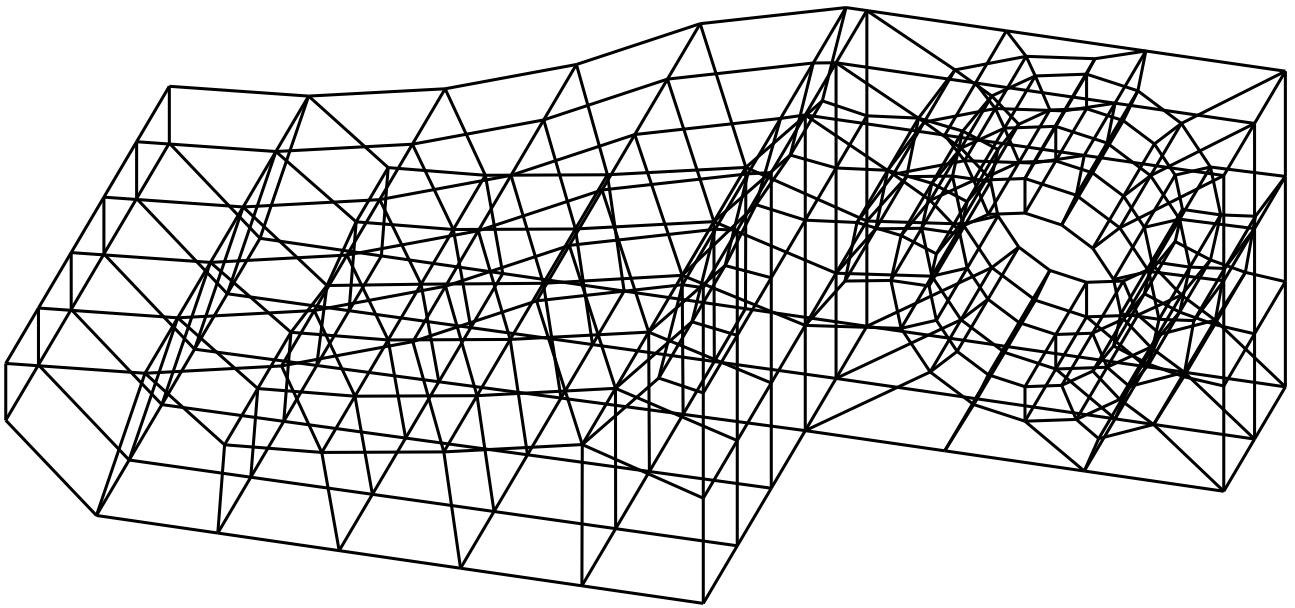


Instituto Tecnológico de Buenos Aires

# Elementos Finitos

UN APUNTE INFORMAL



PATRICIO WHITTINGSLOW

# Índice

<b>0. Introducción al Método de los Elementos Finitos</b>	<b>1</b>
<b>I Programación del método de los elementos finitos</b>	<b>2</b>
1. Mallado	2
2. Mapeo de dof	3
3. Acople de rigidez	3
4. Aplicación de condiciones de borde esenciales	4
5. Aplicación de condiciones de borde naturales	4
6. Procedimiento de resolución	4
<b>II Elementos 1D</b>	<b>5</b>
7. Elemento barra	5
8. Viga 3D de Timoshenko	6
9. Orientación de elementos 1D	8
10.Ejercicios	9
<b>III Elementos 2D</b>	<b>10</b>
11.Protips para el mallado	10
12.Funciones de formas para elementos 2D	12
13.Elementos isoparamétricos	12
14.Elementos Axisimetricos	14
<b>IV Otras cosas</b>	<b>14</b>
15.Desplazamientos iniciales	14
16.Transferencia de Calor	15
17.Formulación elemento Hexaedro de 8 nodos	15
<b>V No-linealidad y análisis dinámico</b>	<b>17</b>
18.Respuesta dinámica estructural	17
19.Transferencia de calor no-lineal y transitoria	18
20.Análisis Dinámico	18
<b>VI Anexo</b>	<b>19</b>

## 0. Introducción al Método de los Elementos Finitos

El análisis o método de elementos finitos (FEA por *finite element analysis*) es usado para obtener una solución numérica de un problema de campo (electrostático, térmico, tensiones). Matemáticamente estos problemas están definidos como ecuaciones diferenciales o como un integral. Ambas expresiones pueden formularse con elementos finitos.

Las ventajas de FEA son

- Es aplicable a cualquier problema de campo
- No hay restricciones geométricas
- No hay restricciones al tipo de cargas o condiciones de borde que se pueden aplicar
- Se puede formular para materiales que no son isotrópicos e incluso el tipo de material puede cambiar dentro de un elemento
- Se pueden combinar distintos tipos de elementos en un modelos, por ejemplo, unir barras con vigas o incluso con elementos 3D.
- La aproximación se puede mejorar fácilmente refinando la malla donde hay gradientes de tensión altos

### Proceso de resolución

El primer paso es identificar y **clasificar** el problema.

- Cuales son los fenómenos físicos involucrados y que resultados se buscan del análisis
- Depende del tiempo? (estático o dinámico)
- Hace falta una resolución iterativa? (no linealidad: radiación, plasticidad)

Luego se comienza el **modelado** del problema.

- Se excluyen los detalles superfluos, dejando lo esencial para describir el problema con un margen de error adecuado sin complicar las cosas innecesariamente
- Un *modelo geométrico* se convierte en un *modelo matemático* cuando se describe su comportamiento mediante ecuaciones diferenciales y condiciones de borde.\*

Un modelo matemático es una idealización donde se simplifican la geometría, propiedades del material, cargas y/o condiciones de borde en base del entendimiento del analista acerca lo que tiene (o no) importancia al momento de obtener los resultados requeridos.

Finalmente llega el momento de la **discretización**. Un modelo matemático se discretiza dividiéndolo en una malla de elementos finitos. De esta forma, un campo continuo es representado como una función partida la cual es definida por una cantidad finita de variables nodales e interpolación dentro de cada elemento.

### Tipos de error

Al momento de discretizar se introduce error conocido como **error de discretización**. Eso sucede porque se aproxima un campo *suave* con una función partida. Aumentar el número de elementos puede disminuir este error pero nunca eliminarlo.

Aún reduciendo el error de discretización se tendría **error numérico** porque toda computadora usa números de finita precisión para efectuar aritmética. Este error suele ser mínimo cuando se discretiza de forma adecuada y no se tiene una situación física propensa al *mal condicionamiento*.

Cabe destacar que se introdujo error antes de hacer una sola cuenta! El **error de modelado** se introduce por necesidad de simplificar el problema. Las cargas puntuales, los soportes fijos y los materiales perfectamente homogéneos no existen en la realidad! [Cook et al. \[2007\]](#).

---

\*Un modelo de FEA no es la realidad, es una *simulación*. Dificilmente se obtengan resultados buenos cuando se aplique FEA a un modelo matemático que no refleja la realidad de forma apropiada.

## Principios básicos de elementos finitos

La ecuación que se resuelve es

$$M\ddot{d} + C\dot{d} + Kd = F^{\text{externas}}$$

para un sistema mecánico. Es común tratar problemas estáticos tener como variables de entrada las fuerzas externas  $F^{\text{externas}}$  (peso propio, fuerzas aplicadas, fuerza centrífuga etc.) y la rigidez del sistema  $K$ . Los desplazamientos  $d$  sería la variable que se desea obtener. La ecuación a resolver entonces es

$$d = K^{-1} \cdot F^{\text{externas}}$$

El método de los elementos finitos entonces tiene su campo de rigidez  $K$  que se suele llamar la matriz de rigidez global  $[K]$ . Esta asocia rigidez con los grados de libertad de los nodos obtenidos de la discretización. Para cuerpos sólidos en el espacio hay 6 grados de libertad, 3 de desplazamiento ( $u, v, w$ ) y tres de giro ( $\varphi, \theta, \psi$ ).

Antes de discretizar un modelo se eligen las direcciones  $x, y, z$  globales. Los desplazamientos obtenidos corresponderán a estas direcciones.

## Parte I

# Programación del método de los elementos finitos

Objeto matemático	Nombre de variable	Definición
$N_{\text{nod.}}$	Nnod	Número de nodos
$N_{\text{elem.}}$	Nelem	Número de elementos
$\vartheta_{\text{nod.}}$	Ndofpornod	Número dof por nodo
$N_{\text{dof}} = \vartheta_{\text{nod.}} \cdot N_{\text{nod.}}$	Ndof o simplemente dof	Número de dof
$n_{\text{nod.}}$	Nnodporelem	Número de nodos por elemento
$n_{\text{dof}} = \vartheta_{\text{nod.}} \cdot n_{\text{nod.}}$	Ndofporelem	Número de dof por elemento
$N_{\text{dim.}}$	Ndim	Número de dimensiones
$[k]$	ke	Matriz de rigidez del elemento
$[K]$	K	Matriz de rigidez global
$\{R\}$	R	Vector de cargas
$\{D\}$	D	Vector de desplazamientos

Tabla 1: Parámetros importantes para la resolución de elementos finitos. Comúnmente denominados *definitions*.

Se sugiere al lector usar los nombres de variables mencionados en la tabla 1. La motivación de los nombres de variable es que sean legibles y cortos a la vez ya que aparecen seguido en un código de elementos finitos. 'N' se lee Número y 'elem' se lee como elementos etc. Por ejemplo, Ndofpornod se leería: Número de dof por nodo.

## 1. Mallado

Primero se requiere discretizar el problema. Se comienza con una matriz de nodos que va unir los elementos.

$$\text{nodos} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 6.0 & 0 \\ 6.0 & 8.0 & 0 \\ 12.0 & 6.0 & 0 \\ 12.0 & 0 & 0 \end{bmatrix}_{N_{\text{nod.}} \times N_{\text{dim.}}} \quad (1)$$

Para acceder a las coordenadas del nodo  $n$ -ésimo en MATLAB: `nodos(n, :)`

La matriz de conectividad de los elementos:

$$\text{elementos} = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 5 & 4 \\ 4 & 3 \end{bmatrix}_{N_{\text{elem.}} \times n_{\text{nod.}}} \quad (2)$$

El orden de la numeración es lo que orienta al elemento. Tener especial cuidado cuando se trabaje con elementos 2D y 3D, numerar los nodos contraria a la formulación escogida puede causar problemas de jacobianos negativos!

Con estas dos matrices se ha definido una malla que podría resolver el caso de la figura 3b.\*

Las matrices nodos y elementos definen algunos parámetros importantes (código MATLAB):

```
[Nnod, Ndim] = size(nodos);
[Nelem, Nnodporelem] = size(elementos);
```

## 2. Mapeo de dof

Para resolver el problema de elementos finitos hay que asociar dof a los elementos. Definido abajo la **función de mapeo** nodo a dof. Se ingresa el nodo de interés  $n$  y la función devuelve los dof asociados a ese nodo.

$$\text{node2dof}(n) = \begin{bmatrix} n \cdot \vartheta_{\text{nod.}} - \vartheta_{\text{nod.}} + 1 \\ n \cdot \vartheta_{\text{nod.}} - \vartheta_{\text{nod.}} + 2 \\ \vdots \\ n \cdot \vartheta_{\text{nod.}} \end{bmatrix}_{\vartheta_{\text{nod.}} \times 1} \quad (3)$$

Se puede entonces obtener la matriz **dof asociados a los elementos**, `elemndof`. En MATLAB:

```
aux = node2dof(reshape(elementos', [], 1)');
elemndof = reshape(aux, size(elemndof'))'
```

La función `reshape(X, m, n)` devuelve una matriz de  $m \times n$  con los elementos de la matriz  $X$  recorriendo las columnas de arriba para abajo.

Considerando  $\vartheta_{\text{nod.}} = 3$  para el ejemplo que se viene tratando se tiene:

$$\text{elemndof} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 & 8 & 9 \\ 13 & 14 & 15 & 10 & 11 & 12 \\ 10 & 11 & 12 & 7 & 8 & 9 \end{bmatrix}_{N_{\text{elem.}} \times n_{\text{dof}}} \quad (4)$$

## 3. Acople de rigidez

Se itera sobre los elementos, obteniendo la rigidez del elemento de alguna forma, sea cual sea (integración directa, cuadratura, matriz predefinida, etc). Una vez obtenida se acopla  $[\mathbf{k}]$  al sistema usando la matriz dof asociados a los elementos para obtener  $[\mathbf{K}]_{N_{\text{dof}} \times N_{\text{dof}}}$

```
% Funcionalmente equivalente a hacer: K = zeros(Ndof);
K = sparse(Ndof, Ndof);
for e = 1:Nelem
    meindof = elemndof(e,:);
    ke = int(B'*E*B,0,L);
    K(meindof, meindof) = K(meindof, meindof) + ke;
end
```

---

\*Para problemas de vigas 3D se tienen 6 dof por nodo, osea  $N_{\text{dof}} = 6$ ;

donde `sparse(m,n)` crea una matriz *sparse* de tamaño  $m \times n$ .<sup>\*</sup> Almacenar la matriz en forma *sparse* no solo ahorra espacio en memoria pero también acorta tiempos de resolución. Es recomendable su uso cuando  $N_{\text{dof}} > 10^4$ .

Si el lector desea profundizar su conocimiento, se lo dirige a leer [Chessa \[2002\]](#).

## 4. Aplicación de condiciones de borde esenciales

La ecuación diferencial que se resuelve mediante el método de elementos finitos tiene la forma

$$L\mathbf{u} + \mathbf{q} = 0$$

donde  $\mathbf{u}$  es el vector de variables primario,  $L$  es el operador diferencial y  $\mathbf{q}$  es el vector de funciones conocidas. Una ecuación diferencial se le pueden aplicar **condiciones de borde naturales** (del tipo Neumann) y **condiciones de borde esenciales** (del tipo Dirichlet) [[Dixit, 2007](#)]

**Esenciales** Es necesario por lo menos una condición de borde esencial para la resolución *completa* del problema. Se imponen a la variable primaria  $\mathbf{u}$  (impuesto a los desplazamientos)

**Naturales** Son las condiciones de borde que involucran términos derivados de orden superior y no son suficientes por si solas para la resolución del problema. Se imponen a la variable secundaria  $\mathbf{q}$  (fuerzas, tracciones etc.)

La condición de borde esencial más simple de aplicar al momento de programar es igualar el desplazamientos a 0. En MATLAB se puede trabajar este concepto usando matrices lógicas con relativa facilidad

```
isFixed = false(Ndof,1); % Crea un vector columna boolean
isFixed(node2dof(n)) = true; % Se 'empotra' el nodo n
isFixed([14 2]) = true; % Restringo los dof 2 y 14
isFree = ~isFixed; % es común llamarlos 'libre' y 'fijo' en castellano
```

En la sección [15](#) se verá como imponer desplazamientos no nulos.

Siguiendo el ejemplo, como aplicaría los empotramientos *A* y *B* de la figura [3b](#)? Y si uno fuera un apoyo simple, como haría?

## 5. Aplicación de condiciones de borde naturales

Ya sabemos lo que es una condición de borde natural de la última sección, queda aplicar dicho conocimiento. Llegado a este punto se remarca la importancia de usar **unidades consistentes**. El autor sugiere usar newtons, metros y pascuales.

```
R = zeros(Ndof,1);
R([1 2 15]) = 9e3; % Cargo mis dof 1 2 y 15 con 9000 unidades
```

## 6. Procedimiento de resolución

El sistema a resolver es  $\{\mathbf{D}\} = [\mathbf{K}]^{-1}\{\mathbf{R}\}$  aunque no es económico invertir la matriz  $[\mathbf{K}]$  desde un punto de vista numérico: La matriz  $[\mathbf{K}]$  es esparsa y su inversa es una matriz *full*.<sup>†</sup> Hay varias formas de resolver el problema que no requieren de la inversa, MATLAB tiene incorporado el operador `mldivide(A,B)` que resuelve un sistema de ecuaciones lineal y devuelve  $\mathbf{x}$  tal que  $A\mathbf{x} = B$ .

```
Dr = K(isFree,isFree)\R(isFree); % == mldivide(K(isFree,isFree),R(isFree));
```

---

<sup>\*</sup>Una matriz esparsa es una matriz cuyos elementos son, en la gran mayoría, igual a cero.

<sup>†</sup>Una matriz full no tiene elementos que son cero.

Se suele llamar a la matriz con condiciones de borde aplicadas la **matriz de rigidez reducida**. En el caso que el sistema sea singular (no hay suficientes condiciones de borde esenciales para hallar una solución), o que el problema esté mal condicionado, `mldivide` devuelve un aviso.

En el código de arriba se calculó  $D_r$  con las matrices de carga y rigidez reducidas, por ende  $D_r$  tiene tamaño  $N_{dof} - N_{cb}$  donde  $N_{cb}$  son la cantidad de dof restringidos.\* Es conveniente obtener el vector de desplazamientos global para el pos-procesado:

```
D = zeros(Ndof,1);
D(isFree) = Dr;
```

Como nadie nació leyendo vectores columnas se pueden reorganizar los desplazamientos en una matriz que muestre los desplazamientos de cada nodo en sus filas. También se puede obtener la posición deformada de los nodos sumando las matrices.

```
desplazamientos = reshape(D, [], Ndofpornod)';
% si los primeros 3 dof son u,v,w puedo sumar:
posiciondeformada = nodos + mag*desplazamientos(:,1:3);
```

donde  $mag$  es una variable para amplificar las deformaciones y que se puedan ver en un gráfico. Suele estar entre 30-100 para problemas con vigas.

Aún se desconocen las reacciones del problema. Se pueden obtener todas las fuerzas externas con:

```
R = K*D;
R(node2dof(n)) % Puedo visualizar reacciones en nodo n
```

$R$  ahora tendrá las reacciones en los dof correspondientes.

## Parte II

# Elementos 1D

## 7. Elemento barra

Uno puede pensar el elemento barra como un resorte que solo ejerce fuerza en la dirección en que apunta su eje local  $x'$ .

$$[k']_{\text{barra}} = \begin{bmatrix} X & -X \\ -X & X \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad \text{donde} \quad X = \frac{EA}{L} \quad (5)$$

si se acopla la matriz a  $[K]$  así como está solo otorga rigidez en la dirección  $x$  global. Si se quiere modelar una barra en el plano  $x$  y se tiene que rotar la barra según su orientación. Si el modelo es plano, o sea solo se modelan  $x$  e  $y$  global, se puede rotar la barra como será visto en la sección 9.

Para orientar una barra que se encuentra en el espacio se puede usar la matriz de rotación para una viga 3D con una leve modificación a la matriz de rigidez de la barra. Esta tiene que incluir todos los grados de libertad del problema! Es decir, la matriz  $[k']_{\text{barra}}$  termina siendo de  $12 \times 12$  para un problema de 6 grados de libertad por nodo.

$$k_{1,1} = k_{7,7} = X, \quad k_{7,1} = k_{1,7} = -X, \quad \text{las demás: } k_{i,j} = 0$$

Un lector mosca ser dará cuenta que la matriz de rigidez de la viga Timoshenko 3-D (expresión 7) es la generalización para todo elemento 1D: barras e incluso vigas en el plano.<sup>†</sup>

### Bar patch test

Llegado a este punto en la lectura, el autor se imagina que el lector debe estar ansioso por poner a prueba sus conocimientos. La figura 1 describe un problema estático que se puede resolver con una breve análisis a mano alzada. Este tipo de problema se denomina *patch test* porque sirve para ensayar la calidad del elemento y método de resolución.

\*A pesar de no tener mucho uso en un programa de elementos finitos se deja la definición:  $N_{cb} = \text{sum}(\text{isFixed})$

<sup>†</sup>Las formulaciones de vigas en el plano más comunes son de dos  $(v, \theta)$  y tres  $(u, v, \theta)$  grados de libertad por nodo.

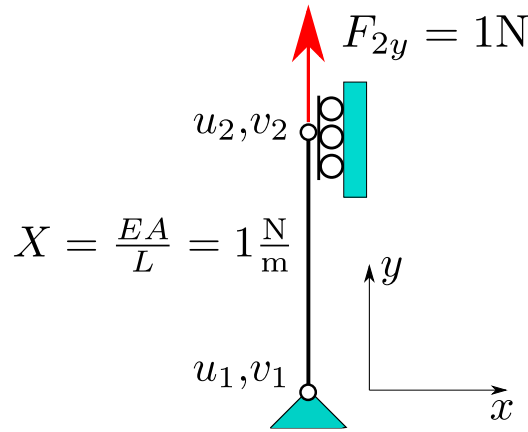


Figura 1: Un *bar patch test* para verificar un *solver* de elementos finitos.  $v_2 = 1\text{m}$ .

El *bar patch test* se puede mallar como un problema plano en el plano  $xy$  con dos nodos ( $x_1 = y_1 = x_2 = 0$ ,  $y_2 = 1\text{m}$ ) y un elemento que los une. La rigidez del material entonces seria  $EA = 1\text{N}$ . Se aplican las condiciones de borde detalladas en la figura: se restringen  $u_1, v_1$  y  $u_2$  y se resuelve el sistema para obtener  $v_2$ . Con este patch test se verifica el método de resolución, la forma en que se rota la matriz de rigidez de la barra, la aplicación de cargas y el elemento en si.

## 8. Viga 3D de Timoshenko

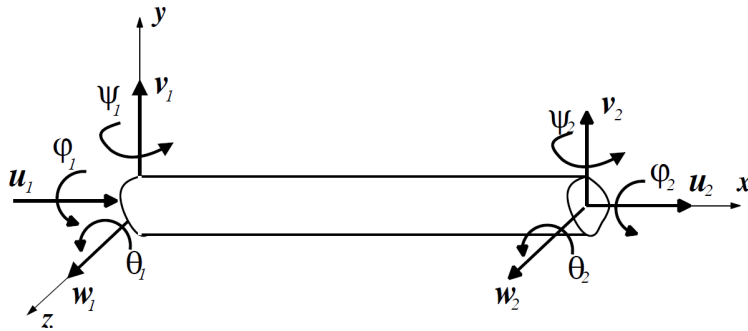


Figura 2: Grados de libertad (*dof*) y coordenadas locales de una viga 3D de dos nodos y 6 dof por nodo.

La viga de Timoshenko 3D tiene las siguientes funciones de forma (forma eficiente tomada de [Luo](#)



[2008]]

$$\left\{ \begin{array}{l} N_1 = 1 - \xi \\ N_2 = \xi \\ H_{v_1} = \beta_y (2\xi^3 - 3\xi^2 + \alpha_y \xi + 1 - \alpha_y) \\ H_{v_2} = \beta_y (-2\xi^3 + 3\xi^2 - \alpha_y \xi) \\ H_{w_1} = \beta_z (2\xi^3 - 3\xi^2 + \alpha_z \xi + 1 - \alpha_z) \\ H_{w_2} = \beta_z (-2\xi^3 + 3\xi^2 - \alpha_z \xi) \\ H_{\theta_1} = L\beta_y [\xi^3 + (\frac{1}{2}\alpha_y - 2)\xi^2 + (1 - \frac{1}{2}\alpha_y)\xi] \\ H_{\theta_2} = L\beta_y [\xi^3 - (1 + \frac{1}{2}\alpha_y)\xi^2 + (\frac{1}{2}\alpha_y)\xi] \\ H_{\psi_1} = L\beta_z [\xi^3 + (\frac{1}{2}\alpha_z - 2)\xi^2 + (1 - \frac{1}{2}\alpha_z)\xi] \\ H_{\psi_2} = L\beta_z [\xi^3 - (1 + \frac{1}{2}\alpha_z)\xi^2 + (\frac{1}{2}\alpha_z)\xi] \\ G_{v_1} = \frac{6\beta_y}{L} (\xi^2 - \xi) \\ G_{v_2} = \frac{6\beta_y}{L} (-\xi^2 + \xi) \\ G_{w_1} = \frac{6\beta_z}{L} (\xi^2 - \xi) \\ G_{w_2} = \frac{6\beta_z}{L} (-\xi^2 + \xi) \\ G_{\theta_1} = \beta_y [3\xi^2 + (\alpha_y - 4)\xi + 1 - \alpha_y] \\ G_{\theta_2} = \beta_y [3\xi^2 - (\alpha_y + 2)\xi] \\ G_{\psi_1} = \beta_z [3\xi^2 + (\alpha_z - 4)\xi + 1 - \alpha_z] \\ G_{\psi_2} = \beta_z [3\xi^2 - (\alpha_z + 2)\xi] \end{array} \right\} \quad \text{Funciones de forma para barras}$$

donde  $x'$  es la coordenada local sobre la viga:

$$\xi = \frac{x'}{L}, \quad \alpha_y = \frac{12EI_y}{kGAL^2}, \quad \beta_y = \frac{1}{1 - \alpha_y}, \quad \alpha_z = \frac{12EI_z}{kGAL^2}, \quad \beta_z = \frac{1}{1 - \alpha_z}$$

donde  $k$  es el **factor de corrección por corte**\* y depende de la sección y del modulo de Poisson  $\nu$ . Algunos valores en la tabla 3 del anexo.

Estas funciones de forma interpolan los desplazamientos sobre la viga:

$$\left\{ \begin{array}{l} u = N_1 u_1 + N_2 u_2 \\ v = H_{v_1} v_1 + H_{\theta_1} \theta_1 + H_{v_2} v_2 + H_{\theta_2} \theta_2 \\ w = H_{w_1} w_1 + H_{\psi_1} \psi_1 + H_{w_2} w_2 + H_{\psi_2} \psi_2 \\ \varphi = N_1 \varphi_1 + N_2 \varphi_2 \\ \theta = G_{v_1} v_1 + G_{\theta_1} \theta_1 + G_{v_2} v_2 + G_{\theta_2} \theta_2 \\ \psi = G_{w_1} w_1 + G_{\psi_1} \psi_1 + G_{w_2} w_2 + G_{\psi_2} \psi_2 \end{array} \right. \quad (6)$$

para luego calcular los esfuerzos usando las mismas formulas vistas en estática y resistencia de materiales.

$$\begin{aligned} M_z &= EI_z \frac{d^2 v}{dx^2}, & V_y &= \frac{dM_z}{dx} = EI_z \frac{d^3 v}{dx^3}, & N_x &= AE \frac{u_2 - u_1}{L} \\ T &= GJ_T \frac{\varphi_2 - \varphi_1}{L}, & M_y &= EI_y \frac{d^2 w}{dx^2}, & V_z &= EI_y \frac{d^3 w}{dx^3} \end{aligned}$$

donde  $J_T$  es la constante torsional de la viga y  $A$  es la sección. <sup>†</sup>

La matriz de rigidez se puede obtener<sup>‡</sup> integrando analíticamente a la funciones de forma mencionadas anteriormente sobre el largo de la viga. En este documento no se va tratar la matriz de rigidez que

\*Ideado por Timoshenko en 1921 para permitir un mejor cálculo de las frecuencias naturales Dong et al. [2010].

<sup>†</sup>La constante torsional  $J_T$  es igual a  $I_p$  para secciones de viga circulares. Para perfiles abiertos de paredes delgadas, como por ejemplo un perfil doble T o un perfil 'C',  $J_T$  es mucho mas chico que  $I_p$ .

<sup>‡</sup>Para una viga uniforme con sección simétrica y de material en su rango elástico.

resulta de dicha integración. Se presenta al lector la matriz de rigidez de una viga Timoshenko 3D clásica Cook et al. [2007]:

$$[\mathbf{k}]_{1D} = \begin{bmatrix} X & 0 & 0 & 0 & 0 & 0 & -X & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_1 & 0 & 0 & 0 & Y_2 & 0 & -Y_1 & 0 & 0 & 0 & Y_2 \\ 0 & 0 & Z_1 & 0 & -Z_2 & 0 & 0 & 0 & -Z_1 & 0 & -Z_2 & 0 \\ 0 & 0 & 0 & S & 0 & 0 & 0 & 0 & 0 & -S & 0 & 0 \\ 0 & 0 & -Z_2 & 0 & Z_3 & 0 & 0 & 0 & Z_2 & 0 & Z_4 & 0 \\ 0 & Y_2 & 0 & 0 & 0 & Y_3 & 0 & -Y_2 & 0 & 0 & 0 & Y_4 \\ -X & 0 & 0 & 0 & 0 & 0 & X & 0 & 0 & 0 & 0 & 0 \\ 0 & -Y_1 & 0 & 0 & 0 & -Y_2 & 0 & Y_1 & 0 & 0 & 0 & -Y_2 \\ 0 & 0 & -Z_1 & 0 & Z_2 & 0 & 0 & 0 & Z_1 & 0 & Z_2 & 0 \\ 0 & 0 & 0 & -S & 0 & 0 & 0 & 0 & 0 & S & 0 & 0 \\ 0 & 0 & -Z_2 & 0 & Z_4 & 0 & 0 & 0 & Z_2 & 0 & Z_3 & 0 \\ 0 & Y_2 & 0 & 0 & 0 & Y_4 & 0 & -Y_2 & 0 & 0 & 0 & Y_3 \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ w_1 \\ \varphi_1 \\ \psi_1 \\ \theta_1 \\ u_2 \\ v_2 \\ w_2 \\ \varphi_2 \\ \psi_2 \\ \theta_2 \end{bmatrix} \quad (7)$$

donde

$$\begin{aligned} X &= \frac{AE}{L}, & Y_4 &= \frac{2EI_z}{L}, & Y_3 &= 2Y_4, & Y_2 &= \frac{3Y_4}{L}, & Y_1 &= \frac{2Y_2}{L} \\ Z_4 &= \frac{2EI_y}{L}, & Z_3 &= 2Z_4, & Z_2 &= \frac{2Z_4}{L}, & Z_1 &= \frac{2Z_2}{L}, & S &= \frac{GJ_T}{L} \end{aligned}$$

## 9. Orientación de elementos 1D

Los elementos detallados en la sección anterior están acostados sobre el eje  $x$ . Para orientar un elemento en el espacio se tiene que empezar de hablar de una *matriz de rotación*  $[\mathbf{T}]$ .

$$[\mathbf{k}]_{\text{rotada}} = [\mathbf{T}]^T [\mathbf{k}'] [\mathbf{T}] \quad (8)$$

donde  $[\mathbf{k}']$  es la matriz de rigidez local del elemento 1D antes de ser rotado. Luego de ser rotada la matriz puede ser acoplada a la rigidez global del sistema  $[\mathbf{K}]$ .

Cabe agregar que los desplazamientos relacionados con el sistema global corresponden a los ejes globales. Para obtener los desplazamientos locales de los elementos se tienen que rotar estos también!

$$\{\mathbf{d}'\} = [\mathbf{T}]\{\mathbf{d}\} \quad (9)$$

donde  $\{\mathbf{d}'\}$  son los desplazamientos del elemento en las coordenadas locales. Obtener  $\{\mathbf{d}'\}$  resulta útil al momento de querer calcular las fuerzas\* locales del elemento:  $[\mathbf{k}']\{\mathbf{d}'\} = -\{\mathbf{r}'\}$  o para interpolar los desplazamientos usando las expresiones en (6).

### Orientación de barras en el plano

La matriz de rigidez de una barra es definida horizontal. Esto significa que toda la rigidez está en  $x$ .

Una barra rotada  $\phi$  grados va tener una nueva rigidez en el eje global  $x$ , e incluso puede no tener rigidez si se rota 90 grados.

La matriz transformación de una barra en el plano está dada por

$$[\mathbf{T}]_{\text{Barra}} = \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 0 \\ 0 & \cos \phi \\ 0 & \sin \phi \end{bmatrix}$$

---

\*El signo negativo es para obtener las fuerzas **internas** actuando en el elemento.

## Orientación de vigas en el espacio

Para orientar una viga se tiene que aportar más información que para una barra. Mientras que una barra se puede orientar con el simple input de sus nodos, la viga de Timoshenko necesita orientar su eje  $y'$  además de su eje  $x'$ .

Algunos programas como ADINA, por ejemplo, permiten al usuario especificar un nodo auxiliar o un vector auxiliar que apunta en la dirección en la cual se desea que la viga tenga mayor momento de inercia.

Supongamos que elegimos este vector auxiliar  $s_y = [0, 1, 0]$ . Esto nos va dar el mayor momento ante la flexión para una carga en  $y$  global (suponiendo que  $I_z > I_y$ ). Para hallar la matriz de transformación  $[T]$  debemos buscar los cosenos directores de nuestra viga en el espacio. Una vez que obtenemos el versor de orientación  $\hat{v}_x$  es trivial obtener  $\hat{v}_z$

$$\hat{v}_z = \frac{\hat{v}_x \times s_y}{\|\hat{v}_x \times s_y\|}$$

Luego de armar los versores  $\hat{v}_x$ ,  $\hat{v}_y$ ,  $\hat{v}_z$  se define la matriz de los cosenos directores  $\lambda$

$$\lambda = \begin{bmatrix} \hat{v}_x \\ \hat{v}_y \\ \hat{v}_z \end{bmatrix}_{3 \times 3}$$

La matriz de transformación entonces es

$$[T]_{1D} = \begin{bmatrix} \lambda & \dots & 0 \\ & \lambda & \vdots \\ \vdots & & \lambda \\ 0 & \dots & \lambda \end{bmatrix}_{12 \times 12} \quad (10)$$

## 10. Ejercicios

### Pórticos

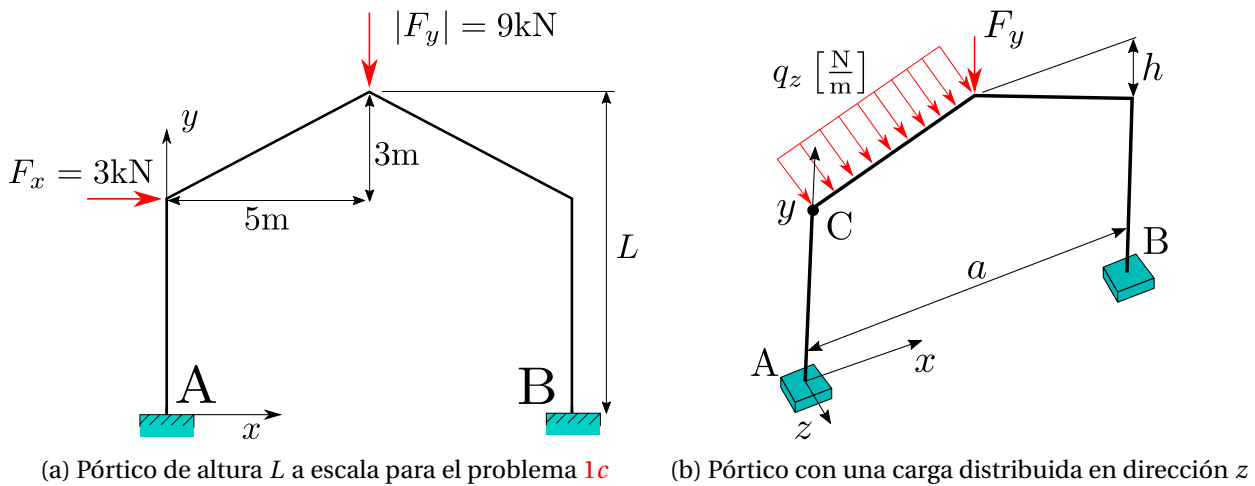


Figura 3

1. Para la figura 3a efectuar un modelo matemático y luego discretizar para resolver:

- Verificar que para el caso dado con  $F_x = 0$  las reacciones en  $y$  de los empotramientos son  $\frac{F_y}{2}$ .
- Se diseñó el pórtico con una altura  $L = 7$  metros con perfiles IPN 160. Un análisis preliminar de una consultora sugiere que el punto superior supera los desplazamientos máximos permitidos de  $d_{\text{máx}} = 15\text{mm}$ . Verificar.
- ¿Que altura debería el pórtico así el punto superior no se desplaza mas que 26.7mm? Usar perfil IPB\* 120. ¿Es esta la configuración de la figura 3a? (hint: si lo es)

\*Se puede buscar también como perfil HEB.

2. Para la figura 3b considerar perfiles HEB 340.

a) Las reacciones considerando  $a = 12\text{m}$ ,  $h = 2\text{m}$  y una altura de  $8\text{m}$ .  $q_z = 500$  y  $F_y = 20\text{kN}$ .

b) ¿Que orientación de vigas es favorable para reducir el desplazamiento del punto C?

Algunas Respuestas considerando  $E = 200\text{GPa}$ :

1b)  $d = 0.0155\text{m}$ ,  $\{\mathbf{R}_A\} \approx \{1, 1; 4, 1; -1, 28\}$  kN o kNm

2a)  $\{\mathbf{R}_B\} \approx \{-6, 2; 10; -0, 79; -0, 64; -6, 4; -0, 5; 16\}$  kN o kNm,

2b)  $d_C = 2.38\text{mm}$

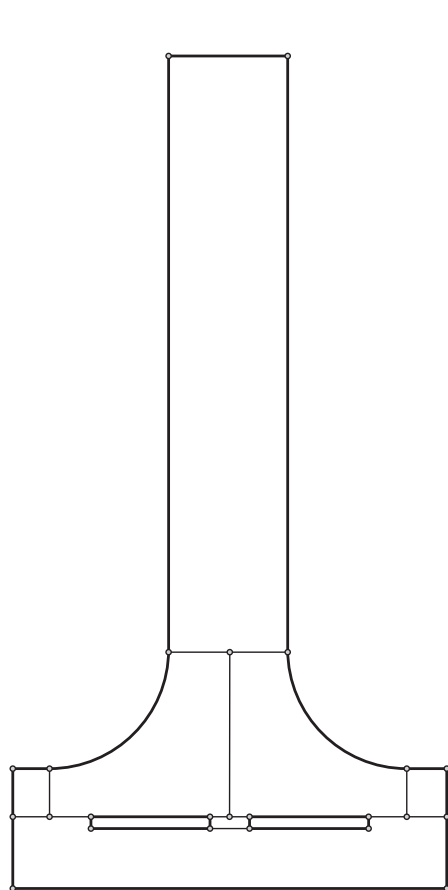
## Parte III

# Elementos 2D

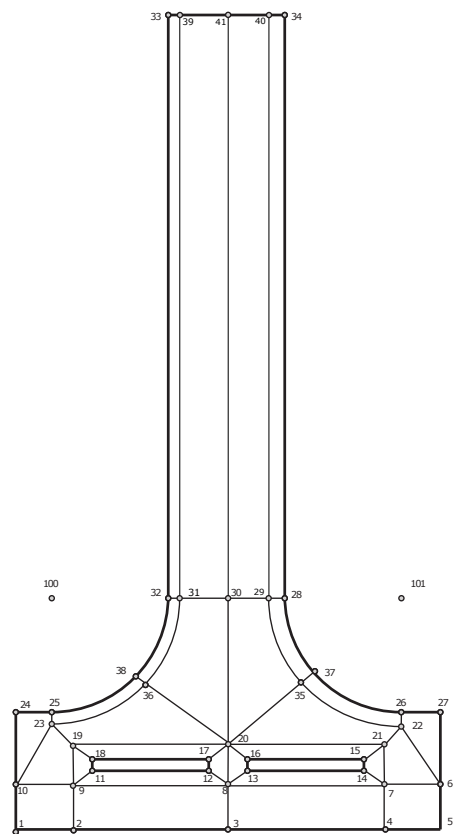
## 11. Protips para el mallado

Para mallar como un pro, empezar dividiendo el dominio. Cosas a tener en mente:

- Donde está la sección de mi dominio que quiero estudiar? Puedo refinar en esa zona?
- Que orientación va tener mi malla para cada división de dominio?
- Que forma tendrán mis elementos con una dada división? Habrá una mejor forma de dividir mi dominio así no tengo jacobianos estrambóticos?



(a) Mallable con triángulos solamente.



(b) Se puede mallar con elementos cuadrilateros. No hay esquinas angulosas. Es posible obtener *una buena malla*. Los nodos 100 y 101 son auxiliares y no forman parte de la malla.

Figura 4: Dos formas de dividir una paleta de una turbina pelton que recibe un chorro incidente en su esquina superior derecha.

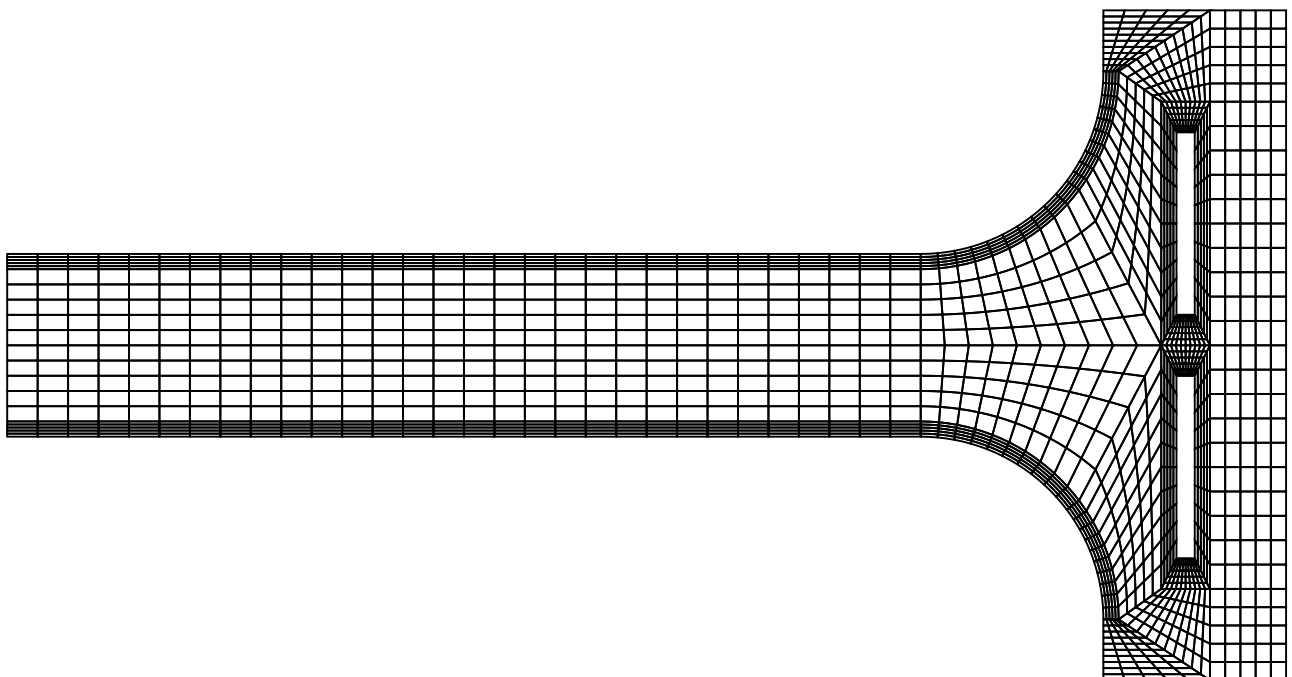


Figura 5: Ejemplo de malla usando divisiones de 4b. Se refina en los costados donde se tiene flexión.

Protip: Necesitas saber que fuerza se tiene que hacer para que se mantenga en posición una arista o punto dado cargas térmicas/fuerza? Apoyalo (fix) y mira las reacciones con la carga térmica/fuerza.

## 12. Funciones de formas para elementos 2D

Se define cuantos nodos se va tener por elemento y se los ubica en el espacio  $(\xi, \eta)$  que por simplicidad se trataran como  $(x, y)$ . Con el triangulo de Pascal para polinomios se elige el grado del polinomio y los términos. Luego se resuelve el sistema de ecuaciones  $[\mathbf{N}] \cdot \mathbf{X} = \mathbf{A}$  donde  $[\mathbf{N}] = [N_1 \ N_2 \ \dots \ N_n]$  y  $\mathbf{X} = [1 \ x \ y \ \dots \ x^{k-1} y^k \ x^k y^k]^T$ , o algo por el estilo. Se tienen que elegir los grados mas convenientes teniendo en cuenta la simetría y el número de nodos, este ultimo te limita el número de términos posibles por la naturaleza de la interpolación. La matriz  $\mathbf{A}$  tendrá en su **espacio fila** el mismo polinomio evaluado en la posición del nodo correspondiente a esa fila.

$$\mathbf{A} = \begin{bmatrix} 1 & x_1 & y_1 & \dots & x_1^{k-1} y_1^k & x_1^k y_1^k \\ 1 & x_2 & y_2 & \dots & x_2^{k-1} y_2^k & x_2^k y_2^k \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_n & y_n & \dots & x_n^{k-1} y_n^k & x_n^k y_n^k \end{bmatrix}$$

Luego, las funciones de forma  $[\mathbf{N}]$  se pueden obtener así:  $[\mathbf{N}] = \mathbf{X}^{-1} \mathbf{A}$

## Cargas 2-D

La ecuación que rige como se cargan elementos, siendo  $\{\mathbf{r}\}$  las cargas nodales del elemento,  $\{\mathbf{F}\}$  fuerzas volumétricas,  $\{\Phi\}$  fuerzas de tracción superficiales,  $\{\epsilon_0\}$  las deformaciones iniciales y  $\{\sigma_0\}$  las tensiones iniciales (pg. 228)

$$\{\mathbf{r}\} = \int [\mathbf{N}]^T \{\mathbf{F}\} dV + \int [\mathbf{N}]^T \{\Phi\} dS + \int [\mathbf{B}]^T [\mathbf{E}] \{\epsilon_0\} dV - \int [\mathbf{B}] \{\sigma_0\} dV \quad (11)$$

## 13. Elementos isoparamétricos

- Un elemento que no esta distorsionado (sigue siendo rectangular) tiene  $J$  constante
- Cuidado con modo espurio. Ver tabla 6.8-1 pg. 226 el tema de full/reduced integration [Cook et al. \[2007\]](#)
- Como cargar tu elemento isoparamétrico en pg. 228

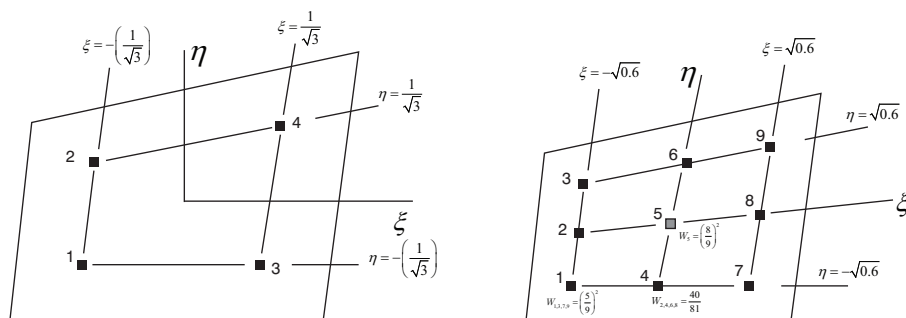


Figura 6: Puntos gauss para ordenes  $n = 2$  y  $n = 3$ . El peso para  $n = 2$  es igual en todos los puntos  $W_i = 1$

## Ejemplo elemento exótico

### Matriz de Rigidez

Imaginemos un elementos Q5 cuadrado de  $2 \times 2$  con espesor  $t$  (igual al Q4 con un nodo en su centro). Si fuéramos a obtener las funciones de formas de dicho elemento quedarían iguales para  $(x, y)$  y para  $(\xi, \eta)$

por las dimensiones usadas. La funcionalidad que uno estaría tentado a seleccionar sería  $[1, x, y, x^2, y^2]$ , pero está trae problemas inesperados debido a que tiene varias soluciones en la interpolación. Como nuestra prioridad siempre es mantener la simetría la funcionalidad será  $[1, x, y, xy, x^2y^2]$ . Tomando el orden de la figura 7.

$$[\mathbf{N}] = \begin{bmatrix} \frac{x^2y^2}{4} + \frac{xy}{4} - \frac{x}{4} - \frac{y}{4}, & \frac{x^2y^2}{4} - \frac{xy}{4} + \frac{x}{4} - \frac{y}{4}, & \frac{x^2y^2}{4} + \frac{xy}{4} + \frac{x}{4} + \frac{y}{4}, & \frac{x^2y^2}{4} - \frac{xy}{4} - \frac{x}{4} + \frac{y}{4}, & 1 - x^2y^2 \end{bmatrix}$$

Llegado a este punto nos interesa obtener la matriz de rigidez. Si queremos lograr “full integration” deberíamos usar Gauss orden  $n = 3$  según  $2n - 1 \geq O([\mathbf{B}]^T[\mathbf{E}][\mathbf{B}])$ .  $[\mathbf{B}]$  es el *strain-deformation matrix*. El producto  $[\mathbf{B}]^T[\mathbf{E}][\mathbf{B}]$  da un polinomio de orden 6 ( $[\mathbf{B}]$  tiene el mismo orden que la derivada de  $[\mathbf{N}]$ ). De esta forma nos aseguramos que nuestro resultado va ser exacto para el elemento sin distorsionar.

Para esté ejemplo, no se pide *full integration* entonces no pasa nada si queremos *underintegrate*. Usamos Gauss orden  $n = 2$ .

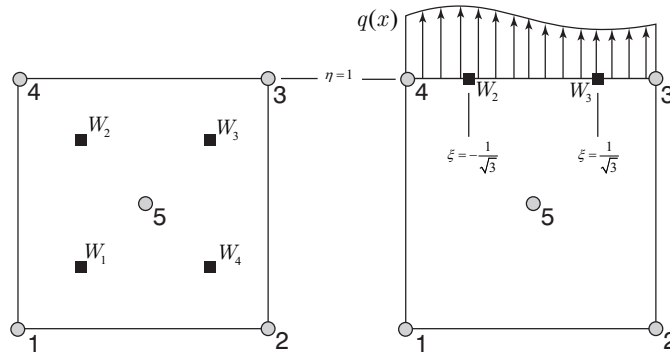


Figura 7: Elemento Q5 rectangular.

La rigidez de un elemento está dada por

$$[\mathbf{k}] = \int [\mathbf{B}]^T[\mathbf{E}][\mathbf{B}]dV \quad (12)$$

para un elemento plano la ecuación anterior es

$$[\mathbf{k}]_{2D} = \iint [\mathbf{B}]^T[\mathbf{E}][\mathbf{B}]t dx dy = \int_{-1}^1 \int_{-1}^1 [\mathbf{B}]^T[\mathbf{E}][\mathbf{B}]t |\mathbf{J}| d\xi d\eta$$

donde  $[\mathbf{B}]$  es la matriz deformación-desplazamiento del elemento,  $[\mathbf{E}]$  es la matriz constitutiva, y  $|\mathbf{J}|$  es el determinante de la matriz Jacobiana, el cual se le suele decir simplemente el Jacobiano.

Este ultimo se calcula a partir de la derivada de las funciones de forma

### Carga de linea

Si el elemento está cargado sobre la linea 4-3 con una distribuida  $q(x)$  (en  $[N m^{-1}]$ ) entonces procedemos de la siguiente manera según el segundo término de (11):

$$r_{xi} = \int_{-1}^1 N_i(\tau \mathbf{J}_{11} - \sigma \mathbf{J}_{12})t d\xi \quad (13)$$

$$r_{yi} = \int_{-1}^1 N_i(\sigma \mathbf{J}_{11} + \tau \mathbf{J}_{12})t d\xi \quad (14)$$

donde  $\sigma$  es la sollicitación normal a la superficie y  $\tau$  es la tangencial. Para la fuerza sobre el nodo 4 se tiene

$$r_{y4} = N_4(\xi_2)t[\sigma(\xi_2)\mathbf{J}_{11} + \tau(\xi_2)\mathbf{J}_{12}] \cdot \mathbf{W}_2 + N_4(\xi_3)t[\sigma(\xi_3)\mathbf{J}_{11} + \tau(\xi_3)\mathbf{J}_{12}] \cdot \mathbf{W}_3$$

Si consideramos que solo hay una *carga distribuida de linea* a tracción/compresión como indica la figura 7, se reduce la ecuación anterior

$$r_{y4} = N_4(\xi_2) \mathbf{J}_{11} q(\xi_2) + N_4(\xi_3) \mathbf{J}_{11} q(\xi_3) = N_4 q \mathbf{J}_{11} \Big|_{\xi_2} + N_4 q \mathbf{J}_{11} \Big|_{\xi_3}$$

similarmente  $r_{y3} = N_3 q \mathbf{J}_{11} \Big|_{\xi_2} + N_3 q \mathbf{J}_{11} \Big|_{\xi_3}$  donde la matriz Jacobiana también se evalúa para cada punto de Gauss!

## Tensiones

- Las tensiones en los nodos suele ser de mayor interés que sobre los puntos de gauss (mas comprometidas, permiten estimar error)

## 14. Elementos Axisimetricos

Resuelvo problema 3-D en el plano. Los resultados son por cada unidad radian. Como sigo teniendo dos grados de libertad tengo las mismas funciones de forma. Cambia mi operador derivada.

$$\begin{Bmatrix} \sigma_r \\ \sigma_\theta \\ \sigma_z \\ \tau_{rz} \end{Bmatrix} = \frac{(1-\nu)E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & f & f & 0 \\ & 1 & f & 0 \\ & & 1 & 0 \\ \text{sim.} & & & g \end{bmatrix} \begin{pmatrix} \begin{Bmatrix} \epsilon_r \\ \epsilon_\theta \\ \epsilon_z \\ \gamma_{rz} \end{Bmatrix} - \begin{Bmatrix} \alpha T \\ \alpha T \\ \alpha T \\ 0 \end{Bmatrix} \end{pmatrix}$$

donde

$$f = \frac{\nu}{1-\nu} \quad \text{y} \quad g = \frac{1-2\nu}{2(1-\nu)}$$

Una carga puntual  $P$  aplicada sobre un elemento axisimétrico no tiene el mismo significado físico que en elementos plane stress/strain.

$$P = 2\pi r q$$

donde  $q$  es la carga distribuida en [N/m],  $r$  es la distancia al eje de revolución y  $2\pi$  es el resultado de integrar la fuerza distribuida sobre  $\theta$ .

$$\{\mathbf{r}_e\} = \int \int_{-\pi}^{\pi} [\mathbf{N}]^T \begin{Bmatrix} \rho r \omega^2 \\ 0 \end{Bmatrix} r d\theta dA$$

## Parte IV

## Otras cosas

## 15. Desplazamientos iniciales

Si nuestro sistema tiene desplazamientos iniciales conocidos se puede formular un sistema a resolver:

$$\begin{bmatrix} [\mathbf{K}_{cc}] & [\mathbf{K}_{cx}] \\ [\mathbf{K}_{xc}] & [\mathbf{K}_{xx}] \end{bmatrix} \begin{Bmatrix} \{\mathbf{D}_c\} \\ \{\mathbf{D}_x\} \end{Bmatrix} = \begin{Bmatrix} \{\mathbf{R}_x\} \\ \{\mathbf{R}_c\} \end{Bmatrix} \quad (15)$$

Note que para los nodos donde se conocen los desplazamientos (los nodos cc) *se desconocen las cargas*.

$$\{\mathbf{D}_x\} = [\mathbf{K}_{xx}]^{-1} (\{\mathbf{R}_x\} - [\mathbf{K}_{xc}]\{\mathbf{D}_c\})$$

luego de obtener los desplazamientos desconocidas se puede obtener las cargas desconocidas (sobre los nodos conocidos):

$$\{\mathbf{R}_c\} = [\mathbf{K}_{cx}]\{\mathbf{D}_x\} + [\mathbf{K}_{cc}]\{\mathbf{D}_c\}$$



## 16. Transferencia de Calor

Tenemos el flujo de calor que es  $\underline{K}\nabla T$ , la divergencia de esto es el flujo neto que pasa por un punto.

$$\nabla(\underline{K}\nabla T) + Q = \rho c \frac{\partial T}{\partial \tau}$$

Ley de Newton:

$$q_c = h_c(s, T)(T - T_\infty)$$

Ley de Stefan Boltzmann

$$q_r = \varepsilon_r \sigma (T^4 - T_\infty^4)$$

Es mas jodido el tema de modelar temperaturas fijas e imponer calor transferido, pues son aseveraciones no tan reales como imponer desplazamiento cero sobre un apoyo y fuerzas sobre vigas.

Para aplicar conveccion

Hay una parte que depende de la temperatura interna, la parte que no depende la dejo como vector de cargas. La parte que depende la tengo que sumar a mi matriz de conductividades

$$R_{C_i} = \oint_{\Gamma_{conv}} N_i h (T(x) - T_{fl}) d\gamma$$

Lo escribe sebas:  $kT = q_c = HT - RT_{fl} \longrightarrow (K - H)T = -RT_{fl}$

$$\text{Cargas : } R_H : \oint_{\Gamma_{conv}} N_i h T_{fl} d\gamma \quad \text{Conductividad : } K : \oint_{\Gamma_{cm}} N_i h N_j d\gamma$$

### Resolución de problemas típico

Condiciones de borde de temperatura

$$\{\mathbf{T}_x\} = [\mathbf{K}_{xx}]^{-1} (\{\mathbf{Q}_x\} - [\mathbf{K}_{xc}]\{\mathbf{T}_c\})$$

luego de obtener las temperaturas desconocidas se puede obtener el flujo desconocido (sobre los nodos conocidos):

$$\{\mathbf{Q}_c\} = [\mathbf{K}_{cx}]\{\mathbf{T}_x\} + [\mathbf{K}_{cc}]\{\mathbf{T}_c\}$$

## 17. Formulación elemento Hexaedro de 8 nodos

El modelado con elementos isoparamétricos hexaedros de 8 nodos es un buen punto de partida para comenzar a manejar los elementos finitos en 3 dimensiones. Un código hecho para obtener  $[\mathbf{K}]$  con elementos H8 se adapta con facilidad para los H20.

Cada nodo tendrá 3 grados de libertad, dándonos 24 dof por elemento. El elemento H20 de la figura 8 esta planteado de tal forma que los primeros nodos del 1 al 8 son los nodos del H8 que se va formular a continuación.

La funcionalidad a usar es la siguiente

$$X_{H8} = [1, \xi, \eta, \zeta, \xi\eta, \xi\eta, \eta\zeta, \xi\eta\zeta]$$

y la matriz constitutiva para el espacio 3D con 3 dof por nodo se escribe:

$$[\mathbf{E}] = \begin{bmatrix} 2G + \lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2G + \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2G + \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & G & 0 & 0 \\ 0 & 0 & 0 & 0 & G & 0 \\ 0 & 0 & 0 & 0 & 0 & G \end{bmatrix} \quad [\mathbf{E}]^{-1} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & f & 0 & 0 \\ 0 & 0 & 0 & 0 & f & 0 \\ 0 & 0 & 0 & 0 & 0 & f \end{bmatrix} \quad (16)$$

donde  $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$  y  $G = \frac{E}{2(1+\nu)}$  y  $f = 2 + 2\nu$ .

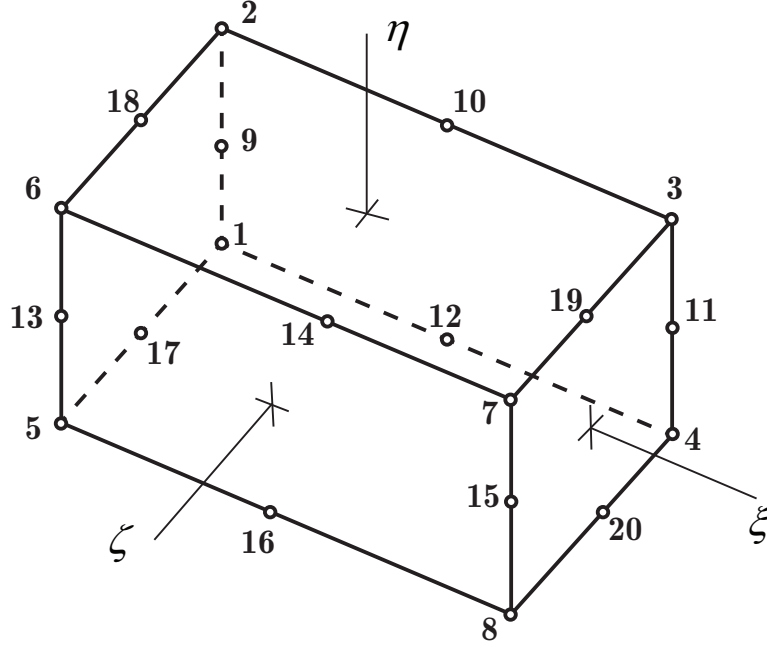


Figura 8: Numeración de nodos H20 en ADINA(Ejes sugeridos)

### Formulación elementos

El jacobiano tiene la forma

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}$$

pudiendo ser calculado de la siguiente forma

$$\mathbf{J} = \begin{bmatrix} \frac{\partial}{\partial \xi}[\mathbf{N}] \\ \frac{\partial}{\partial \eta}[\mathbf{N}] \\ \frac{\partial}{\partial \zeta}[\mathbf{N}] \end{bmatrix} \cdot \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ x_4 & y_4 & z_4 \\ x_5 & y_5 & z_5 \\ x_6 & y_6 & z_6 \\ x_7 & y_7 & z_7 \\ x_8 & y_8 & z_8 \end{bmatrix} \quad (17)$$

donde la primer matriz termina siendo  $3 \times 8$  para un elemento H8. La segunda matriz son las posiciones *globales* de los nodos del elemento. El jacobiano se puede entonces utilizar para calcular

$$[\partial \mathbf{N}] = \begin{bmatrix} \frac{\partial}{\partial x}[\mathbf{N}] \\ \frac{\partial}{\partial y}[\mathbf{N}] \\ \frac{\partial}{\partial z}[\mathbf{N}] \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \frac{\partial}{\partial \xi}[\mathbf{N}] \\ \frac{\partial}{\partial \eta}[\mathbf{N}] \\ \frac{\partial}{\partial \zeta}[\mathbf{N}] \end{bmatrix} \quad (18)$$

con lo obtenido se puede calcular la matriz  $[\mathbf{B}]$ .

La matriz *strain-deformation* queda

$$[\mathbf{B}] = [B_1 \quad B_2 \quad B_3 \quad B_4 \quad B_5 \quad B_6 \quad B_7 \quad B_8]$$

donde

$$B_i = \begin{bmatrix} \partial N_i / \partial x & 0 & 0 \\ 0 & \partial N_i / \partial y & 0 \\ 0 & 0 & \partial N_i / \partial z \\ 0 & \partial N_i / \partial z & \partial N_i / \partial y \\ \partial N_i / \partial z & 0 & \partial N_i / \partial x \\ \partial N_i / \partial y & \partial N_i / \partial x & 0 \end{bmatrix} \quad (19)$$

Finalmente un calcula la rigidez del elemento usando (12).

## Parte V

# No-linealidad y análisis dinámico

## 18. Respuesta dinámica estructural

$$[M]\{\ddot{\mathbf{D}}\} + [C]\{\dot{\mathbf{D}}\} + [K]\{\mathbf{D}\} = \{\mathbf{R}^{\text{ext}}\} \quad (20)$$

Se puede resolver la ecuación de arriba para un sistema dado sin amortiguamiento y sin cargas externas\* para obtener sus frecuencias naturales y los modos asociados a estos.

$$\{\mathbf{D}\} = \{\bar{\mathbf{D}}\} \sin \omega t, \quad \{\ddot{\mathbf{D}}\} = -\omega^2 \{\bar{\mathbf{D}}\} \sin \omega t$$

sin considerar la matriz de amortiguamiento se obtiene el problema de autovalores:

$$([K] - \omega^2[M])\{\bar{\mathbf{D}}\} = \{\mathbf{0}\}$$

donde  $\omega^2$  es un autovalor y la raíz de estos son las frecuencias naturales.

Amortiguamiento  $[C] = \alpha[M] + \beta[K]$  cede una matriz no diagonal. Se complica la resolución. Existen dos otros modelos que tratan con una matriz  $[C_\Phi]$  diagonal donde las ecuaciones se desacoplan.

Amortiguamiento Modal: Se elige un  $\zeta$  para cada modo

$$[C_\Phi] = \begin{bmatrix} 2\omega_n\zeta_n & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 2\omega_1\zeta_1 \end{bmatrix} \quad (21)$$

Amortiguamiento proporcional. Se basa el análisis

$$[C_\Phi] = [\Phi]^T (\alpha[M] + \beta[K]) [\Phi] = \alpha\delta[I] + \beta[\Omega^2] \quad (22)$$

Si se quiere estudiar un rango de frecuencias de excitación tal que  $\omega_{\text{exc}} \in [\omega_1, \omega_2]$  y eligiendo dos valores de damping para ambas frecuencias  $\zeta_1$  y  $\zeta_2$  se tiene:

$$\alpha = 2\omega_1\omega_2(\zeta_1\omega_2 - \zeta_2\omega_1)/(\omega_2^2 - \omega_1^2)$$

$$\beta = 2(\zeta_2\omega_2 - \zeta_1\omega_1)/(\omega_2^2 - \omega_1^2)$$

Una vez obtenida  $[C_\Phi]$  se pueden obtener los desplazamientos modales  $\{\mathbf{Z}\}$ . Tome en cuenta que debido a la diagonalidad de  $[\Omega^2]$  y  $\{\mathbf{R}_\Phi\}$  se desacoplan las ecuaciones de 20 y por ende se pasa a tratar dichas matrices diagonales como vectores columnas. Una vez desacopladas se tiene

$$\{\ddot{\mathbf{Z}}\} + 2\{\Omega\}\{\mathbf{C}_\Phi\}\{\dot{\mathbf{Z}}\} + \{\Omega^2\}\{\mathbf{Z}\} = \{\mathbf{R}_\Phi\}$$

$$\{\mathbf{Z}\} = \frac{\{\mathbf{R}_\Phi\}}{\{\Omega^2\}\sqrt{(1-\chi^2)^2 + (2\{\mathbf{C}_\Phi\}\chi)^2}}$$

donde  $\chi = \frac{\omega_{\text{exc}}}{\{\Omega\}}$ .

## Sine Sweep

A medida que la frecuencia de excitación aumenta la *amplitud del sistema disminuye*<sup>†</sup>. Es interesante pensar que si aumentara no tendría sentido buscar las frecuencias naturales porque estas son caracterizadas por un máximo de amplitud. Las curvas del barrido de frecuencia son decrecientes en lejanía de una frecuencia natural porque para una fuerza cíclica  $F(t) = F_0 \sin \omega t$  el tiempo que actúa en una dirección es inversamente proporcional a la frecuencia. Por ende la estructura no tiene tiempo para moverse lejos antes de que se invierta la dirección de la fuerza.

\*Se denomina vibración "libre" cuando no hay cargas asociadas. Si no hay amortiguamiento el desplazamiento es regido por  $u = \bar{u} \sin \omega t$ , donde  $\bar{u}$  es la amplitud de vibración y  $\omega$  es la frecuencia circular Cook et al. [2007].  $\omega$  es obtenida en radianes por segundo.

<sup>†</sup>Excepto en cercanías de una frecuencia natural

## 19. Transferencia de calor no-lineal y transitoria

### Radiacion

Cuando se tienen problemas de radiación se puede iterar para obtener el perfil usando **relajación**.

$$\begin{cases} \{\mathbf{T}_x\}_{\text{unrelaxed}}^{n+1} = [\mathbf{K}_{xx}]^{-1} (\{\mathbf{R}_x\}^n - [\mathbf{K}_{xc}]\{\mathbf{T}_c\}^n) \\ \{\mathbf{R}\}^n = \{\mathbf{R}_{\text{generado}}\} + \{\mathbf{R}_{\text{rad}}\}^n \\ \{\mathbf{T}\}^{n+1} = \{\mathbf{T}\}^n + \frac{1}{k_R} \cdot (\{\mathbf{T}\}_{\text{unrelaxed}}^{n+1} - \{\mathbf{T}\}^n) \end{cases} \quad (23)$$

donde  $k_R$  es la relajación o factor de atenuación de temperaturas. Cuanto mayor es más “amortiguada” es la convergencia del perfil. Se puede asegurar la convergencia de la solución pero a la vez será más lenta a mayores  $k_R$ .

### Transitorio

Matriz capacidad

$$[C] = \int_{\Omega} [N]^T \rho c [N] d\omega$$

Temperature-Heat flux matrix  $[\mathbf{B}]$  para Q4:

$$[\mathbf{B}] = \begin{Bmatrix} \frac{\partial N}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial N}{\partial \eta} \frac{\partial \eta}{\partial x} \\ \frac{\partial N}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial N}{\partial \eta} \frac{\partial \eta}{\partial y} \end{Bmatrix}$$

Si a beta le digo que vale cero el futuro muere en cambio si beta vale 1 entonces TODO depende del futuro.

$$\beta [C] \{\dot{T}\}^{n+1} + (1-\beta) [C] \{\dot{T}\}^n + [K] \beta \{T\}^{n+1} + [K] (1-\beta) \{T\}^n = (1-\beta) \{R_T\}^n + \beta \{R_T\}^{n+1}$$

Ecuación iterativa:

$$\{\mathbf{T}\}^{n+1} = ([\mathbf{C}] + \Delta t \beta [\mathbf{K}])^{-1} \left( ([\mathbf{C}] - \Delta t (1-\beta) [\mathbf{K}]) \{\mathbf{T}\}^n + \Delta t ((1-\beta) \{\mathbf{R}\}^n + \beta \{\mathbf{R}\}^{n+1}) \right) \quad (24)$$

$\beta = 0$	Euler Forward Difference
$\beta = 0.5$	Crank Nicholson
$\beta = 0.666$	Galerkin
$\beta = 1$	Backward Difference

## 20. Análisis Dinámico

### Método Integración Directa

Resuelve 20.

$$\begin{cases} \{\mathbf{V}\}^{n+1} = \{\mathbf{V}\}^n + \frac{\Delta t}{2} [\mathbf{M}]^{-1} (\{\mathbf{R}^{\text{ext}}\} - [\mathbf{C}]\{\mathbf{V}\} - [\mathbf{K}]\{\mathbf{D}\}) \\ \{\mathbf{D}\}^{n+1} = \{\mathbf{D}\}^n + \frac{\Delta t}{2} \{\mathbf{V}\} \end{cases}$$

### Método Runge-Kutta K4

Tengo que resolver las dos ecuaciones simultáneamente

$$\begin{cases} \{\mathbf{V}\}^{n+1} = \{\mathbf{V}\}^n + \frac{\Delta t}{6} (\{k_1^{\dot{V}}\} + 2\{k_2^{\dot{V}}\} + 2\{k_3^{\dot{V}}\} + \{k_4^{\dot{V}}\}) \\ \{\mathbf{D}\}^{n+1} = \{\mathbf{D}\}^n + \frac{\Delta t}{6} (\{k_1^{\dot{D}}\} + 2\{k_2^{\dot{D}}\} + 2\{k_3^{\dot{D}}\} + \{k_4^{\dot{D}}\}) \end{cases}$$

donde

$$\begin{cases} \{k_1^{\dot{V}}\} = [\mathbf{M}]^{-1} (\{\mathbf{R}^{\text{ext}}\}^n - [\mathbf{C}]\{\mathbf{V}\}^n - [\mathbf{K}]\{\mathbf{D}\}^n) \\ \{k_1^{\dot{D}}\} = \{\mathbf{V}\}^n \\ \{k_2^{\dot{V}}\} = [\mathbf{M}]^{-1} (\{\mathbf{R}^{\text{ext}}\}^{n+\frac{1}{2}} - [\mathbf{C}](\{\mathbf{V}\}^n + \frac{\Delta t}{2}\{k_1^{\dot{V}}\}) - [\mathbf{K}](\{\mathbf{D}\}^n + \frac{\Delta t}{2}\{k_1^{\dot{D}}\})) \\ \{k_2^{\dot{D}}\} = \{\mathbf{V}\}^n + \frac{\Delta t}{2}\{k_1^{\dot{V}}\} \\ \{k_3^{\dot{V}}\} = [\mathbf{M}]^{-1} (\{\mathbf{R}^{\text{ext}}\}^{n+\frac{1}{2}} - [\mathbf{C}](\{\mathbf{V}\}^n + \frac{\Delta t}{2}\{k_2^{\dot{V}}\}) - [\mathbf{K}](\{\mathbf{D}\}^n + \frac{\Delta t}{2}\{k_2^{\dot{D}}\})) \\ \{k_3^{\dot{D}}\} = \{\mathbf{V}\}^n + \frac{\Delta t}{2}\{k_2^{\dot{V}}\} \\ \{k_4^{\dot{V}}\} = [\mathbf{M}]^{-1} (\{\mathbf{R}^{\text{ext}}\}^{n+1} - [\mathbf{C}](\{\mathbf{V}\}^n + \Delta t\{k_3^{\dot{V}}\}) - [\mathbf{K}](\{\mathbf{D}\}^n + \Delta t\{k_3^{\dot{D}}\})) \\ \{k_4^{\dot{D}}\} = \{\mathbf{V}\}^n + \Delta t\{k_3^{\dot{V}}\} \end{cases}$$

## Parte VI

# Anexo

### Tablas

$n=0$				1				
$n=1$			$x$		$y$			
$n=2$			$x^2$		$xy$		$y^2$	
$n=3$		$x^3$		$x^2y$		$xy^2$		$y^3$
$n=4$		$x^4$		$x^3y$		$x^2y^2$		$xy^3$
$n=5$	$x^5$		$x^4y$		$x^3y^2$		$x^2y^3$	
$n=6$	$x^6$		$x^5y$		$x^4y^2$		$x^3y^3$	
							$x^2y^4$	
								$xy^5$
								$y^6$

Tabla 2: El triangulo de pascal de orden  $n = 6$ .

Factor Corrección para $\nu \approx 0,25$	
Perfil	$k$
Rectangular $\frac{h}{b} \gtrsim 1$	$\frac{5}{6}$
Rectangular $\frac{h}{b} = 0,5$	0,7961
Rectangular $\frac{h}{b} = 0,25$	0,6308
Circular	$\frac{9}{10}$
Tubo de paredes delgadas circular	$\frac{1}{2}$
Wide Flange doble T	$k_y \approx \frac{A_f}{1,2A}$
	$k_z \approx \frac{A_w}{A}$

Tabla 3:  $A_w$  es el área del alma y  $A_f$  es el área del ala.

### Expresiones útiles

$$\sigma_v = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2 - \sigma_x \sigma_y - \sigma_x \sigma_z - \sigma_y \sigma_z + 3(\sigma_{xy}^2 + \sigma_{xz}^2 + \sigma_{yz}^2)} \quad (25)$$

$$\sigma_v = \sqrt{\frac{1}{2}[(\sigma_{11} - \sigma_{22})^2 + (\sigma_{11} - \sigma_{33})^2 + (\sigma_{22} - \sigma_{33})^2] + 3(\sigma_{12}^2 + \sigma_{13}^2 + \sigma_{23}^2)} \quad (26)$$

$$\lambda = \frac{E \nu}{(1 + \nu)(1 - 2\nu)} \quad \mu = G = \frac{E}{2(1 + \nu)} \quad (27)$$

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix} = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ 2\varepsilon_{xy} \end{bmatrix} \quad (28)$$

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \frac{E}{(1 + \nu)(1 - 2\nu)} \begin{bmatrix} 1 - \nu & \nu & 0 \\ \nu & 1 - \nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{bmatrix} \quad (29)$$

$$\sigma_n = \frac{\sigma_{xx} + \sigma_{yy}}{2} + \left( \frac{\sigma_{xx} - \sigma_{yy}}{2} \right) \cos 2\theta + \tau_{xy} \sin 2\theta \quad (30)$$

$$\tau_n = - \left( \frac{\sigma_{xx} - \sigma_{yy}}{2} \right) \sin 2\theta + \tau_{xy} \cos 2\theta \quad (31)$$

$$I = \int_{-1}^1 \phi(\xi) d\xi \approx \phi(\xi_1)W_1 + \phi(\xi_2)W_2 \dots \phi(\xi_n)W_n \quad (32)$$

$$I = \int_{-1}^1 \int_{-1}^1 \phi(\xi, \eta) d\xi d\eta \approx \sum_i \sum_j W_i W_j \phi(\xi, \eta) \quad (33)$$

Se suele requerir que  $\eta \leq 0,05$

$$\|U\|^2 = \sum_{i=1}^m \int_{v_e} \{\boldsymbol{\varepsilon}\}_i^T [\mathbf{E}] \{\boldsymbol{\varepsilon}\}_i dV$$

$$\|e\|^2 = \sum_{i=1}^m \int_{v_e} (\{\boldsymbol{\varepsilon}^*\}_i - \{\boldsymbol{\varepsilon}\}_i)^T [\mathbf{E}] (\{\boldsymbol{\varepsilon}^*\}_i - \{\boldsymbol{\varepsilon}\}_i) dV$$

$$\|e\|^2 = \sum_{i=1}^m \int_{v_e} (\{\boldsymbol{\sigma}^*\}_i - \{\boldsymbol{\sigma}\}_i)^T [\mathbf{E}]^{-1} (\{\boldsymbol{\sigma}^*\}_i - \{\boldsymbol{\sigma}\}_i) dV$$

$$\eta = \sqrt{\frac{\|e\|^2}{\|e\|^2 + \|U\|^2}}$$

## Figuras

## Referencias

- Robert D Cook et al. *Concepts and applications of finite element analysis*. John Wiley & Sons, 2007.
- Jack Chessa. Programing the finite element method with matlab. *Northwestern University*, 20(02), 2002.
- Uday S Dixit. Finite element method: an introduction. *Indian Institute of Technology Guwahati-781*, 39, 2007.
- Yunhua Luo. An efficient 3d timoshenko beam element with consistent shape functions. *Adv. Theor. Appl. Mech*, 1(3):95–106, 2008.
- Stanley B Dong, Can Alpdogan, and Ertugrul Taciroglu. Much ado about shear correction factors in timoshenko beam theory. *International Journal of Solids and Structures*, 47(13):1651–1665, 2010.

