x^3/L^2 :
> for i from 1 to N_pb do
> fp[i,1] := collect(subs(L = a - b, x = r - b, fp[i,1]), {r})
> end do:
> ft := Matrix(N_θ·(N_pb), 1, 0) :
> tot := 0 :
> for k from 1 by 1 while k ≤ N_θ do
> for i from 1 to N_pb + N_r do
> ft[i+tot,1] := fp[i,1]·sin(k·θ)
> end do
> tot := (N_pb)·k
> end do:
> w := ⟨fp, ft⟩
> dw1r := Matrix((N_θ + 1)·N_pb, 1, 0) :
> for i from 1 to (N_θ + 1)·N_pb do
> dw1r[i,1] := ∂/∂r w[i,1]
> end do:
> dwr := dw1r
> ddwr := Matrix((N_θ + 1)·N_pb, 1, 0) :
> for i from 1 to (N_θ + 1)·N_pb do
> ddwr[i,1] := ∂/∂r dwr[i,1]
> end do:
> ddwr := ddwr
> dwrθ := Matrix((N_θ + 1)·N_pb, 1, 0) :
> for i from 1 to (N_θ + 1)·N_pb do
> dwrθ[i,1] := ∂/∂θ ∂/∂r w[i,1]
> end do:
> dwrθ := dwrθ
> dwθ := Matrix((N_θ + 1)·N_pb, 1, 0) :
> for i from 1 to (N_θ + 1)·N_pb do
> dwθ[i,1] := ∂/∂θ w[i,1]
> end do:
> dwθ := dwθ
> ddwθ := Matrix((N_θ + 1)·N_pb, 1, 0) :
> for i from 1 to (N_θ + 1)·N_pb do
> ddwθ[i,1] := ∂/∂θ ∂/∂θ w[i,1]
> end do:
```

```
> ddwθ := ddwθ
> end if:
```

Deformada y sus Derivadas Parciales

```
> w :
> dwr :
> ddwr :
> dwrθ:
> dwθ:
> ddwθ:
```

Calculo das integrais

Integração da primeira parte

> $V1 := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:

> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**

> $V1_{i,1} := \left(ddwr_{i,1} + \frac{1}{r} \cdot dwr_{i,1} + \frac{1}{r^2} \cdot ddw\theta_{i,1}\right)$

> **end do**:

> $V1T := \mathrm{Transpose}(V1)$:

> $M1 := Multiply(V1, V1T)$:

> $INT1 := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), (N_\theta + 1) \cdot (N_{pb} + N_r), 0)$:

> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**

> **for** j **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**

> $INT1_{i,j} := evalf\left(\int_b^a \int_0^{2\pi} Do \cdot M1_{i,j} \cdot r \, \mathrm{d}\theta \, \mathrm{d}r\right)$

> **end do**

> **end do**

> $INT1 := INT1$:

Integração da segunda parte

> $V2 := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:

> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**

> $V2_{i,1} := (ddwr_{i,1})$

> **end do**:

> $V3 := Matrix((N_\theta + 1) \cdot (N_{pb} + N_r), 1, 0)$:

> **for** i **from** 1 **to** $(N_\theta + 1) \cdot (N_{pb} + N_r)$ **do**

> $V3_{i,1} := \left(\frac{1}{r} \cdot dwr_{i,1} + \frac{1}{r^2} \cdot ddw\theta_{i,1}\right)$

> **end do**:

> $\mathrm{V2T} := \mathrm{Transpose}(V2)$:

> $V3T := Transpose(V3)$:

```
> M2a := Multiply(V2, V3T) :
> M2b := Multiply(V3, V2T) :
> M2 := Matrix((ν − 1)·M2a + (ν − 1)·M2b) :
> INT2 := Matrix((N_θ + 1)·(N_pb + N_r), (N_θ + 1)·(N_pb + N_r), 0) :
> for i from 1 to (N_θ + 1)·(N_pb + N_r) do
> for j from 1 to (N_θ + 1)·(N_pb + N_r) do
> INT2[i, j] := evalf(∫_b^a ∫_0^(2π) Do·M2[i, j]·r dθ dr)
> end do
> end do:
> INT2 := INT2 :
```

Integração da terceira parte

```
> V4 := Matrix((N_θ + 1)·(N_pb + N_r), 1, 0) :
> for i from 1 to (N_θ + 1)·(N_pb + N_r)do
> V4[i, 1] := ((1/r)·dwrθ[i, 1] − (1/r^2)·dwθ[i, 1])
> end do:
> V4T := Transpose(V4) :
> M3 := 2·(1 − ν)·Multiply(V4, V4T) :
> INT3 := Matrix((N_θ + 1)·(N_pb + N_r), (N_θ + 1)·(N_pb + N_r), 0) :
> for i from 1 to(N_θ + 1)·(N_pb + N_r) do
> for j from 1 to (N_θ + 1)·(N_pb + N_r) do
> INT3[i, j] := evalf(∫_b^a ∫_0^(2π) Do·M3[i, j]·rdθ dr)
> end do
> end do:
> INT3 := INT3 :
```

Matriz de Apoios

```
> Km := Matrix((N_pb + N_r)·(N_θ + 1), (N_pb + N_r)·(N_θ + 1), 0) :
> interface(rtablesize = 50) :

> if N_r = 0 and N_θ = 0then
> Rm := Matrix(N_pb, 1, 0) :
> Rm[1, 1] := mola[1] : Rm[2, 1] := mola[2] : Rm[3, 1] := mola[3] : Rm[4, 1] := mola[4] :
> VR := Rm

> elif N_r > 0 and N_θ = 0 then
> Rm := Matrix(N_pb + N_r, 1, 0) :
> Rm[1, 1] := mola[1] : Rm[2, 1] := mola[2] : Rm[3, 1] := mola[3] : Rm[4, 1] := mola[4] :
> for n from 1 to N_r do
```

```
> Rm[N[pb]+n, 1] := 0
> end do:
> VR := Rm

> elif N[r] > 0 and N[θ] > 0 then
> Rm := Matrix(N[pb] + N[r], 1, 0) :
> Rm[1, 1] := mola[1] : Rm[2, 1] := mola[2] : Rm[3, 1] := mola[3] : Rm[4, 1] := mola[4] :
> for n from 1 to N[r] do
> Rm[N[pb]+n, 1] := 0
> end do:
> Rma := Matrix(N[θ]·(N[pb] + N[r]), 1, 0) :
> tot := 0 :
> for k from 1 by 1 while k ≤ N[θ] do
> for i from 1 to N[pb] + N[r] do
> Rma[i+tot, 1] := Rm[i, 1]·1
> end do
> tot := (N[pb] + N[r])·k
> end do:
> VR := ⟨Rm, Rma⟩

> elif N[r] = 0 and N[θ] > 0 then
> Rm := Matrix(N[pb], 1, 0) :
> Rm[1, 1] := mola[1] : Rm[2, 1] := mola[2] : Rm[3, 1] := mola[3] : Rm[4, 1] := mola[4] :
> Rma := Matrix(N[θ]·(N[pb]), 1, 0) :
> tot := 0 :
> for k from 1 by 1 while k ≤ N[θ] do
> for i from 1 to N[pb] + N[r] do
> Rma[i+tot, 1] := Rm[i, 1]
> end do
> tot := (N[pb])·k
> end do:
> VR := ⟨Rm, Rma⟩
> end if:
> VR :
> for i from 1 to (N[pb] + N[r])·(N[θ] + 1) do
> Km[i, i] := VR[i, 1]
> end do:
```

Matriz de rigidez elástica

```
> Ke := Matrix((N[θ] + 1)·(N[pb] + N[r]), (N[θ] + 1)·(N[pb] + N[r]), 0) :
> for i from 1 to (N[θ] + 1)·(N[pb] + N[r]) do
> for j from 1 to (N[θ] + 1)·(N[pb] + N[r]) do
> (Ke)[i, j] := (INT1[i, j] + INT2[i, j] + INT3[i, j])
> end do
> end do:
```

Matriz de rigidez de apoio elástico

> $wT := \text{Transpose}(w)$:

> $Buz := Multiply(w, wT)$:

> $Dimension(Buz)$:

> $KAp_{Elasico} := Matrix\left(\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right), \left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right), 0\right)$:

> **for** i **from** 1 **to** $\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right)$ **do**

> **for** j **from** 1 **to** $\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right)$ **do**

> $\left(KAp_{Elasico}\right)_{i,j} := evalf\left(k\cdot\int_{b}^{a}\int_{0}^{2\pi} Buz_{i,j}\cdot r\,\mathrm{d}\theta\,\mathrm{d}r\right)$

> **end do**

> **end do**:

> $KAp_{Elasico}$:

Vetor de carregamento distribuído

> $q := Matrix\left(\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right), 1, 0\right)$:

> **for** i **from** 1 **to** $\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right)$ **do**

> $q_{i,1} := evalf\left(\int_{b}^{a}\int_{0}^{2\cdot\pi} w_{i,1}\cdot \mathrm{qz}\cdot r\mathrm{d}\theta\,\mathrm{d}r\right)$

> **end do**:

Vetor de força pontual

> $P := Matrix\left(\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right), 1, 0\right)$:

> **for** i **from** 1 **to** $\left(N_{\theta}+1\right)\cdot\left(N_{pb}+N_{r}\right)$ **do**

> $P_{i,1} := evalf\left(\int_{0}^{b}\int_{0}^{2\cdot\pi} w_{i,1}\cdot \mathrm{Pz}\mathrm{d}\theta\,\mathrm{d}r\right)$

> **end do**:

Obtenção dos deslocamentos generalizados

> $K_{Elastica} := \left(Ke + Km + KAp_{Elasico}\right)$:

> $KInv := inverse\left(K_{Elastica}\right)$:

> $VF := (q + P)$:

> $Cn := evalm(KInv.VF)$:

Deformada

> $Deformada := expand(Transpose(w).Cn)$:

> $plot3d\left(\left[r, \text{theta}, Deformada_{1,1}\right], \text{theta} = 0..2*\text{Pi}, r = b..a, coords = cylindrical, axesfont = [\text{"ROMAN"}, 22]\right)$:

> $contourplot\left(\left[r, \text{theta}, Deformada_{1,1}\right], \text{theta} = 0..2*\text{Pi}, r = b..a, coords\right.$

$= cylindrical)$:

> $evalf\left(subs\left(\theta = \frac{\pi}{2}, r = b, Deformada_{1,1}\right)\right)$:

Momento Mr

> $Mr := -Do \cdot \left(\frac{\partial}{\partial r}\frac{\partial}{\partial r} Deformada_{1,1} + \nu \cdot \left(\frac{1}{r}\cdot\frac{\partial}{\partial r} Deformada_{1,1} + \frac{1}{r^2}\cdot\frac{\partial}{\partial \theta}\frac{\partial}{\partial \theta} Deformada_{1,1}\right)\right)$:

> plot3d([r, theta, Mr], theta = 0 .. 2 * Pi, r = b .. a, coords = cylindrical, axesfont = ["ROMAN", 22]) :

> contourplot([r, theta, Mr], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical) :

> $evalf\left(subs\left(\theta = \frac{\pi}{2}, r = b, Mr\right)\right)$:

Momento Mθ

> $M\theta := -Do \cdot \left(\frac{1}{r}\cdot\frac{\partial}{\partial r} Deformada_{1,1} + \frac{1}{r^2}\cdot\frac{\partial}{\partial \theta}\frac{\partial}{\partial \theta} Deformada_{1,1} + \nu\cdot\frac{\partial}{\partial r}\frac{\partial}{\partial r} Deformada_{1,1}\right)$:

> plot3d([r, theta, Mθ], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical, axesfont = ["ROMAN", 22]) :

> contourplot([r, theta, Mθ], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical) :

> $evalf\left(subs\left(\theta = \frac{\pi}{2}, r = 4.5, M\theta\right)\right)$:

Cortante Qr

> $Qr := -Do \cdot \left(\frac{\partial}{\partial r}\frac{\partial}{\partial r}\frac{\partial}{\partial r} Deformada_{1,1} + \frac{1}{r}\cdot\frac{\partial}{\partial r}\frac{\partial}{\partial r} Deformada_{1,1} - \frac{1}{r^2}\cdot\frac{\partial}{\partial r} Deformada_{1,1} + \frac{1}{r^2}\cdot\frac{\partial}{\partial r}\frac{\partial}{\partial \theta}\frac{\partial}{\partial \theta} Deformada_{1,1}\right)$:

> plot3d([r, theta, Qr], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical) :

> $evalf\left(subs\left(\theta = \frac{\pi}{2}, r = a, Qr\right)\right)$:

Tension Max

> $\sigma_{max} := -\frac{6\cdot Mr}{h^2}$:

> plot3d([r, theta, σ_{max}], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical, axesfont = ["ROMAN", 22]) :

> contourplot([r, theta, σ_{max}], theta = 0 ..2 * Pi, r = b ..a, coords = cylindrical) :

B. Modelo Computacional para a obtenção de esforços e deslocamentos placa espessa

Calculo da placa circular espessa

```
> restart : with(plots) : with(LinearAlgebra) : with(linalg) :
> alias(gamma = `γ`) : #Para Liberar gamma
```

Polinômio adicional

> $Eq4Par := \left(1 - \left(\frac{(2 \cdot x)}{L} - 1\right)^{2 \cdot n}\right):$

> $Eq4Impar := \left(\frac{(2 \cdot x)}{L} - 1\right) \cdot \left(1 - \left(\frac{(2 \cdot x)}{L} - 1\right)^{(2 \cdot n)}\right):$

> $Eq4Par := collect(subs(L = a - b, x = r - b, Eq4Par), \{r\}):$

> $Eq4Impar := simplify(collect(subs(L = a - b, x = r - b, Eq4Impar), \{r\})):$

Funções Básicas $N_{pb} = 6$

Numero de funções básicas

> $N_{pb} := 6:$

Polinômios

> $interface(rtablesize = 100):$

> $FuncBasicas := Matrix(N_{pb}, 1, 0):$

> $FuncBasicas_{1,1} := \left(1 - \frac{x}{L}\right):$

> $FuncBasicas_{2,1} := \left(1 - \frac{x}{L}\right):$

> $FuncBasicas_{3,1} := \left(1 - \frac{x}{L}\right):$

> $FuncBasicas_{4,1} := \frac{x}{L}:$

> $FuncBasicas_{5,1} := \frac{x}{L}:$

> $FuncBasicas_{6,1} := \frac{x}{L}:$

> **for** i **from** 1 **to** N_{pb} **do**

> $FuncBasicas_{i,1} := collect(subs(L = a - b, x = r - b, FuncBasicas_{i,1}), \{r\}):$

> **end do:**

Numero de funções adicionais

Numero de funções adicionais em w (Nw=Só pode numero par, contem polinômio par e impar)

> $Nw := 6:$

Numero de funções adicionais em r

> $N\varphi r := 6:$

Numero de funções adicionais em θ

> $N\varphi\theta := 6:$

Numero de funções trigonométricas adicionais circunferenciais

> $N\theta := 0:$

Numero total de polinômios adicionais radiais

> $NR := Nw + N\varphi r + N\varphi\theta:$

Deslocamentos em z e giros , Giros em r e em θ

Vetor de deslocamentos

> $u_z := w:$

> $u_r := \varphi_\theta \cdot z:$

> $u_\theta := -\varphi_r \cdot z:$

Deformações

Deformação em r

> $\varepsilon_{rr} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\varepsilon_{rr})_{i,1} := \frac{\partial}{\partial r}(u_r)_{i,1}$

> **end do:**

> $\varepsilon_{rr}:$

Deformação em θ

> $\varepsilon_{\theta\theta} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\varepsilon_{\theta\theta})_{i,1} := \frac{1}{r}\cdot\left(\left(\frac{\partial}{\partial\theta}(u_\theta)_{i,1}\right)+\left((u_r)_{i,1}\right)\right)$

> **end do:**

> $\varepsilon_{\theta\theta}:$

Deformação cisalhante rz

> $\gamma_{rz} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\gamma_{rz})_{i,1} := \left(\frac{\partial}{\partial z}(u_r)_{i,1}\right)+\left(\frac{\partial}{\partial r}(u_z)_{i,1}\right)$

> **end do:**

> $\gamma_{rz}:$

Deformação cisalhante θz

> $\gamma_{\theta z} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\gamma_{\theta z})_{i,1} := \left(\frac{1}{r}\cdot\frac{\partial}{\partial\theta}(u_z)_{i,1}\right)+\left(\frac{\partial}{\partial z}(u_\theta)_{i,1}\right)$

> end do:

> $\gamma_{\theta z}$:

Deformação cisalhante rθ

> $\gamma_{r\theta} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $(\gamma_{r\theta})_{i,1} := \left(\frac{\partial}{\partial r}(u_\theta)_{i,1}\right) + \left(\frac{1}{r}\cdot\frac{\partial}{\partial\theta}(u_r)_{i,1}\right) - \left(\frac{(u_\theta)_{i,1}}{r}\right)$

> end do:

> $\gamma_{r\theta}$:

Introdução de dados iniciais

Características geométricas do elemento (Dimensões da placa circular) [L]

> $b := 3 : a := 9 : h0 := 0.2 : h1 := 0.6$:

Densidade de massa [M/L^3]

> $\rho := 0$:

Função da variação de espessura "h"

> $h := \left(\frac{(h1-h0)}{(a-b)}\cdot(r-b)\right) + h0$:

Características do material

Coeficiente de Poisson

> $\nu := 0.3$:

Modulo de Elasticidade Longitudinal do material [F/L]^2

> $E := 2\cdot 10^{10}$:

Modulo de Elasticidad transversal do material [F/L]^2

> $G := \frac{E}{2\cdot(1+\nu)}$:

Rigidez a flexão da placa

> $E'' := \frac{E}{(1-\nu^2)}$:

Caracteristicas do Suelo [F/ L]

Coeficiente de rigidez

> $kr := 0$:

Fator de correção do esforço cortante (Reddy)

> $FR := evalf\left(\sqrt{\frac{\pi^2}{12}}\right)$:

Condições das cargas externas

Função do carregamento

> $qz := qo + \frac{(q1 - qo)}{(a - b)} \cdot (r - b) + po \cdot \left(\frac{r}{a}\right) \cdot \sin(N\theta \cdot \theta)$:

Valor da carga linear no bordo interno [F/L]

> $qo := -50000$:

Valor da carga linear no bordo externo [F/L]

> $q1 := -50000$:

Carregamento de variação linear [F/L]

> $po := 0$:

Carregamento Puntual[F]

> $Pz := -\frac{0}{2 \cdot \pi \cdot b}$:

Condições de apoio

> $mola_1 := 0 \cdot 10^{15}$:

> $mola_2 := 1 \cdot 10^{15}$:

> $mola_3 := 1 \cdot 10^{15}$:

> $mola_4 := 1 \cdot 10^{15}$:

> $mola_5 := 1 \cdot 10^{15}$:

> $mola_6 := 1 \cdot 10^{15}$:

Calculo das integrais

Primera Integral

> $\varepsilon rrT := \text{Transpose}(\varepsilon_{rr})$:

> $Br := Multiply(\varepsilon_{rr}, \varepsilon rrT)$:

> $Dimension(Br)$:

> $Kr := Matrix(((N\theta + 1) \cdot (N_{pb} + (2 \cdot NR))), ((N\theta + 1) \cdot (N_{pb} + (2 \cdot NR))), 0)$:

> **for** i **from** 1 **to** $((N\theta + 1) \cdot (N_{pb} + (2 \cdot NR)))$ **do**

> **for** j **from** 1 **to** $((N\theta + 1) \cdot (N_{pb} + (2 \cdot NR)))$ **do**

> $Kr_{i,j} := evalf\left(E'' \cdot \int_b^a \int_0^{2\pi} \int_{-\frac{h}{2}}^{\frac{h}{2}} Br_{i,j} \cdot r \, dz \, d\theta \, dr\right)$

> end do
> end do:
> Kr:

Segunda integral

> $\varepsilon\theta\theta T := \mathrm{Transpose}(\varepsilon_{\theta\theta})$:
> $B\theta := Multiply(\varepsilon_{\theta\theta}, \varepsilon\theta\theta T)$:
> $K\theta := Matrix\big(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0\big)$:
> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> **for** j **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> $$K\theta_{i,j} := evalf\left(E''\cdot\int_b^a\int_0^{2\pi}\int_{-\frac{h}{2}}^{\frac{h}{2}} B\theta_{i,j}\cdot r\,\mathrm{d}z\,\mathrm{d}\theta\,\mathrm{d}r\right)$$
> end do
> end do:
> $K\theta$:

Terceira Integral

> $B\theta r := Multiply(\varepsilon_{\theta\theta}, \varepsilon rrT)$:
> $K\theta r := Matrix\big(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0\big)$:
> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> **for** j **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> $$K\theta r_{i,j} := evalf\left(\nu\cdot E''\cdot\int_b^a\int_0^{2\pi}\int_{-\frac{h}{2}}^{\frac{h}{2}} B\theta r_{i,j}\cdot r\,\mathrm{d}z\,\mathrm{d}\theta\,\mathrm{d}r\right)$$
> end do
> end do:
> $K\theta r$:

Quarta Integral

> $Br\theta := Multiply(\varepsilon_{rr}, \varepsilon\theta\theta T)$:
> $Kr\theta := Matrix\big(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0\big)$:
> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> **for** j **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**
> $$Kr\theta_{i,j} := evalf\left(\nu\cdot E''\cdot\int_b^a\int_0^{2\pi}\int_{-\frac{h}{2}}^{\frac{h}{2}} Br\theta_{i,j}\cdot r\,\mathrm{d}z\,\mathrm{d}\theta\,\mathrm{d}r\right)$$
> end do

```
> end do:
> Krθ:
```

Quinta Integral

```
> γrzT := Transpose(γ_rz) :
> Brz := Multiply(γ_rz, γrzT) :
> Krz := Matrix(((Nθ+1)·(N_pb + (2·NR))), ((Nθ+1)·(N_pb + (2·NR))), 0) :
> for i from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
> for j from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
```

$$Krz_{i,j} := evalf\left(FR^2 \cdot G \cdot \int_b^a \int_0^{2\pi} \int_{-\frac{h}{2}}^{\frac{h}{2}} Brz_{i,j} \cdot r \, dz \, d\theta \, dr \right)$$

```
> end do
> end do:
> Krz:
```

Sexta Integral

```
> γθzT := Transpose(γ_θz) :
> Bθz := Multiply(γ_θz, γθzT) :
> Kθz := Matrix(((Nθ+1)·(N_pb + (2·NR))), ((Nθ+1)·(N_pb + (2·NR))), 0) :
> for i from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
> for j from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
```

$$K\theta z_{i,j} := evalf\left(FR^2 \cdot G \cdot \int_b^a \int_0^{2\pi} \int_{-\frac{h}{2}}^{\frac{h}{2}} B\theta z_{i,j} \cdot r \, dz \, d\theta \, dr \right)$$

```
> end do
> end do:
```

Sétima Integral

```
> γrθT := Transpose(γ_rθ) :
> Bcrθ := Multiply(γ_rθ, γrθT) :
> Kcrθ := Matrix(((Nθ+1)·(N_pb + (2·NR))), ((Nθ+1)·(N_pb + (2·NR))), 0) :
> for i from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
> for j from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
```

$$Kcr\theta_{i,j} := evalf\left(G \cdot \int_b^a \int_0^{2\pi} \int_{-\frac{h}{2}}^{\frac{h}{2}} Bcr\theta_{i,j} \cdot r \, dz \, d\theta \, dr \right)$$

```
> end do
> end do:
> Kcrθ:
```

Vetor de carregamento aplicado

Carregamento distribuído

> $d := (w):$

> $q := Matrix((N\theta+1)\cdot(N_{pb}+(2\cdot NR)), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $q_{i,1} := evalf\left(\int_b^a\int_0^{2\cdot\pi} d_{i,1}\cdot qz\cdot r\,d\theta\,dr\right)$

> **end do:**

Força Puntual

> $P := Matrix((N\theta+1)\cdot(N_{pb}+(2\cdot NR)), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $P_{i,1} := evalf\left(\int_0^b\int_0^{2\cdot\pi} d_{i,1}\cdot Pz\,d\theta\,dr\right)$

> **end do:**

Vetor do carregamento Total

> $VF := (q+P):$

Matriz de rigidez elastica K_E

> $K_{Flex} := Matrix(Kr + K\theta r + Kr\theta + K\theta):$

> $K_{Cisall} := Matrix(Krz + K\theta z + Kcr\theta):$

> $K_E := K_{Flex} + K_{Cisall}:$

Matriz de rigidez de apoio elástico $KAp_{Elastico}$

> $uzT := Transpose(u_z):$

> $Buz := Multiply(u_z, uzT):$

> $KAp_{Elastico} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> **for** j **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(KAp_{Elastico})_{i,j} := evalf\left(kr\cdot\int_b^a\int_0^{2\pi} Buz_{i,j}\cdot r\,d\theta\,dr\right)$

> **end do**

```
> end do:
> KAp_Elastico :
```

Matriz de Apoios

```
> K_Apoyo := Matrix( ((Nθ+1)·(N_pb + (2·NR))), ((Nθ+1)·(N_pb + (2·NR))), 0) :
> if NR > 0 and Nθ=0 then #⇒ Primera condicion
> K_mola := Matrix( (Nθ+1)·(N_pb + (2·NR)), 1, 0) :
> (K_mola)_{1,1} := mola_1 : (K_mola)_{2,1} := mola_2 : (K_mola)_{3,1} := mola_3 : (K_mola)_{4,1} := mola_4 : (K_mola)_{5,1}
     := mola_5 : (K_mola)_{6,1} := mola_6 :
> for n from 1 to (2·NR) do
> (K_mola)_{N_pb + n, 1} := 0
> end do:
> elif NR > 0 and Nθ > 0 then  #⇒ Segunda condicion
> K_mola := Matrix( (N_pb + (2·NR)), 1, 0) :
> (K_mola)_{1,1} := mola_1 : (K_mola)_{2,1} := mola_2 : (K_mola)_{3,1} := mola_3 : (K_mola)_{4,1} := mola_4 : (K_mola)_{5,1}
     := mola_5 : (K_mola)_{6,1} := mola_6 :
> for n from 1 to (2·NR) do
> (K_mola)_{N_pb + n, 1} := 0
> end do:
> K_molaC := Matrix(Nθ·(N_pb + (2·NR)), 1, 0) :
> tot := 0 :
> for k from 1 by 1 while k ≤ Nθ do
> for i from 1 to N_pb + (2·NR) do
> (K_molaC)_{i + tot, 1} := (K_mola)_{i, 1} ·1
> end do
> tot := (N_pb + (2·NR)) ·k
> end do:
> K_mola := ⟨K_mola, K_molaC⟩
> end if:
> for i from 1 to (N_pb + (2·NR))·(Nθ+1) do
> (K_Apoyo)_{i,i} := (K_mola)_{i,1}
> end do:
> K_Apoyo :
```

Obtenção dos deslocamentos generalizados

```
> K_Elastica := evalm(K_E + K_Apoyo + KAp_Elastico) :
> #K_Elastica := subs(r = b, evalm(K_Elastica)) :
> #det(K_Elastica) :
> KInv := inverse(evalm(K_Elastica)) :
> Cn := evalm(KInv.VF) :
```

Deformada

```
> d := (w) :
> Deformada := expand(Transpose(d).Cn) :
> plot3d([r, theta, Deformada[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, axesfont = ["ROMAN",
      22]) :
> contourplot([r, theta, Deformada[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
> evalf(subs(θ = π/2, r = b, Deformada[1, 1])) :
```

Momento Mr

```
> Mr := Matrix((Nθ + 1)·(N[pb] + (2·NR)), 1, 0) :
> for i from 1 to ((Nθ + 1)·(N[pb] + (2·NR))) do
> Mr[i, 1] := evalf(E''·∫_{-h/2}^{h/2} (((ε[rr])[i, 1] + ν·(ε[θθ])[i, 1])·z dz)
> end do:
> CT := transpose(Cn) :
> Mrr := evalf(CT.Mr) :
> plot3d([r, theta, Mrr[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, axesfont = ["ROMAN", 22]) :
> contourplot([r, theta, Mrr[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
> evalf(subs(θ = π/2, r = 4.5, Mrr[1, 1])) :
```

Momento Mθ

```
> Mθ := Matrix((Nθ + 1)·(N[pb] + (2·NR)), 1, 0) :
> for i from 1 to ((Nθ + 1)·(N[pb] + (2·NR))) do
> Mθ[i, 1] := evalf(E''·∫_{-h/2}^{h/2} (((ε[θθ])[i, 1] + ν·(ε[rr])[i, 1])·z dz)
> end do:
> Mθθ := evalf(CT.Mθ) :
> plot3d([r, theta, Mθθ[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, coords = cylindrical, axesfont
      = ["ROMAN", 22]) :
> contourplot([r, theta, Mθθ[1, 1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
> evalf(subs(θ = π/2, r = 4.5, Mθθ[1, 1])) :
```

Cortante Qr

```
> Qr := Matrix((Nθ + 1)·(N[pb] + (2·NR)), 1, 0) :
```

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $Qr_{i,1} := FR^2\cdot G\cdot\int_{-\frac{h}{2}}^{\frac{h}{2}} (\gamma_{rz})_{i,1}\,dz$

> end do:

> $Qr := evalf(CT.Qr)$:

> $plot3d([r, \text{theta}, (Qr)_{1,1}], \text{theta}=0..2*\text{Pi}, r=b..a, coords=cylindrical, axesfont=[\text{"ROMAN"}, 22])$:

> $contourplot([r, \text{theta}, Qr_{1,1}], \text{theta}=0..2*\text{Pi}, r=b..a, coords=cylindrical)$:

> $evalf\left(subs\left(\theta=\frac{\pi}{2}, r=4.5, Qr_{1,1}\right)\right)$:

Tensão σrmax

> $\sigma r := Matrix((N\theta+1)\cdot(N_{pb}+(2\cdot NR)), 1, 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $\sigma r_{i,1} := E''\cdot\left((\varepsilon_{rr})_{i,1}+\nu\cdot(\varepsilon_{\theta\theta})_{i,1}\right)$

> end do:

> $\sigma r := subs\left(z=-\frac{h}{2}, \sigma r\right)$:

> $\sigma rr := evalf(CT.\sigma r)$:

> $plot3d([r, \text{theta}, (\sigma rr)_{1,1}], \text{theta}=0..2*\text{Pi}, r=b..a, coords=cylindrical, axesfont=[\text{"ROMAN"}, 22])$:

> $contourplot([r, \text{theta}, (\sigma rr)_{1,1}], \text{theta}=0..2*\text{Pi}, r=b..a, coords=cylindrical)$:

> $evalf\left(subs\left(\theta=\frac{\pi}{2}, r=b, \sigma rr_{1,1}\right)\right)$:

C. Modelo Computacional para a obtenção de frequências da placa espessa.

Calculo das frequências da placa espessa

```
> restart : with(plots) : with(LinearAlgebra) : with(linalg) :
> alias(gamma = `γ`) :
```

▶ Polinômio adicional

▶ Funções Básicas $N_{pb} = 6$

▶ Numero de funções adicionais

▶ Deslocamentos em z e giros , Giros em r e em θ

▶ Vetor de deslocamentos

▼ Deformações

▼ Rotação relativa respeito do eixo r

> $uz_r := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $(uz_r)_{i,1} := \frac{\partial}{\partial r}(u_z)_{i,1}$

> end do:

> uz_r :

▼ Rotação relativa respeito do eixo θ

> $uz_\theta := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $(uz_\theta)_{i,1} := \frac{1}{r^2}\cdot\frac{\partial}{\partial\theta}(u_z)_{i,1}$

> end do:

> uz_θ :

▼ Deformação em r

> $\varepsilon_{rr} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $(\varepsilon_{rr})_{i,1} := \frac{\partial}{\partial r}(u_r)_{i,1}$

> end do:

> ε_{rr} :

> $Dimension(\varepsilon_{rr})$:

Deformação em θ

> $\varepsilon_{\theta\theta} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\varepsilon_{\theta\theta})_{i,1} := \frac{1}{r}\cdot\left(\left(\frac{\partial}{\partial\theta}(u_\theta)_{i,1}\right)+\left((u_r)_{i,1}\right)\right)$

> **end do**:

> $\varepsilon_{\theta\theta}$:

> $Dimension(\varepsilon_{rr}):$

Deformação cisalhante rz

> $\gamma_{rz} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\gamma_{rz})_{i,1} := \left(\frac{\partial}{\partial z}(u_r)_{i,1}\right)+\left(\frac{\partial}{\partial r}(u_z)_{i,1}\right)$

> **end do**:

> γ_{rz}:

Deformação cisalhante θz

> $\gamma_{\theta z} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\gamma_{\theta z})_{i,1} := \left(\frac{1}{r}\cdot\frac{\partial}{\partial\theta}(u_z)_{i,1}\right)+\left(\frac{\partial}{\partial z}(u_\theta)_{i,1}\right)$

> **end do**:

> $\gamma_{\theta z}$:

Deformação cisalhante rθ

> $\gamma_{r\theta} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 1, 0):$

> **for** i **from** 1 **to** $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ **do**

> $(\gamma_{r\theta})_{i,1} := \left(\frac{\partial}{\partial r}(u_\theta)_{i,1}\right)+\left(\frac{1}{r}\cdot\frac{\partial}{\partial\theta}(u_r)_{i,1}\right)-\left(\frac{(u_\theta)_{i,1}}{r}\right)$

> **end do**:

> $\gamma_{r\theta}$:

▶ Introdução de dados iniciais

▶ Calculo das integrais

▶ Matriz de Rigidez K_E

▶ Matriz de rigidez de apoio elástico $KAp_{Elastico}$

▶ Matriz de Apoios

▼ Matriz de Massa

▼ INERCIA TRANSLACIONAL

> $uzT := Transpose(u_z)$:

> $Bm1 := Multiply(u_z, uzT)$:

> $K1_{Masa} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0)$:

> **for** i **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> **for** j **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> $$(K1_{Masa})_{i,j} := \rho\cdot\int_{-\frac{h}{2}}^{\frac{h}{2}}\int_{b}^{a}\int_{0}^{2\pi} Bm1_{i,j}\cdot r\, d\theta\, dr\, dz$$

> **end do**

> **end do**

> $K1_{Masa}$:

▼ INERCIA ROTACIONAL

▼ *Inercia Rotacional eixo r*

> $urT := Transpose(u_r)$:

> $Bm2 := Multiply(u_r, urT)$:

> $K2_{Masa} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0)$:

> **for** i **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> **for** j **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> $$(K2_{Masa})_{i,j} := \rho\cdot\int_{-\frac{h}{2}}^{\frac{h}{2}}\int_{b}^{a}\int_{0}^{2\pi} Bm2_{i,j}\cdot r\, d\theta\, dr\, dz$$

> **end do**

> **end do**

> $K2_{Masa}$:

▼ *Inercia rotacional eixo θ*

> $u\theta T := Transpose(u_\theta)$:

> $Bm3 := Multiply(u_\theta, u\theta T)$:

> $K3_{Masa} := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0)$:

> **for** i **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> **for** j **from** 1 **to** $(N\theta+1)\cdot(N_{pb}+(2\cdot NR))$ **do**

> $$(K3_{Masa})_{i,j} := \rho\cdot\int_{-\frac{h}{2}}^{\frac{h}{2}}\int_{b}^{a}\int_{0}^{2\pi} Bm3_{i,j}\cdot r\, d\theta\, dr\, dz$$

```
> end do
> end do
> K3_Masa:
> K_Masa := K1_Masa + K2_Masa + K3_Masa:
```

Frequências

```
> K_Elastica := (K_E + K_Apoyo + KAp_Elastico) :
> λ := Eigenvalues(Multiply(MatrixInverse(K_Elastica), K_Masa)) :
> ω := Matrix((N_pb + (2·NR))·(Nθ+1), 1, 0) :
> for i from 1 to (N_pb + (2·NR))·(Nθ+1) do
> ω[i,1] := evalf((1/λ[i])^(1/2))
> end do:
> interface(rtablesize = 150) :
> ω := sort(Column(ω, 1)) : Re(ω) :
> λ := Matrix((Nθ+1)·(N_pb + (2·NR)), 1, 0) :
> for i from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
> λ[i,1] := a^2·sqrt(ρ·h/((E·h^3)/(12·(1 - ν^2))))·ω[i]
> end do:
> Re(λ) :
```

$$\omega_{i,\,1} := evalf\left(\left(\frac{1}{\lambda_i}\right)^{\frac{1}{2}}\right)$$

$$\lambda_{i,\,1} := a^2\cdot\sqrt{\frac{\rho\cdot h}{\left(\frac{E\cdot h^3}{12\cdot(1-\nu^2)}\right)}}\cdot\omega_i$$

Modos de vibração

```
> Evl, Evc := Eigenvectors(Multiply(MatrixInverse(K_Masa), K_Elastica)) :
> s := Re(Evl) :
> sort(s, `<`) :
```

Primeiro modo de vibração

```
> DesG1 := Transpose(Re(Evc(.., 11))) :
> Modo1 := Multiply(DesG1, w) :
> plot3d([r, theta, Modo1[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, axesfont = ["ROMAN",
      22]) :
> contourplot([r, theta, Modo1[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
```

Segundo modo de vibração

```
> DesG2 := Transpose(Re(Evc(.., 5))) :
> Modo2 := Multiply(DesG2, w) :
> plot3d([r, theta, Modo2[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, axesfont = ["ROMAN",
      22]) :
```

```
> contourplot([r, theta, Modo2[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
```

Terceiro modo de vibração

```
> DesG3 := Transpose(Re(Evc(.., 8))) :
> Modo3 := Multiply(DesG3, w) :
> plot3d([r, theta, Modo3[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical, axesfont = ["ROMAN",
    22]) :
> contourplot([r, theta, Modo3[1]], theta = 0..2*Pi, r = b..a, coords = cylindrical) :
```

D. Modelo Computacional para a obtenção da carga crítica de flambagem da placa espessa.

Calculo da carga critica de flambagem da placa circular espessa

```
> restart : with(plots) : with(LinearAlgebra) : with(linalg) :
> alias(gamma = `γ`) :
```

▶ Polinômio adicional

▶ Funções Básicas $N_{pb} = 6$

▶ Numero de funções adicionais

▶ Deslocamentos em z e giros , Giros em r e em θ

▶ Vetor de deslocamentos

▶ Deformações

▶ Introdução de dados iniciais

▶ Calculo das integrais

▶ Matriz de rigidez elastica K_E

▶ Matriz de rigidez de apoio elástico $KAp_{Elastico}$

▼ Matriz Geometrica K_G

▼ Primeira Integral

> uzrT := Transpose(uz_r) :

> $Bzr := Multiply(uz_r, uzrT)$:

> $K\sigma r := Matrix(((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), ((N\theta+1)\cdot(N_{pb}+(2\cdot NR))), 0)$:

> for i from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> for j from 1 to $((N\theta+1)\cdot(N_{pb}+(2\cdot NR)))$ do

> $K\sigma r_{i,j} := evalf\left(\int_{-\frac{h}{2}}^{\frac{h}{2}}\int_{b}^{a}\int_{0}^{2\pi}\sigma_r\cdot Bzr_{i,j}\cdot r\, d\theta\, dr\, dz\right)$

```
> end do
> end do:
> Kσr:
```

Segunda integral

```
> uzθT := Transpose(uz_θ):
> Bzθ := Multiply(uz_θ, uzθT):
> Kσθ := Matrix(((Nθ+1)·(N_pb + (2·NR))), ((Nθ+1)·(N_pb + (2·NR))), 0):
> for i from 1 to ((Nθ+1)·(N_pb + (2·NR))) do
> for j from 1 to ((Nθ+1)·(N_pb + (2·NR)))do
```

$$K\sigma\theta_{i,j} := evalf\left(\int_{-\frac{h}{2}}^{\frac{h}{2}}\int_{b}^{a}\int_{0}^{2\pi}\sigma_\theta \cdot Bz\theta_{i,j}\cdot r\, d\theta\, dr\, dz\right):$$

```
> end do
> end do:
> Kσθ:
```

Rigidez Geometrica

```
> KG := Kσr + Kσθ:
```

Matriz de Apoios

Carga Crítica

$> K_{Elastica} := (K_E + K_{Apoyo} + KAp_{Elastico}):$

$> KG := subs(P=1, KG):$

$> \lambda := \text{Eigenvalues}(Multiply(MatrixInverse(K_{Elastica}), KG)):$

$> Pcr := Vector((N_{pb} + (2\cdot NR))\cdot(N\theta+1), 1, 0):$

$>$ **for** i **from** 1 **to** $(N_{pb} + (2\cdot NR))\cdot(N\theta+1)$ **do**

$$> Pcr_i := evalf\left(-\frac{1}{\lambda_i}\right)$$

```
> end do:
> Po := Re(Pcr):
> interface(rtablesize = 150):
> select(type, Po, negative):
```

E. Modelo Computacional para a obtenção da carga crítica de flambagem dinâmica da placa espessa.

Calculo de flambagem dinâmica da placa espessa

```
> restart : with(plots) : with(LinearAlgebra) : with(linalg) :
> alias(gamma = `γ`) : #Para Liberar gamma
```

▶ **Polinômio adicional**

▶ **Funções Básicas** $N_{pb} = 6$

▶ **Numero de funções adicionais**

▶ **Deslocamentos em z e giros , Giros em r e em θ**

▶ **Vetor de deslocamentos**

▶ **Deformações**

▼ **Introdução de dados iniciais**

▼ **Características geométricas do elemento (Dimensões da placa circular) [L]**

> $b := 2.7 : a := 9 : h0 := 0.8 : h1 := 0.8 : P := -7.73599214393585\ 10^7 : \lambda := 5.5142749 :$

Densidade de massa [M/L^3]

> $\rho := 27000 :$

Função da variação de espessura "h"

> $h := \left(\frac{(h1 - h0)}{(a - b)} \cdot (r - b)\right) + h0 :$

▶ **Características do material**

▶ **Condições das cargas externas**

▶ **Condições de apoio**

▶ **Calculo das integrais**

▶ **Matriz de rigidez elastica** K_E

▶ **Matriz de rigidez de apoio elástico** $KAp_{Elastico}$

▶ Matriz de Apoios

▶ Matriz de Massa

▶ Matriz Geometrica K_G

▼ Matriz de Carga Seguidora K_L

> $KL := Matrix\left((N\theta+1)\cdot\left(N_{pb}+(2\cdot NR)\right), (N\theta+1)\cdot\left(N_{pb}+(2\cdot NR)\right), 0\right):$

> $(KL)_{4,5} := evalf(-2\cdot\pi\cdot a\cdot P) : (KL)_{4,6} := evalf(2\cdot\pi\cdot a\cdot P):$

> $KL:$

▼ Frequência de Vibração

> $KG := (KG+KL):$

> $KGL := (\lambda\cdot KG):$

> $K_{Elastica} := \left(K_E + K_{Apoyo} + KAp_{Elastico}\right):$

> $A := \left(KGL + K_{Elastica}\right):$

> $\Omega := \mathrm{Eigenvalues}\left(Multiply\left(MatrixInverse(A), K_{Masa}\right)\right):$

> $\omega := Matrix\left(\left(N_{pb}+(2\cdot NR)\right)\cdot(N\theta+1), 1, 0\right):$

> **for** i **from** 1 **to** $\left(N_{pb}+(2\cdot NR)\right)\cdot(N\theta+1)$ **do**

> $\omega_{i,1} := evalf\left(\left(\frac{1}{\Omega_i}\right)^{\frac{1}{2}}\right)$

> **end do**:

> $interface(rtablesize = 150):$

> $\omega := sort(Column(\omega, 1)):$

> $\lambda := Matrix\left((N\theta+1)\cdot\left(N_{pb}+(2\cdot NR)\right), 1, 0\right):$

> **for** i **from** 1 **to** $\left((N\theta+1)\cdot\left(N_{pb}+(2\cdot NR)\right)\right)$ **do**

> $\lambda_{i,1} := a^2\cdot\sqrt{\frac{\rho\cdot h}{\left(\frac{E\cdot h^3}{12\cdot(1-\nu^2)}\right)}}\cdot\omega_i$

> **end do**:

> $\lambda:$