

UNIVERSIDAD CATÓLICA DE LA SANTÍSIMA CONCEPCIÓN
FACULTAD DE INGENIERÍA
INGENIERÍA CIVIL



**DESARROLLO DE UNA HERRAMIENTA NUMÉRICA BASADA EN UN
ELEMENTO FINITO TRIDIMENSIONAL HEXAÉDRICO PARA EL
MODELAMIENTO Y ANÁLISIS MODAL DE PANELES DE
ALBAÑILERÍA.**

DIEGO FERNANDO IDE RIVERA
INFORME DE PROYECTO DE TÍTULO PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL

Profesor Guía

Dr. Edwin Behrens R.

Profesor Informante

Dr. Frank Sanhueza E.

Concepción, Mayo 2017

Resumen

Las construcciones de albañilería son uno de los sistemas constructivos más antiguos y aún se le sigue utilizando ampliamente debido a la durabilidad de este tipo de estructura y al bajo costo asociado a ésta. Sin embargo, aunque constantemente es utilizado en construcciones actuales, los ingenieros estructurales poseen un conocimiento limitado en relación al comportamiento de este tipo de sistemas estructurales bajo solicitaciones extremas (Chávez, 2013). Si bien, en otros países existen estudios dedicados a este tema, en Chile no existen estudios de este tipo y los resultados obtenidos en el extranjero no son extrapolables a la realidad nacional, porque los materiales utilizados, las normas que rigen su construcción y los métodos constructivos son completamente distintos (Galdames, 2014).

Como finalidad de este proyecto de investigación, se generó un código programado en el software Matlab, capaz de generar un mallado en 3D que represente adecuadamente los materiales constituyentes de la albañilería, y realizar una modelación numérica mediante el método de elementos finitos. Se compararon los resultados de un análisis de respuesta modal, del código generado en Matlab, con los resultados entregados por el software comercial SAP2000.

Se analizaron distintos modelos con cuatro geometrías distintas (1m x 1m, 2m x 1m, 2m x 2m y 2m x 3m) y dos tipos de condiciones de borde (empotrado en todos sus extremos y solamente en el extremo inferior). Se comparó la matriz de formas modales obtenidas mediante ambos software considerando deformaciones en los tres ejes del panel. Estas matrices fueron comparadas bajo el criterio de certeza modal (MAC). Los resultados de esta comparación basada en el factor MAC fueron buenos, ya que en la diagonal principal de la matriz MAC se obtuvo correlaciones casi perfectas (100%) en la gran mayoría de los casos.

Para el caso de las frecuencias naturales, se compararon los modos con mayor correlación según MAC, obteniendo un porcentaje de similitud de frecuencias entre dichos modos. Para los ocho modelos analizados, los porcentaje de similitud de frecuencias varían entre 84.01% y 99.99%. Por ejemplo para el modelo 1, que es donde se registró la similitud más baja, los porcentajes varían entre 84.01% y 98.35%. Por el contrario, el modelo 6 registró las similitudes más cercanas al 100%, variando entre 95.23% y 99.98%. El modelo 2 (1m x 1m y empotrado solamente en la base) fue discretizado en 299, 2392 y 19136 elementos en ambos software, mejorando notablemente el porcentaje de similitud entre las frecuencias, variando entre 86.68% - 99.95%, 95.43% - 99.32% y 99.03% - 99.99% para 299, 2392 y 19136 elementos respectivamente.

Abstract

The construction buildings are one of the oldest systems of construction and are still used widely due to the durability of this type of structure and their low cost. Although it is constantly used in current buildings, structural engineers have a limited knowledge related to the performance of this type of structural systems under extreme requirements (Chávez, 2003). Even though there are studies related to this topic in other countries, in Chile, such studies are absent, and the results obtained abroad are not comparable to the national reality because the materials used, the regulations that rule their construction and the structural methods are completely different (Galdames, 2014).

As the objective of this research, a programmed code was produced in the Matlab software, which is capable of creating a 3D model that properly represents the building materials, and making a numerical model with the method of finite elements. The results of an analysis of modal response were compared, from the code obtained in Matlab, with the results given by the software SAP200.

Different models with four different geometries were analyzed (1m x 1m, 2m x 1m, 2m x 2m y 2m x 3m) and two types of edge conditions (set into all of its ends and only into the inferior end). There was a comparison of matrix of modal forms obtained by means of both softwares, considering deformations in the panel shafts. These matrixes were compared under the modal assurance criterion (MAC). The results of this comparison based on the MAC factor were good, because in the main diagonal of the matrix MAC correlations were almost perfect (100%) in most of the cases.

For the case of natural frequencies, the modes with more correlation were compared, according to MAC, obtaining a percentage of frequency similarity between those. For the eight models analyzed, the percentages of frequency similarity vary between 84.01% and 99.99%. For instance, for the model 1, which showed the lowest similarity, the percentages vary around 84.01% and 98.35%. Conversely, the model 6 registered the similarities closer to 100%, varying between 95.23% and 99.98%. The model 2 (1m x 1m and set only into the base) was discretized in 299, 2392 and 19136 elements in both softwares, notably improving the similarity percentage between frequencies, going from 86.68% - 99.95%, 95.43% - 99.32% and 99.03% - 99.99 % para 299, 2392 and 19136 elements respectively.

Tabla de contenidos

1. Introducción	1
1.1. Objetivos	3
2. Marco Teórico.....	4
2.1. Propiedades de los materiales constitutivos de la albañilería.....	4
2.1.1. Introducción	4
2.1.2. Materiales	4
2.1.2.1. Unidades.....	4
2.1.2.1.1. Clasificación por clase.....	4
2.1.2.1.2. Clasificación por grados.....	5
2.1.2.1.3. Clasificación por su uso	5
2.1.2.2. Mortero.....	5
2.1.2.3. Albañilería simple como material compuesto	5
2.1.2.4. Albañilería armada	6
2.1.2.5. Albañilería confinada	7
2.2. Elementos Finitos.....	10
2.2.1. Introducción al método de elementos finitos (FEM).....	10
2.2.1.1. Hipótesis de discretización.....	11
2.2.1.2. Teoría de la elasticidad tridimensional.....	12
2.2.1.2.1. Campo de desplazamientos	12
2.2.1.2.2. Campo de deformaciones	13
2.2.1.2.3. Campo de tensiones.....	13
2.2.1.2.4. Relación tensión-deformación.....	14
2.2.1.2.5. Expresión del principio de los trabajos virtuales.....	14
2.2.1.3. Formulación de elementos finitos y funciones de forma.....	14
2.2.1.3.1. Elemento hexaédrico recto de 8 nodos.....	17
2.2.1.3.2. Discretización del campo de desplazamientos	20

2.2.1.3.3. Matriz de deformación	20
2.2.1.3.4. Matriz de rigidez del elemento	22
2.2.1.3.5. Matriz de masa del elemento	23
2.2.1.3.6. Ensamble de matriz de rigidez y de masa	23
3. Métodos	25
3.1. Modelos de muros de albañilería por el MEF en MATLAB	25
3.1.1. Datos de entrada y salida del código	25
3.1.1.1. Solución dinámica en Matlab	26
3.2. Verificación de los modelos numéricos para estructuras de albañilería en Matlab	26
4. Resultados	28
4.1. Validación del modelo de elementos finitos para muros de albañilería	28
4.2. Modelos	29
4.2.1. Modelo 1: Muro de 1m x 1m, empotrado en todo sus extremos	29
4.2.1.1. Resumen de resultados	31
4.2.1.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	32
4.2.2. Modelo 2: Muro de 1m x 1m, empotrado solamente en la base	35
4.2.2.1. Resumen de resultados	37
4.2.2.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	38
4.2.3. Modelo 3: Muro de 2m x 1m, empotrado en todos sus extremos	41
4.2.3.1. Resumen de resultados	43
4.2.3.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	44
4.2.4. Modelo 4: Muro de 2m x 1m, empotrado solamente en la base	47
4.2.4.1. Resumen de resultados	49
4.2.4.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	50
4.2.5. Modelo 5: Muro de 2m x 2m, empotrado en todos sus extremos	53
4.2.5.1. Resumen de resultados	55
4.2.5.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	56

4.2.6. Modelo 6: Muro 2m x 2m, empotrado solamente en la base	59
4.2.6.1. Resumen de resultados	61
4.2.6.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	62
4.2.7. Modelo 7: Muro 2m x 3m, empotrado en todos sus extremos	65
4.2.7.1. Resumen de resultados	67
4.2.7.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	68
4.2.8. Modelo 8: Muro 2m x 3m, empotrado solamente en la base	71
4.2.8.1. Resumen de resultados	73
4.2.8.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab	74
Modelo 2: Muro 1m x 1m, empotrado solamente en la base, discretizado en 299, 2392 y 19136 elementos.....	77
4.2.8.2. Resumen de resultados	77
5. Conclusiones	79
6. Referencias	81
7. Anexos.....	83
7.1. Código programada en Matlab,basado en el método de elementos finitos para analizar el comportamiento dinámico de un muro de albañilería	83
7.1.1. Datos de entrada.....	83
7.1.2. Mallado en 3D para un muro de albañilería	83
7.1.3. Ploteo malla 3D y muro a color.....	87
7.1.4. Clasificación de elementos según sus dimensiones y material	88
7.1.5. Cálculo de Klocal y Mlocal.....	93
7.1.6. Ensamble de Klocal y Mlocal	96
7.1.7. Condiciones de borde asignadas.....	98
7.1.8. Cálculo de Frecuencias y matriz de formas modales	99
7.2. Código programado en Matlab, para calcular las correlaciones de datos de la matriz de modos ocupando el criterio de certeza modal MAC.....	99

Índice de figuras

Figura 2.1: Disposición de ladrillos en la dirección vertical y horizontal.	6
Figura 2.2: Albañilería armada (Manual del albañil del ICH).	7
Figura 2.3: Albañilería confinada (Fuente: Manual del albañil del ICH).	8
Figura 2.4: Albañilería armada (Astroza y Schmidt, 2004).	9
Figura 2.5: Discretización de un continuo en elementos finitos (Chaves y Mínguez, 2010).	11
Figura 2.6: Sólido tridimensional. Vector de desplazamiento de un punto (Celigüeta, 2011).	12
Figura 2.7: Deformación unitaria (Celigüeta, 2011).	13
Figura 2.8: Aproximación de una función por un polinomio lineal (Chaves y Mínguez, 2010).	16
Figura 2.9: Condiciones que han de cumplir las funciones de forma. Caso lineal (Chaves y Mínguez, 2010).	16
Figura 2.10: Elemento hexaédrico (Chaves y Mínguez, 2010).	18
Figura 2.11: Elemento hexaédrico recto de 8 nodos (Chaves y Mínguez, 2010).	19
Figura 2.12: Función de forma del nodo 1 del hexaedro de 8 nodos (Chaves y Mínguez, 2010). ..	19
Figura 4.1: Mallado del muro en Matlab.	29
Figura 4.2: Materiales y sus propiedades asignadas en Matlab.	29
Figura 4.3: Mallado y condiciones de borde en SAP 2000.	30
Figura 4.4: Materiales y sus propiedades asignadas en SAP 2000.	30
Figura 4.5: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	31
Figura 4.6: Modo 1 (SAP).	32
Figura 4.7: Modo 1 (Matlab).	32
Figura 4.8: Modo 2 (SAP).	33
Figura 4.9: Modo 2 (Matlab).	33
Figura 4.10: Mallado del muro en Matlab.	35
Figura 4.11: Materiales y sus propiedades asignadas en Matlab.	35
Figura 4.12: Mallado y condiciones de borde en SAP 2000.	36
Figura 4.13: Materiales y sus propiedades asignadas en SAP 2000.	36
Figura 4.14: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	37
Figura 4.15: Modo 2 (SAP).	38
Figura 4.16: Modo 2 (Matlab).	38
Figura 4.17: Modo 4 (SAP).	39
Figura 4.18: Modo 4 (Matlab).	39
Figura 4.19: Mallado del muro en Matlab.	41
Figura 4.20: Materiales y sus propiedades asignadas en Matlab.	41

Figura 4.21: Mallado y condiciones de borde en SAP 2000.....	42
Figura 4.22: Materiales y sus propiedades asignadas en SAP 2000.	42
Figura 4.23: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	43
Figura 4.24: Modo 8 (SAP).	44
Figura 4.25: Modo 8 (Matlab).	44
Figura 4.26: Modo 6 (SAP).	45
Figura 4.27: Modo 6 (Matlab).	45
Figura 4.28: Mallado del muro en Matlab.	47
Figura 4.29: Materiales y sus propiedades asignadas en Matlab.	47
Figura 4.30: Mallado y condiciones de borde en SAP 2000.....	48
Figura 4.31: Materiales y sus propiedades asignadas en SAP 2000.	48
Figura 4.32: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	49
Figura 4.33: Modo 5 (SAP).	50
Figura 4.34: Modo 5 (Matlab)	50
Figura 4.35: Modo 8 (SAP).	51
Figura 4.36: Modo 8 (Matlab).	51
Figura 4.37: Mallado del muro en Matlab.	53
Figura 4.38: Materiales y sus propiedades asignadas en Matlab.	53
Figura 4.39: Mallado y condiciones de borde en SAP 2000.....	54
Figura 4.40: Materiales y sus propiedades asignadas en SAP 2000.	54
Figura 4.41: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	55
Figura 4.42: Modo 4 (SAP).	56
Figura 4.43: Modo 4 (Matlab).	56
Figura 4.44: Modo 5 (SAP).	57
Figura 4.45: Modo 5 (Matlab).	57
Figura 4.46: Mallado de muro en Matlab.	59
Figura 4.47: Materiales y sus propiedades asignadas en Matlab.	59
Figura 4.48: Mallado y condiciones de borde en SAP 2000.....	60
Figura 4.49: Materiales y sus propiedades asignadas en SAP 2000.	60
Figura 4.50: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	61
Figura 4.51: Modo 6 (SAP).	62
Figura 4.52: Modo 6 (Matlab).	62
Figura 4.53: Modo 4 (SAP).	63
Figura 4.54: Modo 4 (Matlab).	63

Figura 4.55: Mallado del muro en Matlab.	65
Figura 4.56: Materiales y sus propiedades en Matlab.....	65
Figura 4.57: Mallado y condiciones de borde en SAP 2000.....	66
Figura 4.58: Materiales y sus propiedades asignadas en SAP 2000.	66
Figura 4.59: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	67
Figura 4.60: Modo 1 (SAP).	68
Figura 4.61: Modo 1 (Matlab).	68
Figura 4.62: Modo 5 (SAP).	69
Figura 4.63: Modo 5 (Matlab).	69
Figura 4.64: Mallado del muro en Matlab.	71
Figura 4.65: Materiales y sus propiedades en Matlab.....	71
Figura 4.66: Mallado y condiciones de borde en SAP 2000.....	72
Figura 4.67: Materiales y sus propiedades asignadas en SAP 2000.	72
Figura 4.68: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	73
Figura 4.69: Modo 7 (SAP).	74
Figura 4.70: Modo 8 (Matlab).	74
Figura 4.71: Modo 9 (SAP).	75
Figura 4.72: Modo 9 (Matlab).	75
Figura 4.73: Frecuencias de los modos con mayor correlación entre SAP y Matlab.	77

Índice de tablas

Tabla 1: Características de los ladrillos cerámicos (NCh2123.of1997mod2003).....	5
Tabla 2: Coordenadas locales del elemento.	19
Tabla 3: Matriz de modos con mayor correlación de datos, según MAC.	31
Tabla 4: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	31
Tabla 5: Matriz de modos con mayor correlación de datos, según MAC.	37
Tabla 6: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	37
Tabla 7: Matriz de modos con mayor correlación de datos, según MAC.	43
Tabla 8: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	43
Tabla 9: Matriz de modos con mayor correlación de datos, según MAC.	49
Tabla 10: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	49
Tabla 11: Matriz de modos con mayor correlación de datos, según MAC:	55
Tabla 12: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	55
Tabla 13: Matriz de modos con mayor correlación de datos, según MAC.	61
Tabla 14: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	61
Tabla 15: Matriz de modos con mayor correlación de datos, según MAC.	67
Tabla 16: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	67
Tabla 17: Matriz de modos con mayor correlación de datos, según MAC:	73
Tabla 18: Porcentaje de similitud de frecuencias entre modos con mayor correlación.	73
Tabla 19: Porcentaje de similitud de frecuencias entre modos con mayor correlación. n=0 →299 elementos, n=1 →2392 elementos y n=2 → 19136 elementos.	77

1. Introducción

Las construcciones de albañilería son uno de los sistemas constructivos más antiguos y aún se le sigue utilizando ampliamente debido a la durabilidad de este tipo de estructura y al bajo costo asociado a ésta. Algunos ejemplos que dan testimonio del uso y durabilidad de este material son las ruinas de Jericó (Medio oriente, 9500 a. C), las pirámides de Egipto (2500 a. C), la gran muralla China (200 a. C), las pirámides de Yucatán en México (500 d. C), las murallas de piedra de Machu Pichu en Perú (1200 d. C).

En Chile, la albañilería es ampliamente utilizada en la construcción de viviendas. Según el instituto del cemento y del hormigón de Chile (ICH, 2010) aproximadamente dos tercios de las viviendas que se construyen anualmente corresponden a albañilería de ladrillos cerámicos, lo que equivale a alrededor de 70.000 viviendas al año, esto debido a su menor costo y rapidez de construcción, en comparación al hormigón armado y al acero. A pesar de la constante utilización de ese material en la construcción, los ingenieros estructurales no cuentan con el mismo respaldo en términos de investigación que tiene el hormigón y el acero en relación al comportamiento bajo solicitaciones dinámicas, tales como sismos u otras vibraciones impuestas. Esto debido probablemente a la falta de interés que se les asigna, dado que su valor por unidad de vivienda es comparativamente bajo, no obstante, estas en conjunto suman una parte significativa en el mercado inmobiliario.

Si bien, en otros países existen estudios dedicados a este tema, en Chile no existen estudios de este tipo y los resultados obtenidos en el extranjero no son extrapolables a la realidad nacional, porque los materiales utilizados, las normas que rigen su construcción y los métodos constructivos son completamente distintos. Debido a lo mencionado anteriormente y a la constante utilización de la albañilería en nuestro país y éste siendo un país sísmico, se hace necesario hacer más estudios en relación al comportamiento dinámico de muros de albañilería frente a sismos u otras vibraciones impuestas.

Hacer estudios in situ, puede ser bastante costoso y demandar demasiado tiempo, por ello se hace necesario acudir a programas computacionales tales como **RAM ADVANCE**, **ETABS**, **SAP200**, entre muchos otros, para modelar, con el mayor grado de exactitud, dichas estructuras sometidas a registros sísmicos chilenos. Una característica de dichos programas es la imposibilidad de visualizar o modificar el código con el que operan actuando en forma de “**caja negra**”. En base a lo anterior, se crea una herramienta numérica programada en el software Matlab, basada en el método de elementos finitos para analizar el comportamiento dinámico de un muro de albañilería en sus tres

dimensiones. Los resultados serán validados con el software SAP2000, el cual también usa el método de elementos finitos.

Una ventaja de crear esta herramienta numérica en Matlab, es que al modelar un muro de albañilería, solo será necesario ingresar las dimensiones del ladrillo y muro, propiedades mecánicas de los materiales constituyentes de la albañilería y números de elementos en que será discretizado el muro. Lo anterior disminuye notablemente el tiempo de modelado en comparación con SAP, ya que construir un mallado de un paño de albañilería y asignar propiedades en SAP es un proceso lento y laborioso, tomando en cuenta que no todos los muros tendrán siempre el mismo mallado. Además, se reducirá considerablemente el error humano que se pudiese generar al asignar manualmente las propiedades a los materiales del modelo como también el mallado, lo que claramente es muy ventajoso.

1.1. Objetivos

El objetivo general de este proyecto es:

- Desarrollar un código programado en el software Matlab, basado en el método de elementos finitos para analizar el comportamiento dinámico de un muro de albañilería en sus tres dimensiones.

Los objetivos específicos del proyecto son:

- Generar un programa que automáticamente realice el mallado para un muro de albañilería, ingresando las dimensiones del ladrillo y el muro.
- Generar un programa que analice modalmente muros de albañilería, modelado con distintas geometrías y condiciones de borde mediante el método de elementos finitos.
- Validar resultados obtenidos por el código programado en Matlab, con los resultados del software Sap2000.

2. Marco Teórico

2.1. Propiedades de los materiales constitutivos de la albañilería

2.1.1. Introducción

La Albañilería es un material compuesto, formado generalmente por unidades de ladrillos ligados entre sí por medio de algún material de unión, como por ejemplo el mortero, formando así un conjunto estructural resistente. El ladrillo es un material cerámico y es el principal responsable de otorgar las propiedades mecánicas elásticas del material compuesto. Por otro lado, el mortero es una mezcla de agua, cemento y material granular fino, cuya principal función es la de pegar los ladrillos, proporcionando una unión lo suficientemente resistente que evite la falla en ésta. Por lo general, es en el mortero donde se produce la degradación responsable del comportamiento no lineal del material compuesto.

A continuación se caracterizarán los elementos constituyentes de la albañilería y se realizará un resumen de las principales propiedades de la mampostería como material compuesto.

2.1.2. Materiales

2.1.2.1. Unidades

Las unidades componentes de un elemento de albañilería pueden ser, por ejemplo, bloques, ladrillos, rocas, etc.

El ladrillo es un material cerámico hecho con arcilla, su fabricación parte con el moldeo del ladrillo que le da su geometría rectangular, luego de moldear son secados y finalmente cocidos a altas temperaturas. Los ladrillos le proporcionan la rigidez a las estructuras, la que dependerá de la posición de los ladrillos al momento de la construcción.

La norma chilena NCh 169 clasifica los ladrillos en tres tipos.

2.1.2.1.1. Clasificación por clase

- **Ladrillos Macizos MqM:** Unidades macizas sin perforaciones ni huecos.
- **Ladrillos Perforados MqP:** Unidades que poseen perforaciones y huecos, regularmente distribuidos, cuyo volumen es inferior al 50% del volumen bruto o total.
- **Ladrillos Huecos MqH:** Unidades que poseen huecos y perforaciones, regularmente distribuidos, cuyo volumen es mayor o igual al 50% del volumen bruto o total.

2.1.2.1.2. Clasificación por grados

Los ladrillos se clasifican en grado 1, grado 2 y grado 3, según los requisitos de resistencia a la compresión, adherencia y absorción de agua, que se indica en Tabla 1.

Tabla 1: Características de los ladrillos cerámicos (NCh2123.of1997mod2003).

Requisitos mecánicos	Grados de ladrillos cerámicos						
	1		2		3		
	Clases de ladrillos cerámicos						
	MqM	MqP	MqH	MqP	MqH	MqP	MqH
Resistencia a la compresión, mínima (MPa)	15	15	15	11	11	5	5
Absorción de agua, máxima %	14	14	14	16	16	18	18
Adherencia, mínima (MPa) (área neta)	0,4	0,4	0,4	0,35	0,35	0,30	0,25

NOTA - Los requisitos de los ladrillos cerámicos artesanales son tratados en NCh2123.

2.1.2.1.3. Clasificación por su uso

De acuerdo a su uso, los ladrillos cerámicos se clasifican en ladrillos cara vista (V) y ladrillos para ser revestidos (NV).

2.1.2.2. Mortero

El mortero es una mezcla constituida por cemento, arena y eventualmente otro material conglomerante que con adicción de agua reacciona y adquiere resistencia. Su principal función en la albañilería es pegar los ladrillos, pero debe tener la resistencia suficiente para evitar el deslizamiento de los ladrillos una vez fraguado el mortero. También, existe el mortero pre dosificado al cual sólo basta agregar agua para ser utilizado, cumpliendo con todos los requisitos exigidos por las normas chilenas para morteros.

2.1.2.3. Albañilería simple como material compuesto

La albañilería (combinación de mortero y ladrillos) es capaz de resistir compresión y limitados niveles de cortante, la fragilidad es la característica principal de esta materialidad.

Según la norma chilena NCh 1928, la albañilería se define como materia estructural que se obtiene con unidades de albañilería ordenadas en hiladas según un aparejo prefijado y unido con mortero.

El aparejo sirve para formar una estructura resistente. La disposición de los materiales debe ser lo más homogéneo posible. Los ladrillos son colocados uno tras otro pegados con mortero tanto

horizontal como verticalmente. Para el caso vertical su disposición es intercalada, con el fin de que el ladrillo que está en la línea superior, cubra la unión de ladrillos puestos horizontalmente, así se asegura la continuidad de la estructura de albañilería y se evita que el mortero trabaje al corte. Así, solo interactúan esfuerzos que puedan provocar un posible deslizamiento en la junta horizontal. En la Figura 2.1 se clarifica como deben ser colocados los ladrillos.



Figura 2.1: Disposición de ladrillos en la dirección vertical y horizontal.

2.1.2.4. Albañilería armada

La albañilería armada es aquella en la que se refuerza la mezcla de mortero y ladrillo con barras de acero, dispuestas verticalmente y escalerilla que le otorgan una mayor unidad horizontal y colaboran a resistir el corte. Las barras de refuerzo vertical pasan por los agujeros de los ladrillos. La idea es dar una mayor ductilidad a la estructura de albañilería, que este por lo general es sumamente frágil al igual que el hormigón (Galdames, 2014).



Figura 2.2: Albañilería armada (Manual del albañil del ICH).

La albañilería armada se rige según la NCh 1928 (Albañilería Armada – Requisitos para el diseño y cálculo). Esta exige algunas condiciones para el tipo de ladrillo a usar en un muro estructural. Una de las principales es que deben ser ladrillos industriales del tipo 1 ó 2, también que el área neta del ladrillo debe ser mayor al 50% del área bruta.

2.1.2.5. Albañilería confinada

En este caso los paños de albañilería se confinan con pilares y vigas de hormigón armado, la idea es evitar grandes deformaciones laterales y provocar que muros adyacentes se vean perjudicados, generando un efecto en cadena. Es por eso, que al confinar un muro al que se le está aplicando una carga lateral, sólo éste se ve afectado y al momento de fisurarse pueda disipar energía y no perjudique a los muros adyacentes (Galdames, 2014).

La NCh 2123 (Albañilería Confinada – Requisitos de diseño y cálculo), define los requisitos mínimos de los ladrillos para la albañilería confinada, los más importantes son que deben ser ladrillos industriales de los grados 1 ó 2 y clase MqP o MqHv, además de cumplir con los requisitos de resistencia a la compresión, adherencia y absorción indicados en NCh 169.



Figura 2.3: Albañilería confinada (Fuente: Manual del albañil del ICH).

El confinamiento se logra en la medida que los pilares y cadenas enmarquen completamente el paño de albañilería proporcionándole ante acciones contenidas en el plano del muro capacidad de deformación lateral y de disipación de energía una vez que se agrieta el paño.

El comportamiento de un muro de albañilería confinada depende de la calidad de la albañilería, de las dimensiones y de la cuantía de refuerzo de los elementos de confinamiento, del trabajo conjunto que se logre entre el paño de albañilería y los elementos de confinamiento, de la esbeltez del muro y de la existencia de armadura horizontal en el paño de albañilería. Las armaduras de refuerzo de los elementos de confinamiento deben resistir tanto las tracciones que producen los momentos de flexión debido a las acciones que actúan en el plano del muro como los esfuerzos de flexión y corte que producen las cargas perpendiculares al plano del muro (Astroza y Schmidt, 2004).

Los paños de albañilería no deben presentar fallas de aplastamiento por compresión, es por eso que se recomienda unidades macizas o ladrillos perforados cuyas áreas no sobrepasen el 30% del área bruta del ladrillo, también es aconsejable reforzar con más estribos las zonas por donde se pueda propagar un agrietamiento en el paño de albañilería y evitar que esta atraviese el elemento confinador y produzca una rotulación de elementos de confinamiento (Astroza y Schmidt, 2004).

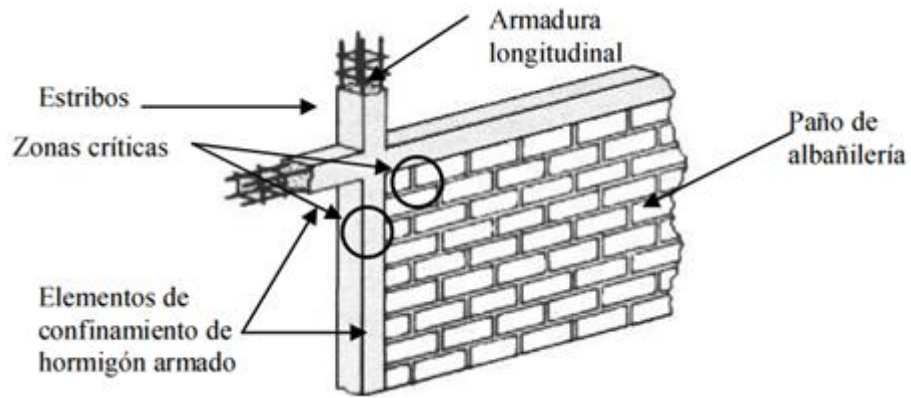


Figura 2.4: Albañilería armada (Astroza y Schmidt, 2004).

A pesar de que la albañilería confinada cumple con los requisitos que exige la norma sísmica chilena (NCh 433), es importante conocer como es su comportamiento bajo cargas dinámicas, además de sus posibles fallas y como se propagan sobre el paño.

2.2. Elementos Finitos

2.2.1. Introducción al método de elementos finitos (FEM)

El principal objetivo de un ingeniero es siempre analizar la realidad para extraer la información más relevante y crear un modelo de cálculo que le permita trabajar. Todo modelo de cálculo se basa en una serie de hipótesis que permiten simplificar el objeto de estudio sin alejarse excesivamente de la realidad. Hasta hace relativamente pocos años, estábamos limitados por el número de variables y elementos que podíamos incluir en un modelo porque no disponíamos de las herramientas de cálculo necesarias. Con la llegada de los ordenadores esto cambió (Portabella, 2014).

Para entender la esencia del método de elementos finitos, debemos comprender que vivimos en una realidad de medios continuos en tres dimensiones.

Si bien las ecuaciones diferenciales que definen el problema continuo pueden ser de fácil deducción para quien tiene algo de conocimiento en la materia, la solución por métodos exactos puede tornarse muy compleja cuando el orden de estas ecuaciones es elevado, consiguiéndose solo para algunas geometrías y condiciones de contorno muy particulares, entonces para resolver de una forma más rápida se acude a métodos numéricos, los cuales entregan un alto nivel de precisión y sofisticación (Becerra, 2016).

El método de elementos finitos es uno de los más usados entre la gran cantidad de métodos numéricos. La idea general del método es la división de un continuo en un conjunto de pequeños elementos interconectados por una serie de puntos llamados nodos. Las ecuaciones que rigen el comportamiento del continuo regirán también el del elemento. De esta forma se consigue pasar de un sistema continuo (infinitos grados de libertad), que es regido por una ecuación diferencial o un sistema de ecuaciones diferenciales, a un sistema con un número de grados de libertad finito cuyo comportamiento sigue siendo gobernado por las mismas ecuaciones diferenciales.

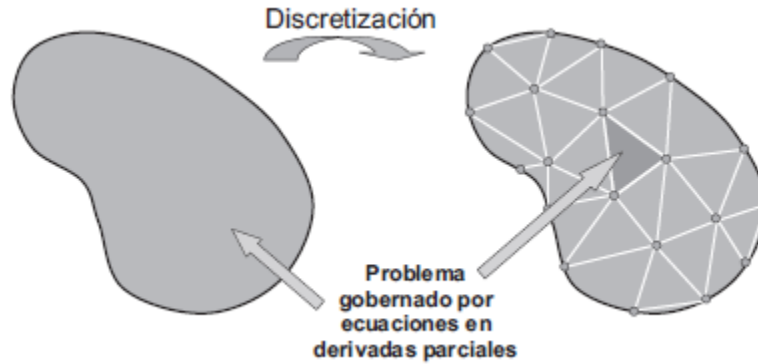


Figura 2.5: Discretización de un continuo en elementos finitos (Chaves y Mínguez, 2010).

2.2.1.1. Hipótesis de discretización

Para un sólido continuo tridimensional, el movimiento de cualquier punto en el espacio queda perfectamente definido por las tres componentes del vector de desplazamientos u .

$$\mathbf{u} = \begin{pmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{pmatrix} \quad (1)$$

Esta función es la solución de la ecuación diferencial que gobierna el problema, que si está bien planteada, cumplirá con las condiciones de contorno impuestas, pero como ya se ha mencionado, esta función u en general no podrá calcularse analíticamente.

Para resolver este problema, el MEF recurre a la hipótesis de discretización, que se basa en lo siguiente (Celigüeta, 2011):

- El continuo se divide por medio de líneas o superficies imaginarias en una serie de regiones contiguas y disjuntas entre sí, de formas geométricas sencillas y normalizadas, llamadas elementos finitos.
- Los elementos finitos se unen entre sí en un número finito de puntos llamados nodos y son regidos por las mismas ecuaciones que rigen al continuo.
- Los desplazamientos de los nodos son las incógnitas fundamentales del problema y determinan la configuración deformada de la estructura. También son conocidos como grados de libertad (GDL) y se consideran independientes.
- Los desplazamientos en cualquier punto de la estructura se determinan a través de funciones de interpolación definidas para cada elemento, cuya función es interpolar las deformaciones de los nodos a cualquier punto dentro del elemento.

- Las funciones de interpolación y los desplazamientos nodales definen el estado de deformación unitaria en el interior del elemento. Estas últimas definen el estado de tensiones en el elemento y en sus bordes a través de las ecuaciones constitutivas.
- Para cada elemento, existe un sistema de fuerzas concentradas en los nodos, que equilibran a las tensiones existentes en el contorno del elemento, y a las fuerzas exteriores que actúan sobre él.

Es importante recalcar que la función solución del problema (u) es aproximada de forma independiente en cada elemento y se apoya en un número finito (y pequeño) de parámetros, que son los valores de dicha función (o a veces su derivada) en los nodos que configuran el elemento. Para una estructura discretizada en varios elementos, pueden utilizarse funciones de interpolación distintas para cada uno de ellos, pero deben cumplir ciertas condiciones de compatibilidad en las fronteras entre elementos (Becerra, 2016).

A continuación se presentarán los conceptos que hay que conocer de la teoría de la elasticidad tridimensional para la utilización del método de los elementos finitos.

2.2.1.2. Teoría de la elasticidad tridimensional

2.2.1.2.1. Campo de desplazamientos

Para un sólido continuo tridimensional, el movimiento de cualquier punto en el espacio queda perfectamente definido por las tres componentes del vector de desplazamientos

$$u = [u(x, y, z), v(x, y, z), w(x, y, z)]^T \quad (2)$$

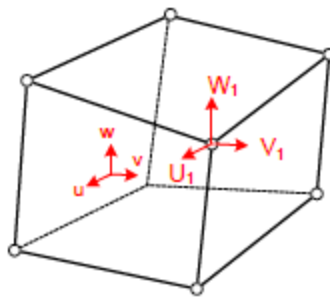


Figura 2.6: Sólido tridimensional. Vector de desplazamiento de un punto (Celigiüeta, 2011).

donde u, v, w son los desplazamientos de cualquier punto según los ejes cartesianos x, y, z respectivamente.

2.2.1.2.2. Campo de deformaciones

Siguiendo la teoría clásica de la elasticidad tridimensional, el vector de deformación en un punto está definido por seis componentes como

$$\varepsilon = [\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{xz}, \gamma_{yz}]^T \quad (3)$$

con

$$\begin{aligned} \varepsilon_x &= \frac{\partial u}{\partial x} ; \varepsilon_y = \frac{\partial v}{\partial y} ; \varepsilon_z = \frac{\partial w}{\partial z} \\ \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} ; \gamma_{xz} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} ; \gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \end{aligned} \quad (4)$$

donde $\varepsilon_x, \varepsilon_y, \varepsilon_z$ son las deformaciones normales y $\gamma_{xy}, \gamma_{xz}, \gamma_{yz}$ las deformaciones tangenciales. Ver Figura 3.3.

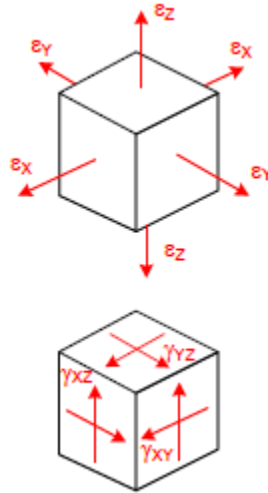


Figura 2.7: Deformación unitaria (Celigieta, 2011).

2.2.1.2.3. Campo de tensiones

El vector de tensiones en un punto contiene seis componentes de tensión, conjugadas con las seis respectivas deformaciones. Así

$$\sigma = [\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{xz}, \tau_{yz}]^T \quad (5)$$

donde $\sigma_x, \sigma_y, \sigma_z$ son las tensiones normales y $\tau_{xy}, \tau_{xz}, \tau_{yz}$ son las tensiones tangenciales.

2.2.1.2.4. Relación tensión-deformación

La ecuación constitutiva que relaciona las tensiones con las deformaciones unitarias, para un material elástico lineal isotrópico, puede escribirse como

$$\sigma = D(\varepsilon - \varepsilon_0) + \sigma_0 \quad (6)$$

y la matriz constitutiva D viene dada por

$$D = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \nu & \nu & 0 & 0 & 0 \\ \nu & 1 & \nu & 0 & 0 & 0 \\ \nu & \nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & e_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & e_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & e_2 \end{bmatrix} \quad (7)$$

con

$$e_1 = \frac{\nu}{(1-\nu)}; e_2 = \frac{1-2\nu}{2(1-\nu)} \quad (8)$$

Se puede ver que la matriz constitutiva D , para un material isotrópico depende sólo de dos parámetros: E el módulo de elasticidad y ν el coeficiente de Poisson, ε_0 es el vector de deformaciones unitarias iniciales existentes en el material y σ_0 el vector de tensiones iniciales presente en el material.

2.2.1.2.5. Expresión del principio de los trabajos virtuales

La expresión integral de equilibrio en problemas de elasticidad tridimensional puede obtenerse haciendo uso del Principio de los Trabajos Virtuales (PTV). Teniendo en cuenta las tensiones y deformaciones que contribuyen al trabajo virtual de la estructura, la expresión PTV puede escribirse en forma matricial como

$$\iiint_V \delta \varepsilon^T \sigma dV = \iiint_V \delta u^T b dV + \iint_A \delta u^T t dA + [\delta a^{(e)}]^T q \quad (9)$$

donde V y A son el volumen y la superficie del cuerpo sobre los que actúan las fuerzas de masa b , de superficie t y puntuales q , respectivamente.

2.2.1.3. Formulación de elementos finitos y funciones de forma

Como se mencionó anteriormente el MEF se basa en discretizar el dominio del problema en un número finito de elementos, tal como indica su nombre. Cada elemento contendrá una porción de información del conjunto, según ubicación en el espacio, y dicha información nos interesará extraerla

o transportarla a los nodos de cada elemento. De esta manera el problema pasa de ser continuo a ser un problema discreto.

Para realizar este proceso, primeramente se han de adoptar unas funciones que aproximen la solución del problema en el subdominio (elemento finito), y gracias a esas funciones de aproximación, usualmente conocidas como funciones de “forma”, se puede representar toda la solución en cada subdominio únicamente empleando los valores de la solución en los nodos. Posteriormente, una vez conocidos los valores de desplazamientos, velocidades, o cualquier otra magnitud en los nodos, a través de las funciones de forma se conocer su valor en cualquier punto de su subdominio (Chaves y Mínguez, 2010).

Se define una función de forma para cada nodo del elemento, por lo que hablaremos de función del nodo i . Cada $N_i^{(e)}$ se caracteriza por valer la unidad en el nodo i y cero en el resto de los nodos.

Las funciones de forma desempeñan un papel clave en el MEF ya que permite:

- Trasladar el comportamiento a lo largo del elemento a los nodos.
- Aproximar los valores de la función en todo el dominio una vez conocidos los valores nodales.

Por estos motivos se van a emplear como funciones de forma de aproximación los *polinomios de Lagrange*. Para ello, supóngase una función genérica $f = f(x)$, tal y como se muestra en la Figura 2.8. Ésta puede aproximarse en el intervalo (x_a, x_b) mediante un polinomio lineal, cuya expresión puede definirse de la siguiente manera:

$$\begin{aligned} f(x) \approx q(x) &= \frac{x_b - x}{l^{(e)}} f_a + \frac{x - x_a}{l^{(e)}} f_b \\ &= N_1 f_a + N_2 f_b, \end{aligned} \tag{10}$$

donde f_a, f_b son los valores conocidos de la función $f(x)$ en los puntos x_a y x_b , respectivamente, y N_1 y N_2 son las denominadas funciones de forma.

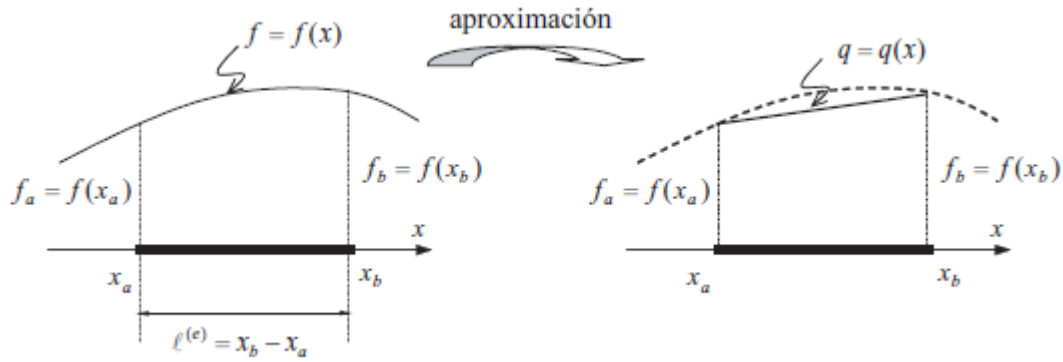


Figura 2.8: Aproximación de una función por un polinomio lineal (Chaves y Mínguez, 2010).

Las funciones de forma, están vinculadas a cada nodo del elemento y han de cumplir dos condiciones:

1. El valor de la función de forma asociada a un nodo correcto es uno en ese nodo y cero en los demás, tal y como se muestra en la Figura 2.9, es decir:

$$N_i(x_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases} \quad (11)$$

2. La suma de las funciones de forma para un elemento es igual a 1 en todo el dominio (ver Figura 2.14), es decir:

$$\sum_{i=1}^n N_i = 1 \quad (12)$$

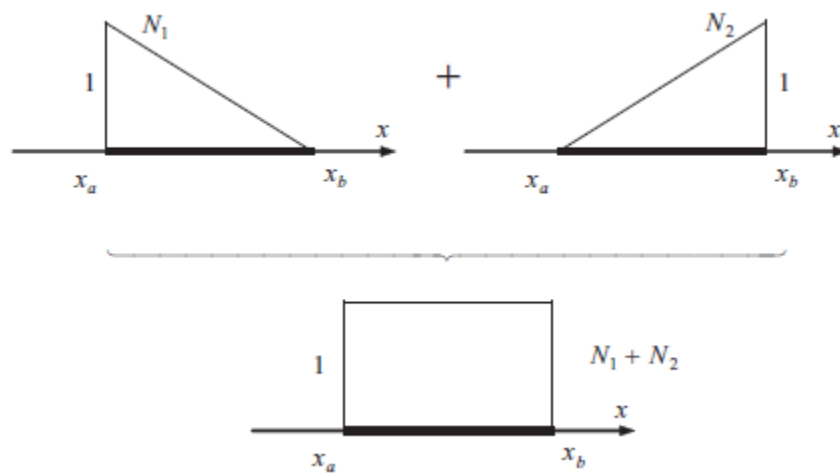


Figura 2.9: Condiciones que han de cumplir las funciones de forma. Caso lineal (Chaves y Mínguez, 2010).

Para el empleo posterior de las funciones de forma dentro del método de elementos finitos, es de gran interés práctico que el intervalo en el que se aproximan las funciones este normalizado. De esa manera la integración numérica en el dominio del elemento puede obtenerse fácilmente incluso cuando las funciones a integrar sean complicadas. Hay que tener en cuenta que con la geometría normalizada siempre se emplean los mismos puntos de integración y los mismos pesos. Dado que el intervalo en el espacio normalizado puede ser cualquiera, en este documento se adopta el intervalo $[-1,1]$ como intervalo de referencia (Chaves y Mínguez, 2010).

Para llevar a cabo la normalización considérese un segmento de recta delimitado por el intervalo $[x_a, x_b]$, y el segundo normalizado. Aplicando semejanza de triángulos se tiene:

$$\frac{l^{(e)}}{2} = \frac{x - \frac{(x_a + x_b)}{2}}{\xi}. \quad (13)$$

Teniendo en cuenta que $l^{(e)} = x_b - x_a$ y reestructurando la expresión anterior se obtiene:

$$x(\xi) = \frac{x_b + x_a}{2} + \frac{x_b - x_a}{2} \xi$$

$$x(\xi) = \frac{1}{2}(1 - \xi)x_a + \frac{1}{2}(1 + \xi)x_b \quad (14)$$

$$x(\xi) = N_1(\xi)x_a + N_2(\xi)x_b,$$

expresión en la que la función de forma son función de la coordenada natural ξ .

Lo mencionado anteriormente fue a modo de ejemplo para entender el objetivo de las funciones de forma, ya que fue mostrado para el caso de elementos unidimensionales.

A continuación se exponen las funciones de forma para el elemento tridimensional hexaédrico recto de 8 nodos, el cual se utilizó durante esta investigación.

2.2.1.3.1. Elemento hexaédrico recto de 8 nodos.

Para su descripción adoptaremos un sistema de coordenadas naturales ξ, η, ζ . Las caras del elemento están en $\xi = \pm 1$, $\eta = \pm 1$, $\zeta = \pm 1$ como se muestra en la Figura 2.11. El elemento de aristas $2a \times 2b \times 2c$ en los ejes ξ , η y ζ respectivamente.

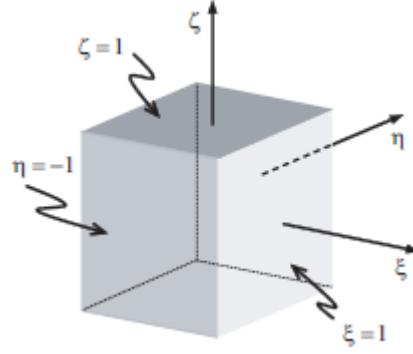


Figura 2.10: Elemento hexaédrico (Chaves y Mínguez, 2010).

Para los elementos hexaédricos rectos las funciones de forma se obtienen combinando los polinomios de Lagrange en cada una de las tres direcciones (ξ, η, ζ), tal como se muestra en la Figura 2.13.

Para el caso concreto del nodo 1 se obtiene mediante el producto del polinomio de Lagrange lineal unidimensional según cada una de las tres direcciones, tal y como se muestra en la Figura 2.18.

$$N_1(\xi, \eta, \zeta) = N_1^{(1)}(\xi)N_1^{(1)}(\eta)N_1^{(1)}(\zeta) \quad (15)$$

$$N_1(\xi, \eta, \zeta) = \frac{1}{2}(1 - \xi)\frac{1}{2}(1 - \eta)\frac{1}{2}(1 - \zeta),$$

que simplificado queda como:

$$N_1(\xi, \eta, \zeta) = \frac{1}{8}(1 - \xi)(1 - \eta)(1 - \zeta), \quad (16)$$

la expresión anterior se puede generalizar para el nodo i donde

$$N_i^{(e)} = \frac{1}{8}(1 + \xi_i\xi)(1 + \eta_i\eta)(1 + \zeta_i\zeta) \quad (17)$$

Tabla 2: Coordenadas locales del elemento.

Número local del nodo	Coordenadas locales		
	ξ_i	η_i	ζ_i
1	-1	-1	-1
2	1	-1	-1
3	1	1	-1
4	-1	1	-1
5	-1	-1	1
6	1	-1	1
7	1	1	1
8	-1	1	1

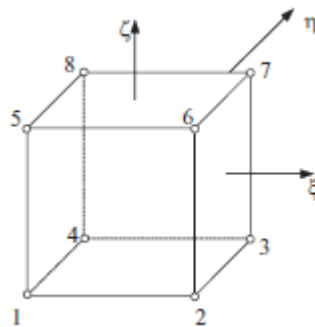


Figura 2.11: Elemento hexaédrico recto de 8 nodos (Chaves y Mínguez, 2010).

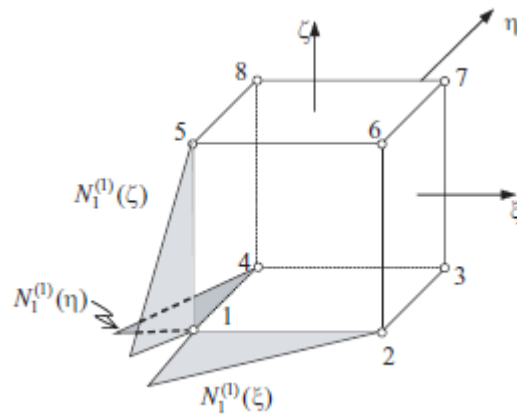


Figura 2.12: Función de forma del nodo 1 del hexaedro de 8 nodos (Chaves y Mínguez, 2010).

2.2.1.3.2. Discretización del campo de desplazamientos

Consideremos un sólido tridimensional discretizado en elementos hexaédricos rectos de 8 nodos. El campo de desplazamientos en el interior del elemento puede aproximarse por

$$\mathbf{u} = \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \begin{Bmatrix} N_1 u_1 + N_2 u_2 + N_3 u_3 + N_4 u_4 + N_5 u_5 + N_6 u_6 + N_7 u_7 + N_8 u_8 \\ N_1 v_1 + N_2 v_2 + N_3 v_3 + N_4 v_4 + N_5 v_5 + N_6 v_6 + N_7 v_7 + N_8 v_8 \\ N_1 w_1 + N_2 w_2 + N_3 w_3 + N_4 w_4 + N_5 w_5 + N_6 w_6 + N_7 w_7 + N_8 w_8 \end{Bmatrix}$$

$$\mathbf{u} = \sum_{i=1}^8 N_i \mathbf{a}_i^{(e)} = \mathbf{N} \mathbf{a}^{(e)} \quad (18)$$

$$\mathbf{u} = \mathbf{N} \mathbf{a}^{(e)}$$

donde

$$\mathbf{N} = [\mathbf{N}_1, \mathbf{N}_2, \mathbf{N}_3, \mathbf{N}_4, \mathbf{N}_5, \mathbf{N}_6, \mathbf{N}_7, \mathbf{N}_8]; \mathbf{N}_i = \begin{bmatrix} N_i & 0 & 0 \\ 0 & N_i & 0 \\ 0 & 0 & N_i \end{bmatrix} \quad (19)$$

y

$$\mathbf{a}^{(e)} = \begin{Bmatrix} a_1^{(e)} \\ a_2^{(e)} \\ a_3^{(e)} \\ a_4^{(e)} \\ a_5^{(e)} \\ a_6^{(e)} \\ a_7^{(e)} \\ a_8^{(e)} \end{Bmatrix}; \mathbf{a}_i^{(e)} = \begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} \quad (20)$$

son, respectivamente la matriz de funciones de forma y el vector de desplazamiento del elemento y de un nodo.

2.2.1.3.3. Matriz de deformación

Sustituyendo (18) en la expresión del vector de deformaciones (3), se obtiene para el elemento hexaédrico de 8 nodos

$$\varepsilon = \sum_{i=1}^8 \left\{ \begin{array}{l} \frac{\partial N_i}{\partial x} u_i \\ \frac{\partial N_i}{\partial y} v_i \\ \frac{\partial N_i}{\partial z} w_i \\ \frac{\partial N_i}{\partial y} u_i + \frac{\partial N_i}{\partial x} v_i \\ \frac{\partial N_i}{\partial z} u_i + \frac{\partial N_i}{\partial x} w_i \\ \frac{\partial N_i}{\partial x} v_i + \frac{\partial N_i}{\partial y} w_i \end{array} \right\} = \sum_{i=1}^8 B_i a_i^{(e)} = \mathbf{B} \mathbf{a}^{(e)} \quad (21)$$

$$\boldsymbol{\varepsilon} = \mathbf{B} \mathbf{a}^{(e)}$$

donde \mathbf{B} es la matriz de deformación

$$\mathbf{B} = [\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{B}_5, \mathbf{B}_6, \mathbf{B}_7, \mathbf{B}_8] \quad (22)$$

siendo \mathbf{B}_i la matriz de deformación del nodo i , con

$$\mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & 0 \\ 0 & \frac{\partial N_i}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} & 0 \\ \frac{\partial N_i}{\partial z} & 0 & \frac{\partial N_i}{\partial x} \\ 0 & \frac{\partial N_i}{\partial x} & \frac{\partial N_i}{\partial y} \end{bmatrix} \quad (23)$$

Por ser el elemento recto, las derivadas cartesianas de las funciones de forma se pueden calcular directamente por la expresión:

$$\frac{\partial N_i}{\partial x} = \frac{1}{a} \frac{\partial N_i}{\partial \xi} ; \frac{\partial N_i}{\partial y} = \frac{1}{b} \frac{\partial N_i}{\partial \eta} ; \frac{\partial N_i}{\partial z} = \frac{1}{c} \frac{\partial N_i}{\partial \zeta} \quad (24)$$

2.2.1.3.4. Matriz de rigidez del elemento

La variación virtual del campo de deformaciones δu y de las deformaciones unitarias $\delta \varepsilon$ pueden escribirse en función de las matrices N y B dada en las ecuaciones (19) y (22) respectivamente.

$$\delta u^T = [\delta a^{(e)}]^T N^T ; \delta \varepsilon^T = [\delta a^{(e)}]^T B^T \quad (25)$$

Sustituyendo la ecuación (25) en (9), se obtiene la ecuación de equilibrio aproximada mediante la hipótesis de interpolación de funciones.

$$\iiint_{V^{(e)}} B^T \sigma dV - \iiint_{V^{(e)}} N^T b dV - \iint_{A^{(e)}} N^T t dA = q^{(e)} \quad (26)$$

La ec. (26) expresa el equilibrio entre las fuerzas nodales de equilibrio y las fuerzas debidas a la deformación del elemento (primera integral), las fuerzas másicas (segunda integral) y las de superficie (tercera integral). Sustituyendo ahora el vector de tensiones σ por su valor en función de los desplazamientos nodales utilizando la forma más general de la ec. (27),

$$\sigma = D\varepsilon = DBa^{(e)} \quad (27)$$

se obtiene

$$\iiint_{V^{(e)}} B^T DBa^{(e)} dV - \iiint_{V^{(e)}} N^T b dV - \iint_{A^{(e)}} N^T t dA = q^{(e)} \quad (28)$$

o

$$K^{(e)} a^{(e)} - f^{(e)} = q^{(e)} \quad (29)$$

donde

$$K_{24 \times 24}^{(e)} = \iiint_V B_{24 \times 6}^T D_{6 \times 6} B_{6 \times 24} dv \quad (30)$$

es la matriz de rigidez del elemento, y

$$f^{(e)} = f_b^{(e)} + f_t^{(e)} \quad (31)$$

el vector de fuerzas nodales equivalente del elemento, siendo

$$f_b^{(e)} = \iiint_{V^{(e)}} N^T b dV \quad (32)$$

$$f_t^{(e)} = \iint_{A^{(e)}} N^T t dA \quad (33)$$

Para un elemento diferencial de volumen de aristas $2ax2bx2c$, la relación del volumen en coordenadas cartesianas y naturales (se utiliza una formulación isoparamétrica para el cambio de sistema de coordenadas) viene dado por:

$$dxdydz = abcd\xi d\eta d\zeta \quad (34)$$

Por lo tanto, el cálculo de la matriz de rigidez del elemento en coordenadas naturales queda definido como sigue

$$K_{24 \times 24}^{(e)} = \int_{-1}^{-1} \int_{-1}^1 \int_{-1}^1 B_{24 \times 6}^T D_{6 \times 6} B_{6 \times 24} abcd\xi d\eta d\zeta \quad (35)$$

2.2.1.3.5. Matriz de masa del elemento

Para el elemento hexaédrico recto de ocho nodos, la matriz de masa local o del elemento queda definida por la ec. (36),

$$M_{24 \times 24}^{(e)} = \iiint_V \rho [N_{24 \times 3}^T N_{3 \times 24}] dv \quad (36)$$

donde ρ , corresponde a la densidad del material.

Reemplazando (34) en (36), la matriz de masa en coordenadas naturales queda como sigue

$$M_{24 \times 24}^{(e)} = \int_{-1}^{-1} \int_{-1}^1 \int_{-1}^1 \rho [N_{24 \times 3}^T N_{3 \times 24}] abcd\xi d\eta d\zeta \quad (37)$$

2.2.1.3.6. Ensamble de matriz de rigidez y de masa

Las matrices de rigidez y masa de un elemento ($K_i^{(e)}$ y $M_i^{(e)}$) tienen tantas filas y columnas como GDL tenga el elemento, por ejemplo, si un elemento tridimensional de ocho nodos tiene 3 GDL por nodo, dichas matrices serán de 24×24 , donde cada fila y columna está asociada a un GDL. Dado

que se requiere obtener el comportamiento de la estructura completa (formada por un número finitos de elementos), cada matriz elemental se deberá ensamblar en la matriz del sistema $K^{(global)}$ y $M^{(global)}$.

El muro estudiado en esta investigación fue discretizado con elementos hexaédricos rectos de ocho nodos y a cada nodo se le asignaron tres GDL, correspondientes a desplazamientos. Básicamente se enumeró cada nodo del muro (y se identificó los ocho nodos correspondientes a cada elemento) y como se mencionó anteriormente, en cada nodo se asignaron tres GDL (Ej. **Nodo₁** → *gdl 1, gdl 2 y gdl 3*, **Nodo₂** → *gdl 4, gdl 5 y gdl 6*,, **Nodo_n** → *gdl 3 * (n - 2), gdl 3 * (n - 1) y gdl (3 * n)*). Según la enumeración dada, a cada elemento de ocho nodos le corresponden 24 GDL ya identificados y guardados en un vector, comúnmente conocido como vector de conectividad. Este vector de conectividad define la posición que tomará cada componente de la matriz $K^{(e)}$ en la matriz general del sistema o matriz ensamblada $K^{(global)}$. Idénticamente el mismo procedimiento para ensamblar la matriz de masa de cada elemento en la matriz general del sistema o matriz ensamblada $M^{(global)}$.

3. Métodos

El modelado a través de elementos finitos es un método numérico de fácil aplicación mediante la utilización de matrices, pero la solución manual puede tornarse muy difícil cuando estas matrices tienen dimensiones muy elevadas, es por esto que se utilizan diferentes herramientas de cálculo numérico que ayudan al ingeniero con las operaciones del álgebra lineal. Una de estas herramientas es el software MATLAB cuyo nombre proviene de MATrix LABoratory (Laboratorio de matrices) y es un programa muy potente, que incluye herramientas de cálculo científico y técnico y de visualización gráfica, así como un lenguaje de programación propio de alto nivel (M) (Becerra, 2016).

3.1. Modelos de muros de albañilería por el MEF en MATLAB

Se creó una rutina en el software Matlab (ver código en anexos del presente trabajo), que fuese capaz de con tan solo ingresar las dimensiones del ladrillo, espesor del mortero y dimensiones del muro, generar automáticamente un mallado del muro. Además, este mallado debe ser capaz de generar los nodos que definen a cada elemento del muro e identificar a que material corresponde cada elemento (ladrillo, mortero o un material interface). Otra característica que tiene este modelo, es que el mallado depende de la variable “ n ”, la cual subdivide cada elemento del muro en 8^n elementos.

3.1.1. Datos de entrada y salida del código

Las propiedades que se deben asignar al código son:

1. **Propiedades geométricas:** Largo y alto del muro (m), largo, alto y ancho del ladrillo (m) y espesor del mortero.
2. **Propiedades mecánicas:** Modulo de elasticidad (E), módulo de Poisson (ν) y densidad (ρ), de cada material.
3. **Condiciones de borde:** Empotrado sólo en un lado, mixto o en todos sus extremos.

Los materiales a asignar al código están compuestos por tres tipos: ladrillo, mortero y un material de interfase. Este material de interfase está ubicado en la mitad de cada ladrillo que compone el muro, tiene como función modelar el agrietamiento del ladrillo, que pudiese ocurrir en la mitad de éste cuando el muro falla por corte, por eso este material de interfase tiene menor resistencia que el ladrillo, generando así que este material falle antes que el ladrillo, de manera que represente las líneas de agrietamiento del muro. Para mayor información sobre este material interfase, ver (Gambarotta y Lagomarsino, 1997).

Como el muro tiene distintos materiales con distintas propiedades mecánicas, la rutina debe ser capaz de identificar la ubicación de cada material dentro de la configuración del muro y generar matrices de rigidez y masa para cada material.

3.1.1.1. Vibración libre en Matlab

La solución dinámica, se logra mediante la utilización de la función `eigs`, cuya notación es:

$$[V, D]=eigs(Kglobal,Mglobal,10,'sm')$$

Esta función determina los valores y vectores propios (V y D), que son las soluciones no triviales del sistema $(K - V * M) * D = 0$. Se deduce de la ecuación de equilibrio dinámico para problemas de vibraciones libres, que D corresponde al vector de desplazamientos ϕ y V a la frecuencia elevada al cuadrado ω^2 . En resumen, los datos de salidas son las matrices de frecuencias y los modos de vibrar, además de graficas que muestran los desplazamientos por modo.

3.2. Verificación de los modelos numéricos para estructuras de albañilería en Matlab

Debido a las numerosas técnicas disponibles para modelación de estructuras y el fácil acceso a los softwares comerciales, existe una constante preocupación en el desarrollo de las técnicas para determinar si los datos de salidas que entregan los modelos, concuerdan con el comportamiento real de la estructura en estudio. En este estudio se verificarán los resultados entre dos softwares.

El primer paso en la verificación modal, es comparar los resultados de la predicción numérica con la respuesta experimental de la estructura, identificando las similitudes y diferencias. La aproximación más común es contrastar las propiedades modales experimentales (frecuencia modal, rigidez, amortiguación modal y modos de vibrar), con las propiedades equivalentes obtenidas desde el análisis numérico.

Estas similitudes y diferencias deben ser expresadas en orden para evaluar los grados de similitud y discrepancia entre el modelo y la realidad. Para este propósito, se han desarrollado una serie de factores de correlación. El factor más usado es el criterio de certeza modal (MAC, por su sigla en inglés), que se denota de la siguiente forma

$$MAC_{ij} = \frac{|\{\phi_i^a\}^T \{\phi_j^x\}|^2}{(\{\phi_i^a\}^T \{\phi_i^a\})(\{\phi_j^x\}^T \{\phi_j^x\})} \quad 0 \leq MAC_{ij} \leq 1 \quad (38)$$

donde $\{\phi_i^a\}$ representa al modo i analítico determinado por el modelo numérico y $\{\phi_j^x\}$ corresponde al modo j experimental extraído desde las mediciones. Usualmente, los factores de MAC son expresados matricialmente, donde su diagonal principal se espera que tenga valores cercanos a uno, mientras que los otros valores debiesen ser cero si los modos han sido correctamente emparejados.

En esta tesis usaremos el MAC para contrastar los resultados entregados por ambos softwares (Matlab y SAP2000). Se realizó una rutina en Matlab, capaz de calcular las correlaciones de datos de la matriz de modos ocupando el criterio de certeza modal MAC. Esta rutina compara los diez primeros modos entregados por los softwares. Los resultados son almacenados en una matriz de 10x10.

Se aprovechó el MAC para comparar los resultados entregados por ambos software (no hay resultados experimentales).

4. Resultados

4.1. Validación del modelo de elementos finitos para muros de albañilería

A partir de los resultados obtenidos del código Matlab, donde los output son las frecuencias naturales de la estructura (ω) y la matriz de modos de vibrar (ϕ), se procedió a validar estos resultados con los entregados por el software Sap2000 (en ambos softwares se modeló considerando la misma geometría, propiedades mecánicas de los materiales y condiciones de borde), donde se comparó la similitud de las frecuencias naturales, para cada muro de estudio expresada en porcentajes y los modos de vibrar de las estructuras, comparando la matriz de modos normalizados obtenidos en Matlab, versus la matriz de desplazamientos obtenidos del software Sap2000. El método utilizado para comparar los modos se basa en el criterio de certeza modal (*MAC*), cuyos resultados fueron expresados en porcentajes dentro de la matriz de correlación de modos. Cabe señalar que los valores cercanos a 100%, significa que tienen una muy buena correlación de datos, por ende los modos obtenidos serán muy similares entre ambos modelos.

Para validar los resultados del código programado en Matlab, se analizaron ocho tipos de muros de albañilería, donde se ocuparon las mismas propiedades mecánicas de los materiales, pero variando la geometría y condiciones de borde de los muros.

Las geometrías y condiciones de bordes de los muros utilizados son:

- Muro de 1m x 1m, empotrado en todos sus extremos.
- Muro de 1m x 1m, empotrado solamente en la base.
- Muro de 2m x 1m, empotrado en todos sus extremos.
- Muro de 2m x 1m, empotrado solamente en la base.
- Muro de 2m x 2m, empotrado en todos sus extremos.
- Muro de 2m x 2m, empotrado solamente en la base.
- Muro de 2m x 3m, empotrado en todos sus extremos.
- Muro de 2m x 3m, empotrado solamente en la base.

4.2. Modelos

4.2.1. Modelo 1: Muro de 1m x 1m, empotrado en todo sus extremos

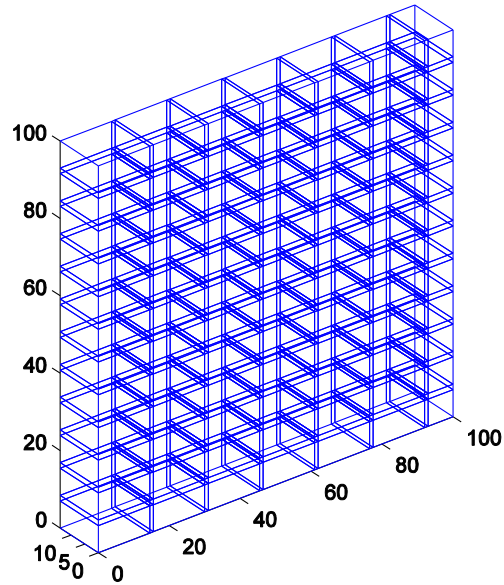


Figura 4.1: Mallado del muro en Matlab.

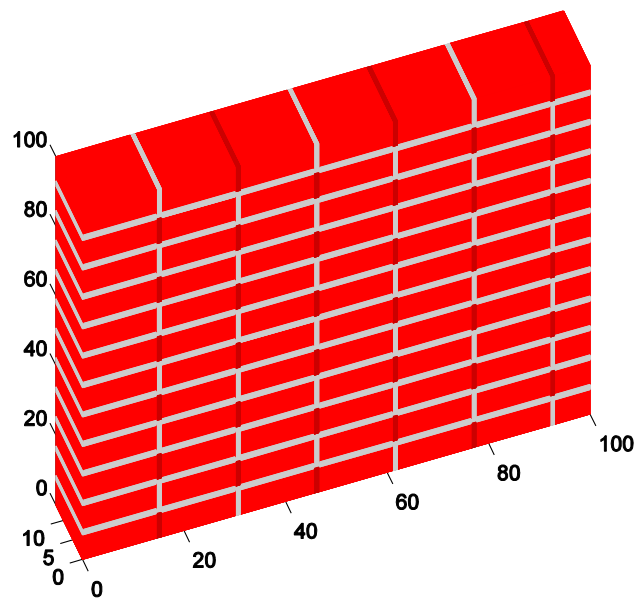


Figura 4.2: Materiales y sus propiedades asignadas en Matlab.

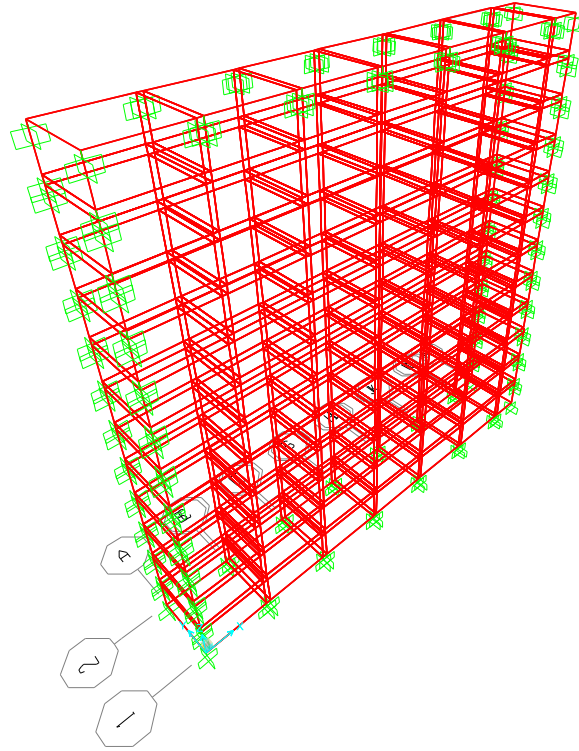


Figura 4.3: Mallado y condiciones de borde en SAP 2000.

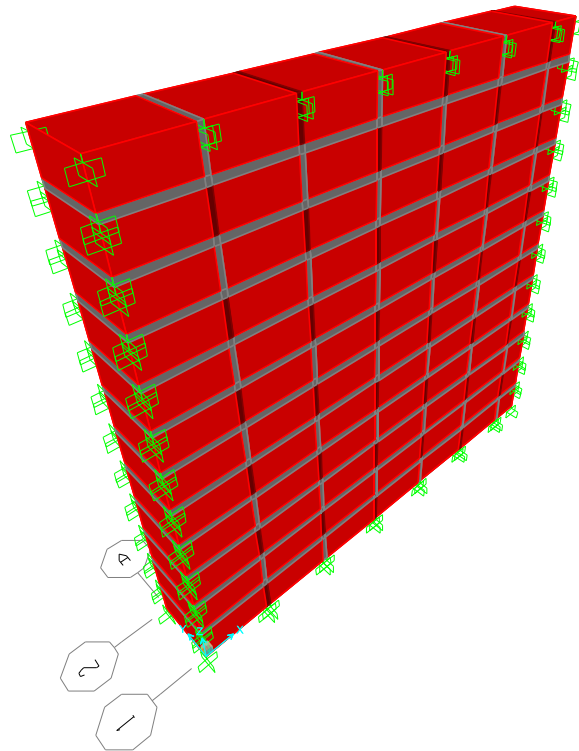


Figura 4.4: Materiales y sus propiedades asignadas en SAP 2000.

4.2.1.1. Resumen de resultados

Tabla 3: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	99.99%	0.00%	0.00%	0.00%	0.00%	0.10%	0.00%	0.00%	0.00%	0.27%
SAP 2	0.00%	99.94%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 3	0.00%	0.00%	99.90%	0.00%	0.00%	0.00%	0.00%	0.00%	0.10%	0.00%
SAP 4	0.00%	0.00%	0.00%	0.00%	99.78%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 5	0.00%	0.00%	0.00%	99.98%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 6	0.00%	0.00%	0.00%	0.00%	0.00%	99.61%	0.00%	0.00%	0.00%	0.07%
SAP 7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.99%	0.00%	0.00%	0.00%
SAP 8	0.00%	0.00%	0.02%	0.00%	0.00%	0.01%	0.00%	0.00%	99.28%	0.03%
SAP 9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	98.99%	0.00%	0.00%
SAP 10	0.01%	0.00%	0.01%	0.00%	0.00%	0.03%	0.00%	0.00%	0.03%	98.89%

Tabla 4: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SIMILITUD		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	2302.65	1	2404.11	1	1	95.78%
2	3799.50	2	4072.18	2	2	93.30%
3	4460.89	3	4942.13	3	3	90.26%
4	5398.35	4	5817.35	5	4	98.35%
5	5721.63	5	6106.30	4	5	88.41%
6	5834.35	6	6420.45	6	6	90.87%
7	6453.64	7	6544.01	7	7	98.62%
8	6907.47	8	7380.25	9	8	93.59%
9	7048.63	9	7998.93	8	9	88.12%
10	7172.95	10	8535.12	10	10	84.04%

(A)

(B)

(C)

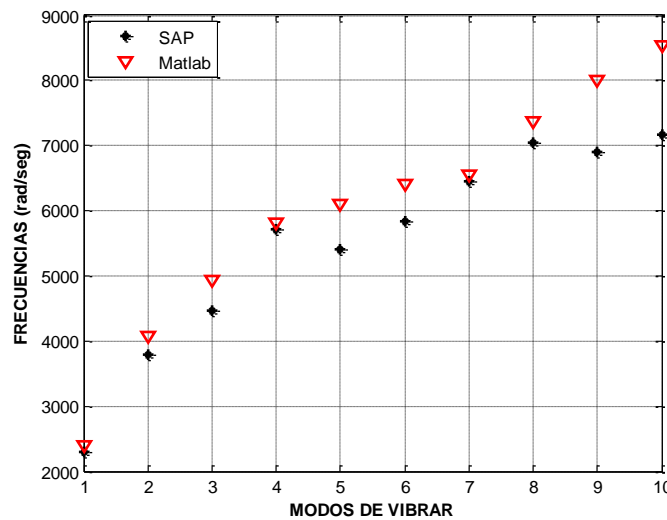


Figura 4.5: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.1.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

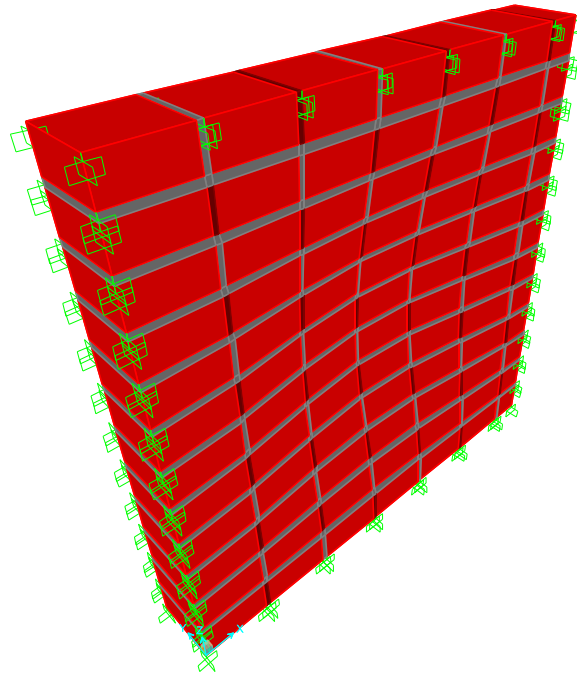


Figura 4.6: Modo 1 (SAP).

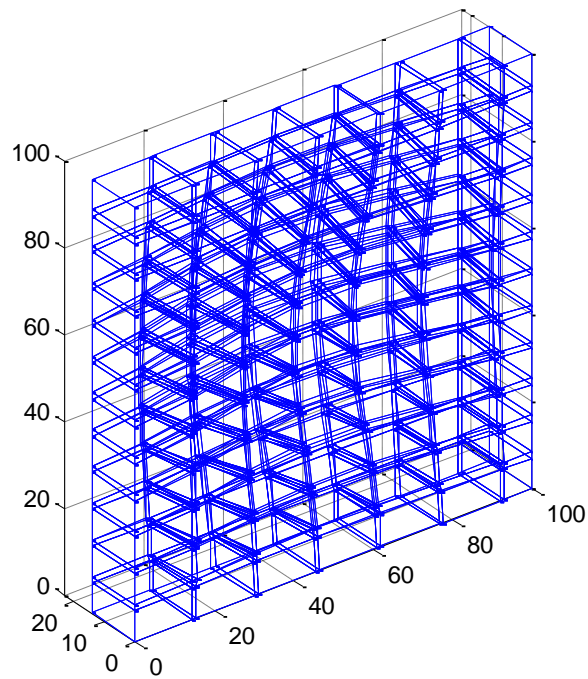


Figura 4.7: Modo 1 (Matlab).

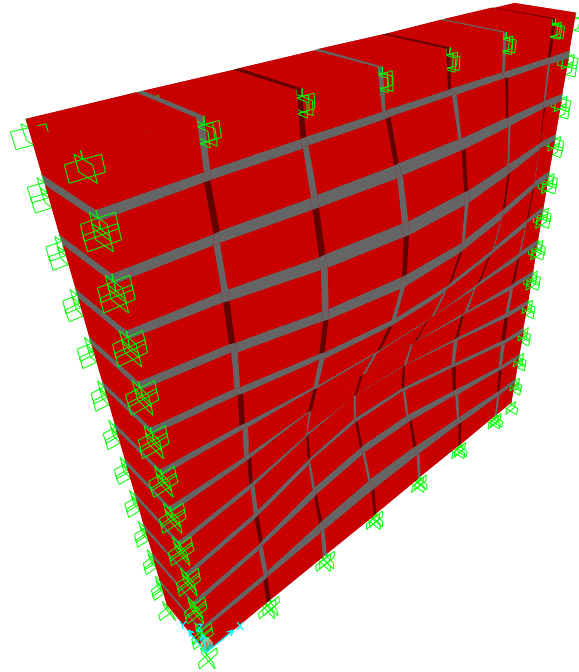


Figura 4.8: Modo 2 (SAP).

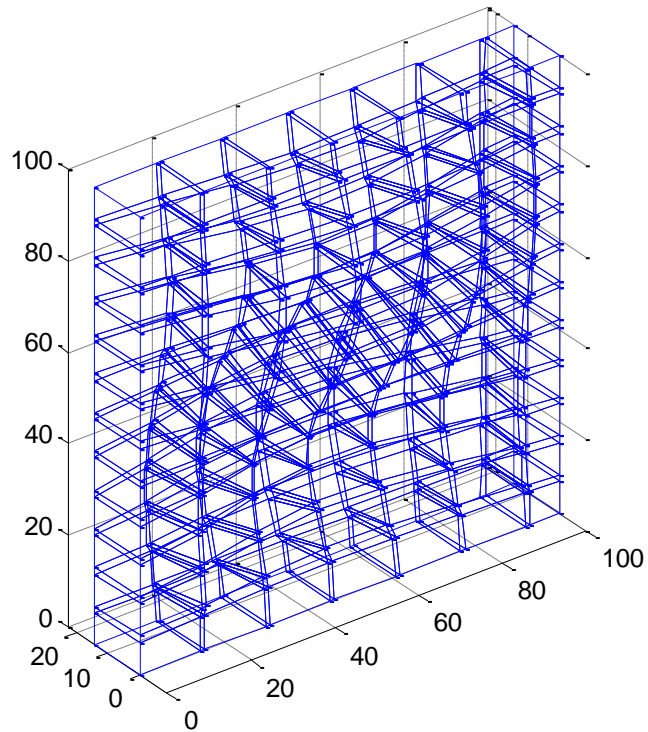


Figura 4.9: Modo 2 (Matlab).

Se realizó una rutina en Matlab, capaz de calcular las correlaciones de datos de la matriz de modos ocupando el criterio de certeza modal *MAC*. Esta rutina compara los 10 primeros modos entregados por ambos software. Los resultados son almacenados en una matriz de 10x10, donde se pueden identificar los modos que tiene mayor correlación de datos. La Tabla 3 corresponde a la matriz de correlación de datos según *MAC*, donde por lo general, la diagonal principal concentra los modos con mayor correlación de datos. Por otro lado, la Tabla 4 muestra el porcentaje de similitud de las frecuencias entre los modos con mayor correlación. En la Figura 4.5 se puede observar la diferencia entre las frecuencias, asociadas a los modos con mayor correlación de datos.

Los resultados entregados, una vez analizados ambos modelos, son muy buenos. Se puede observar en la Tabla 3 que las correlaciones de los datos de cada modo varían entre 98.89% y 99.99%. Para el caso de las frecuencias, el porcentaje de similitud de los modos que coinciden son buenos, donde se obtuvieron porcentajes por sobre el 85 % para 9 modos, de los cuales destaca el modo 7, donde se alcanzó una similitud del 98.6 %.

Una de las características del código Matlab es entregar una gráfica en 3D de los modos de vibrar. A modo de ejemplo, las Figuras 4.6- 4.7 y 4.8 -4.9, muestran las comparaciones visuales de los modos 1 y 2 respectivamente, entre ambos softwares.

4.2.2. Modelo 2: Muro de 1m x 1m, empotrado solamente en la base

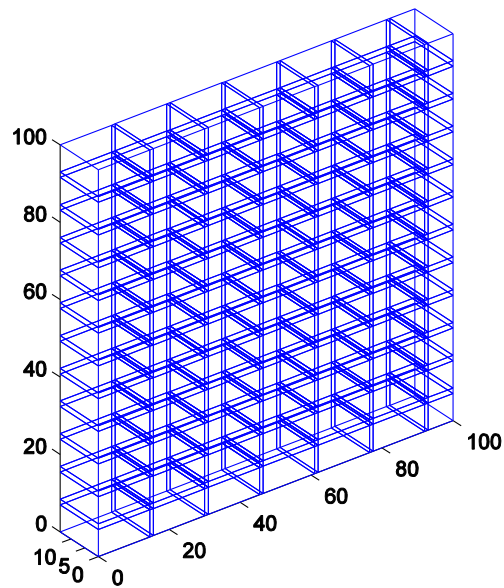


Figura 4.10: Mallado del muro en Matlab.

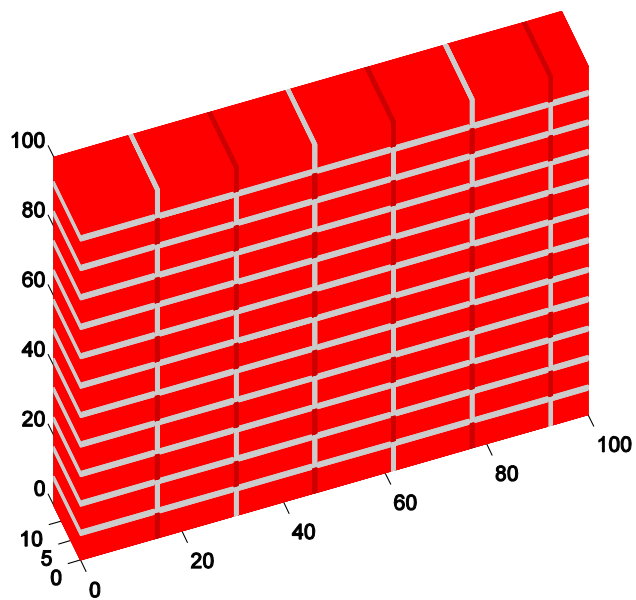


Figura 4.11: Materiales y sus propiedades asignadas en Matlab.

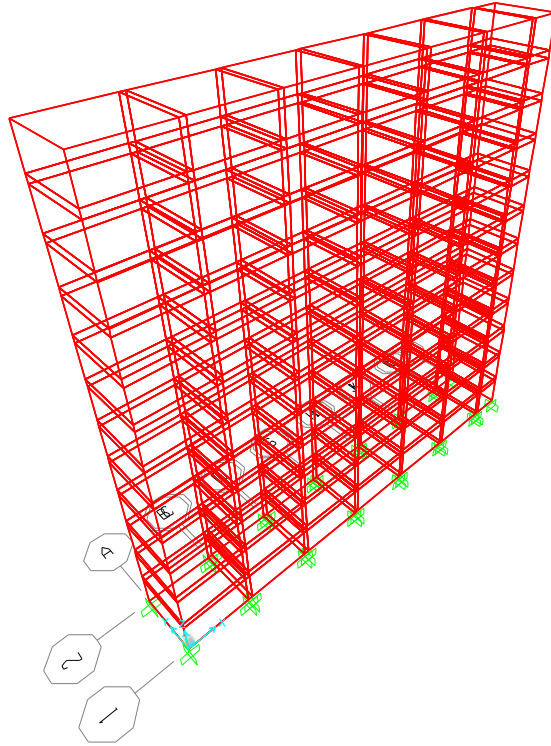


Figura 4.12: Mallado y condiciones de borde en SAP 2000.

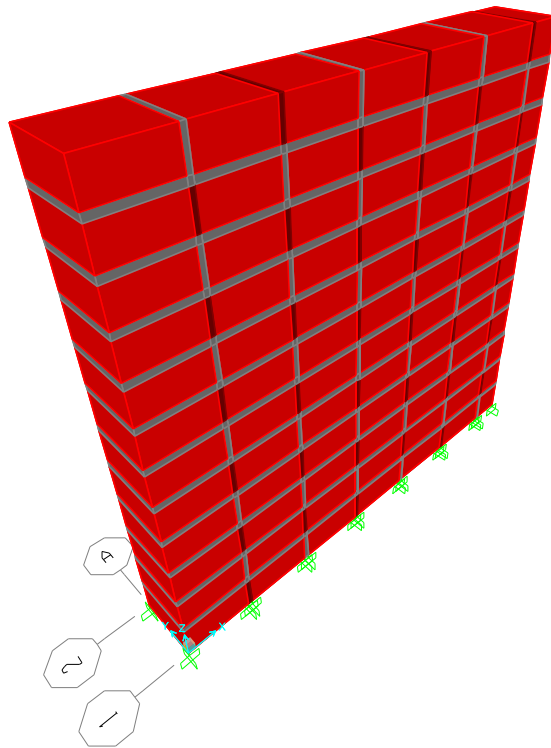


Figura 4.13: Materiales y sus propiedades asignadas en SAP 2000.

4.2.2.1. Resumen de resultados

Tabla 5: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.97%	0.00%	0.20%	0.02%	1.12%	0.00%	0.00%	0.03%	0.21%
SAP 2	0.97%	100.00%	0.00%	0.06%	0.10%	1.93%	0.00%	0.00%	0.02%	0.01%
SAP 3	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.13%	0.25%	0.00%	0.00%
SAP 4	0.02%	0.10%	0.00%	99.84%	1.04%	0.58%	0.00%	0.00%	0.77%	1.37%
SAP 5	0.02%	0.04%	0.00%	1.07%	99.94%	0.01%	0.00%	0.00%	1.75%	0.07%
SAP 6	1.14%	2.05%	0.00%	0.01%	0.00%	99.81%	0.00%	0.00%	0.14%	0.07%
SAP 7	0.00%	0.00%	0.13%	0.00%	0.00%	0.00%	100.00%	0.54%	0.00%	0.00%
SAP 8	0.00%	0.00%	0.15%	0.00%	0.00%	0.00%	0.46%	99.99%	0.00%	0.00%
SAP 9	0.04%	0.01%	0.00%	1.03%	1.97%	0.07%	0.00%	0.00%	99.72%	0.29%
SAP 10	0.10%	0.01%	0.00%	0.05%	0.01%	0.01%	0.00%	0.00%	0.30%	98.67%

Tabla 6: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SIMILITUD		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	234.56	1	236.67	1	1	99.11%
2	496.79	2	517.48	2	2	96.00%
3	998.55	3	1001.05	3	3	99.75%
4	1288.37	4	1357.65	4	4	94.90%
5	1683.59	5	1815.84	5	5	92.72%
6	1806.15	6	2044.48	6	6	88.34%
7	2470.13	7	2471.48	7	7	99.95%
8	2641.31	8	2675.41	8	8	98.73%
9	2766.32	9	3191.56	9	9	86.68%
10	3165.04	10	3459.00	10	10	91.50%

(A)

(B)

(C)

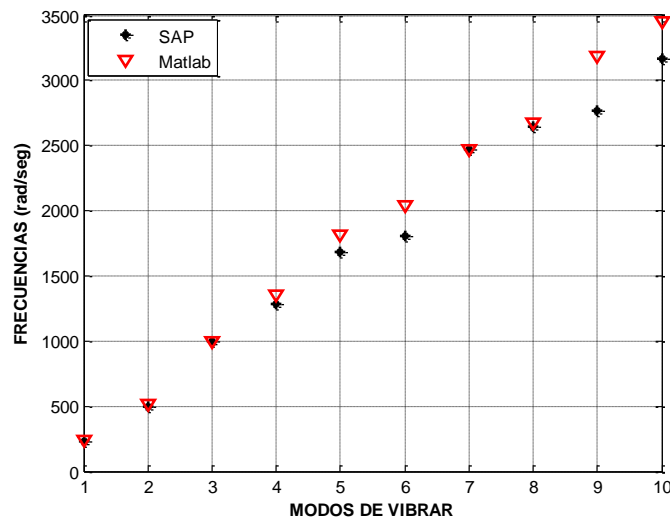


Figura 4.14: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.2.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

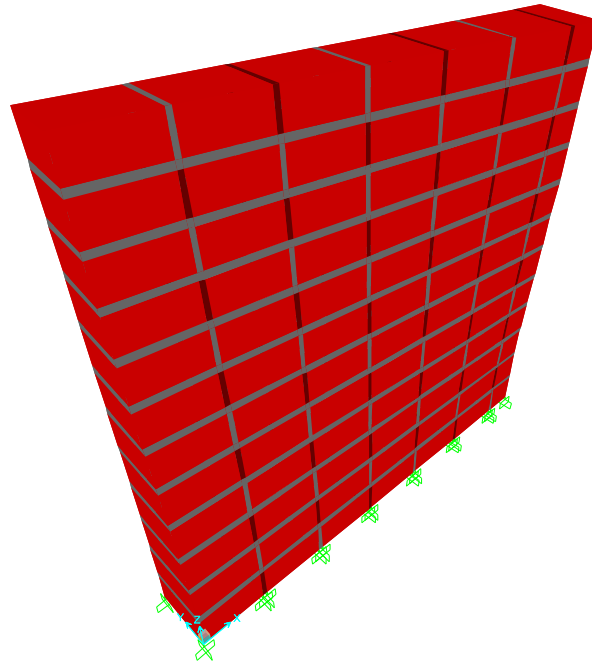


Figura 4.15: Modo 2 (SAP).

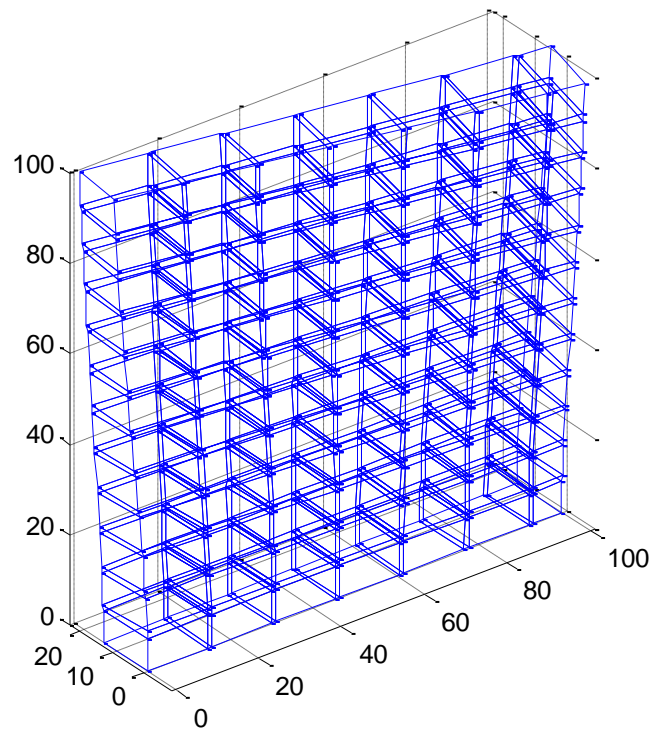


Figura 4.16: Modo 2 (Matlab).

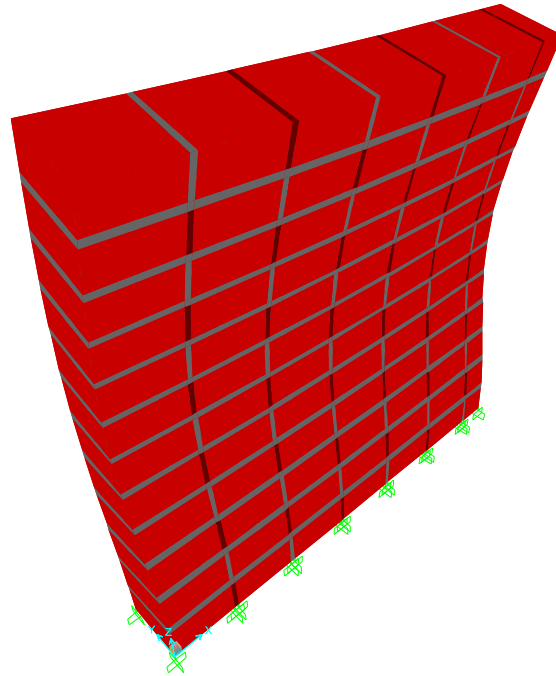


Figura 4.17: Modo 4 (SAP).

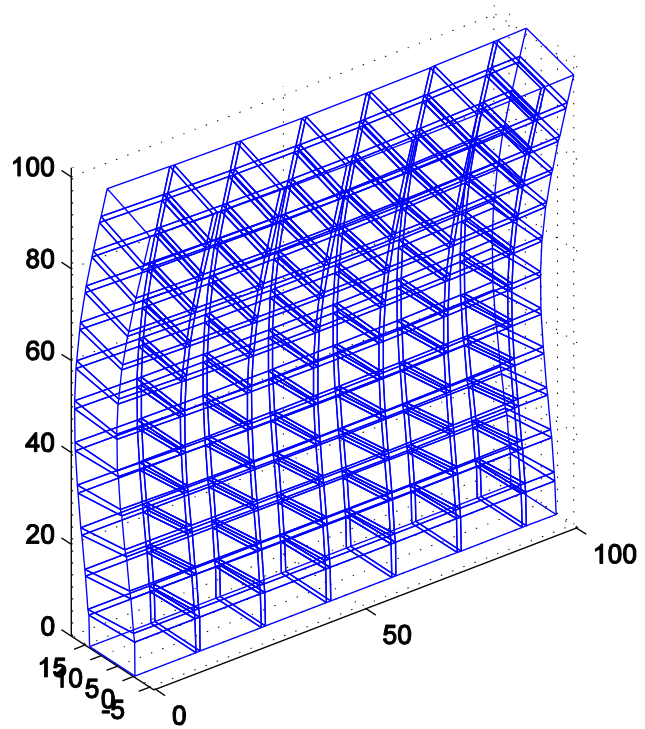


Figura 4.18: Modo 4 (Matlab).

En la Tabla 5 se muestra la matriz de correlación de datos según *MAC*, en la diagonal principal nuevamente se logran muy buenas correlaciones, alcanzando el 100% para varios modos. Por otro lado, la Tabla 6 muestra el porcentaje de similitud de las frecuencias entre los modos con mayor correlación, este porcentaje de similitud varía entre 86.68% y 99.95 %, lo que son muy buenos resultados. En la Figura 4.14 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

A modo comparativo, las frecuencias obtenidas en este modelo son mucho menores que las del modelo empotrado en todos sus lados, esto se debe a que el modelo empotrado solamente en la base es una estructura mucho más flexible que el modelo anterior, lo que genera frecuencias más pequeñas y periodos de vibración más grandes, esto implica desplazamientos más grandes.

Las Figuras 4.15-4-16 y 4.17-4.18, muestran los modos 6 y 4 respectivamente, de ambos softwares.

4.2.3. Modelo 3: Muro de 2m x 1m, empotrado en todos sus extremos

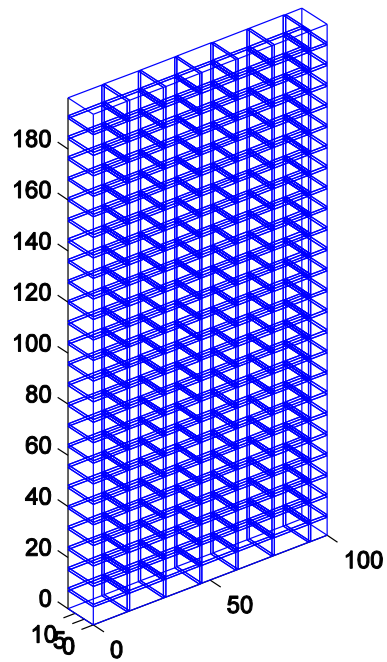


Figura 4.19: Mallado del muro en Matlab.

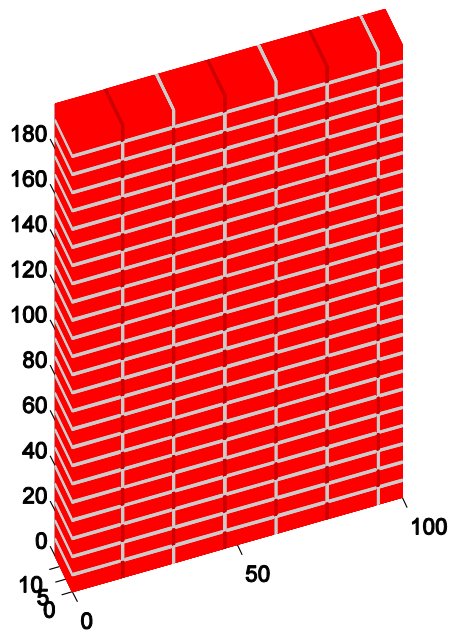


Figura 4.20: Materiales y sus propiedades asignadas en Matlab.

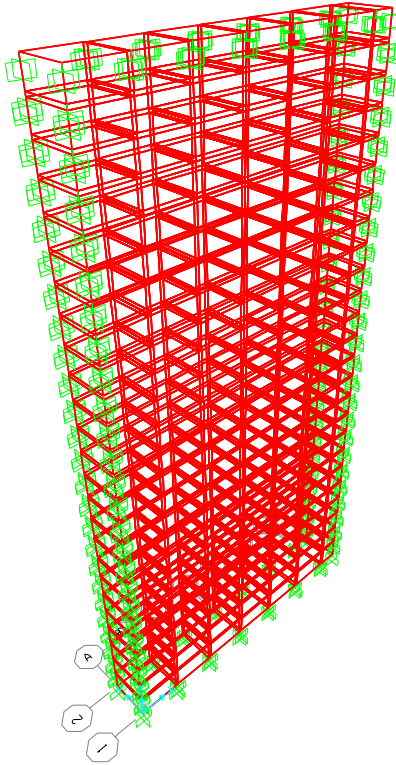


Figura 4.21: Mallado y condiciones de borde en SAP 2000.

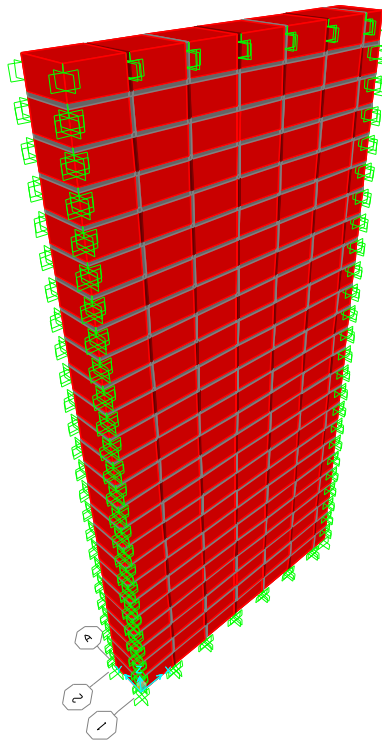


Figura 4.22: Materiales y sus propiedades asignadas en SAP 2000.

4.2.3.1. Resumen de resultados

Tabla 7: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	99.99%	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%
SAP 2	0.00%	99.99%	0.00%	0.04%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 3	0.00%	0.00%	99.96%	0.00%	0.00%	0.00%	0.00%	0.07%	0.00%	0.00%
SAP 4	0.00%	0.01%	0.00%	99.91%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 5	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 6	0.00%	0.00%	0.00%	0.00%	0.00%	99.91%	0.00%	0.00%	0.04%	0.00%
SAP 7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.88%	0.00%	0.00%	0.08%
SAP 8	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	99.82%	0.01%	0.00%
SAP 9	0.00%	0.00%	0.00%	0.00%	0.00%	0.03%	0.00%	0.01%	99.80%	0.00%
SAP 10	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.04%	0.00%	0.00%	99.70%

Tabla 8: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SIMILITUD		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	1871.27	1	1936.31	1	1	96.64%
2	2175.08	2	2273.97	2	2	95.65%
3	2757.47	3	2921.43	3	3	94.39%
4	3581.07	4	3849.87	4	4	93.02%
5	4016.59	5	4070.55	5	5	98.67%
6	4310.07	6	4743.16	6	6	90.87%
7	4491.45	7	4979.31	7	7	90.20%
8	4578.80	8	4993.56	8	8	91.69%
9	4836.35	9	5416.91	9	9	89.28%
10	5362.24	10	6077.62	10	10	88.23%

(A)

(B)

(C)

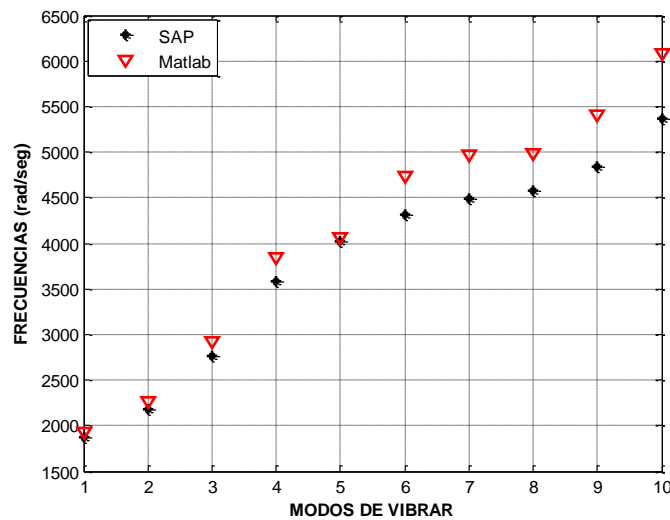


Figura 4.23: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.3.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

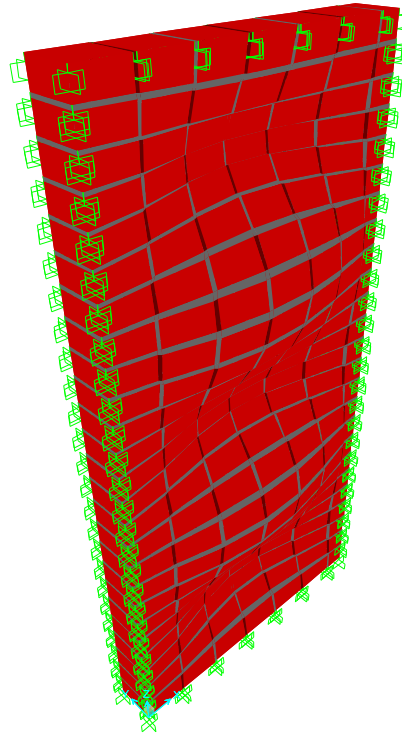


Figura 4.24: Modo 8 (SAP).

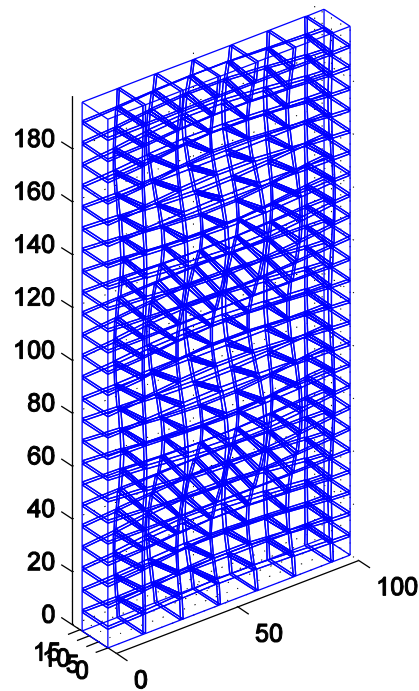


Figura 4.25: Modo 8 (Matlab).

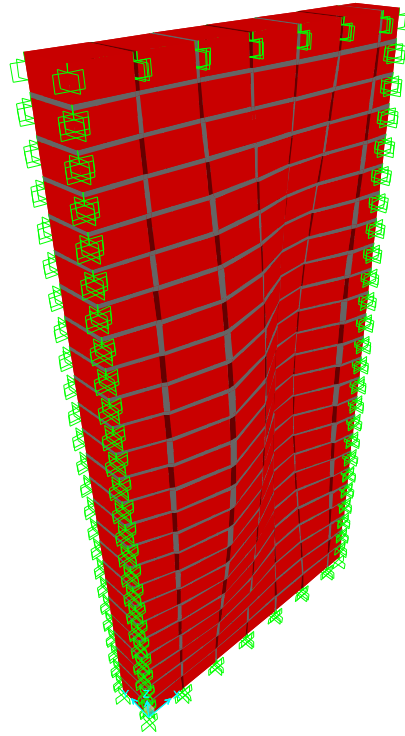


Figura 4.26: Modo 6 (SAP).

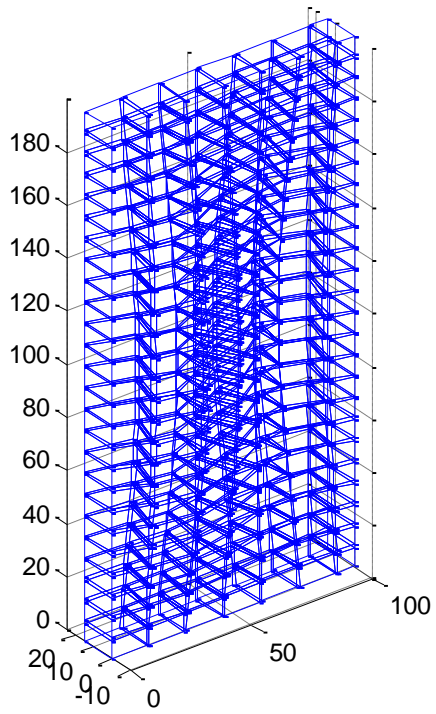


Figura 4.27: Modo 6 (Matlab).

La Tabla 7 muestra en su diagonal principal una correlación de datos muy cercanos al 100% para todos los modos, mientras que el porcentaje de similitud de estos modos varían entre un 88.23% y 98.67%, lo que también son muy buenos valores. En la Figura 4.23 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.24-4.25 y 4.26-4.27, muestran los modos 6 y 4 respectivamente, de ambos softwares.

4.2.4. Modelo 4: Muro de 2m x 1m, empotrado solamente en la base

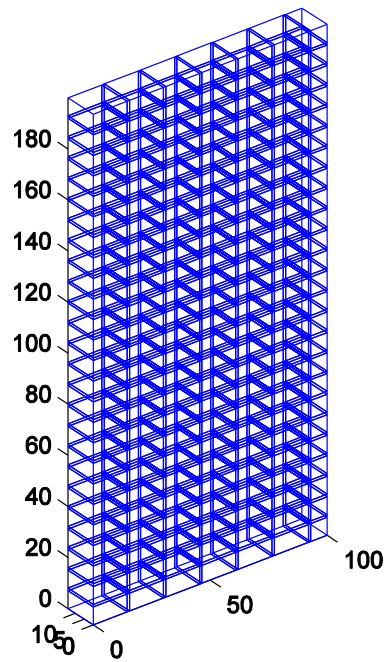


Figura 4.28: Mallado del muro en Matlab.

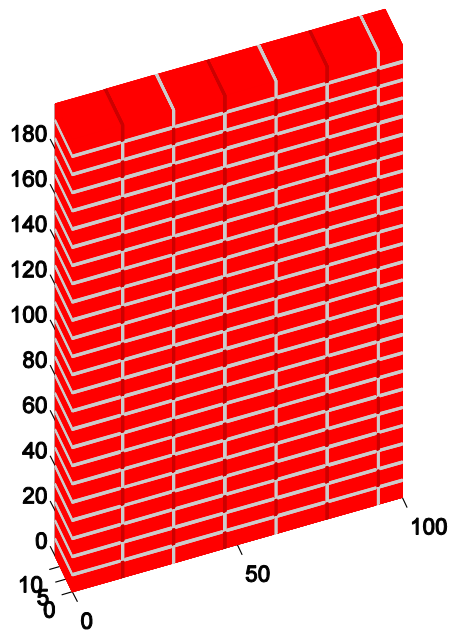


Figura 4.29: Materiales y sus propiedades asignadas en Matlab.

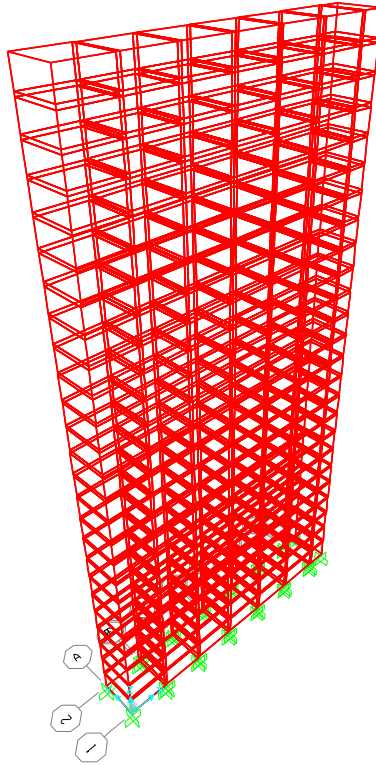


Figura 4.30: Mallado y condiciones de borde en SAP 2000.

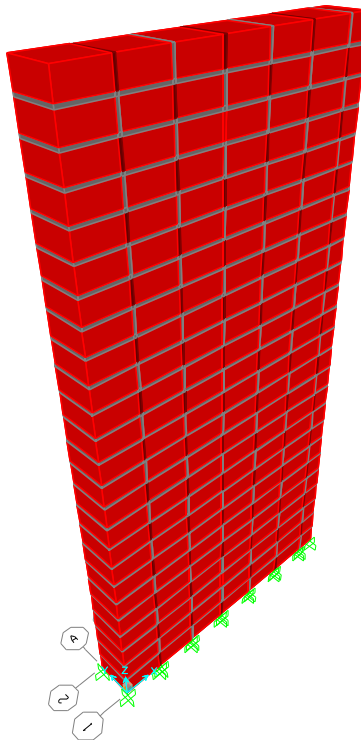


Figura 4.31: Materiales y sus propiedades asignadas en SAP 2000.

4.2.4.1. Resumen de resultados

Tabla 9: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.96%	0.00%	0.03%	0.04%	0.01%	0.00%	0.00%	0.00%	0.53%
SAP 2	0.96%	100.00%	0.00%	0.05%	0.01%	0.00%	0.00%	0.00%	0.00%	1.20%
SAP 3	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.08%	0.06%	0.00%	0.00%
SAP 4	0.01%	0.05%	0.00%	99.99%	0.96%	0.11%	0.00%	0.00%	0.05%	0.09%
SAP 5	0.04%	0.01%	0.00%	0.96%	100.00%	0.04%	0.00%	0.00%	0.03%	0.01%
SAP 6	0.00%	0.01%	0.00%	0.00%	0.06%	99.92%	0.00%	0.00%	1.04%	0.22%
SAP 7	0.00%	0.00%	0.09%	0.00%	0.00%	0.00%	99.86%	0.71%	0.00%	0.00%
SAP 8	0.00%	0.00%	0.03%	0.00%	0.00%	0.00%	0.01%	99.85%	0.00%	0.00%
SAP 9	0.00%	0.01%	0.00%	0.04%	0.00%	1.06%	0.00%	0.00%	99.97%	0.02%
SAP 10	1.01%	1.99%	0.00%	0.06%	0.00%	0.00%	0.00%	0.00%	0.02%	81.93%

Tabla 10: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SIMILITUD		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	62.54	1	62.66	1	1	99.80%
2	228.94	2	237.52	2	2	96.25%
3	355.76	3	356.11	4	3	99.90%
4	377.83	4	383.63	3	4	98.46%
5	737.12	5	770.95	5	5	95.41%
6	1006.65	6	1040.32	6	6	96.65%
7	1304.61	7	1304.58	7	7	99.99%
8	1314.25	8	1320.26	8	8	99.54%
9	1381.38	9	1467.36	9	9	93.77%
10	1718.09	10	1913.89	10	10	88.60%

(A)

(B)

(C)

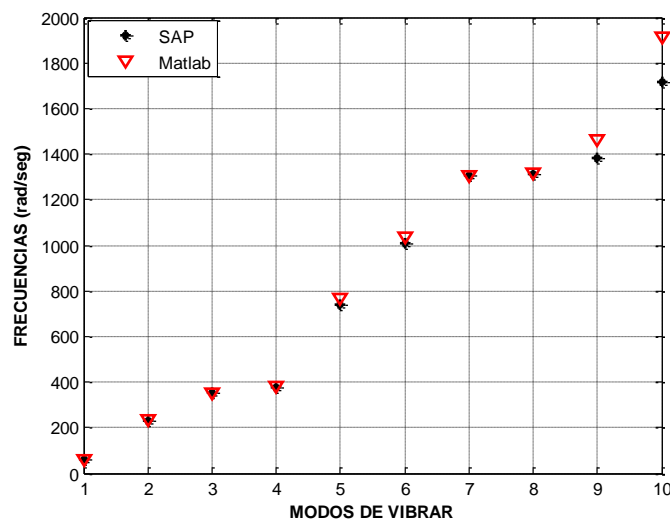


Figura 4.32: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.4.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

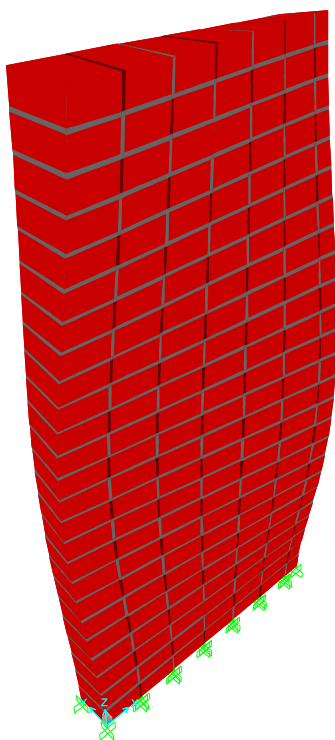


Figura 4.33: Modo 5 (SAP).

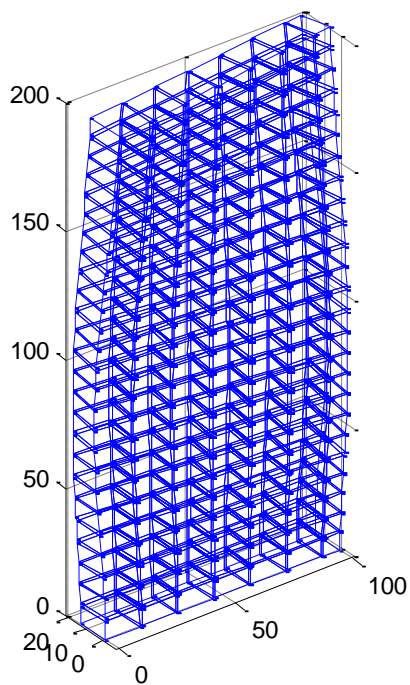


Figura 4.34: Modo 5 (Matlab)

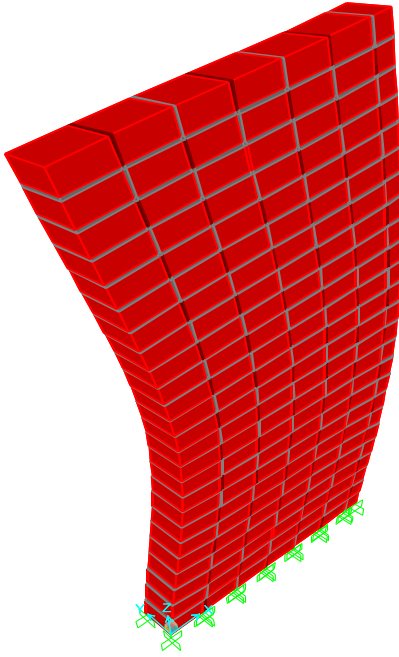


Figura 4.35: Modo 8 (SAP).

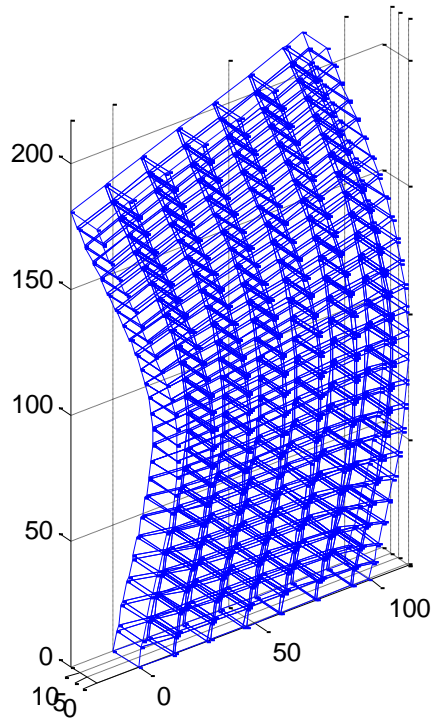


Figura 4.36: Modo 8 (Matlab).

La Tabla 9 muestra en su diagonal principal una correlación de datos muy buena, mientras que el porcentaje de similitud de estos modos varían entre un 88.6% y 99.99%, lo que también son muy buenos valores. En la Figura 4.32 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.33-4-34 y 4.35-4.36, muestran los modos 6 y 4 respectivamente, de ambos softwares.

Comparando los resultados de frecuencias del modelo 2, este modelo tiene frecuencias menores, ya que es menos rígido que el modelo 2, esto debido a su mayor altura, implicando desplazamientos mayores.

4.2.5. Modelo 5: Muro de 2m x 2m, empotrado en todos sus extremos

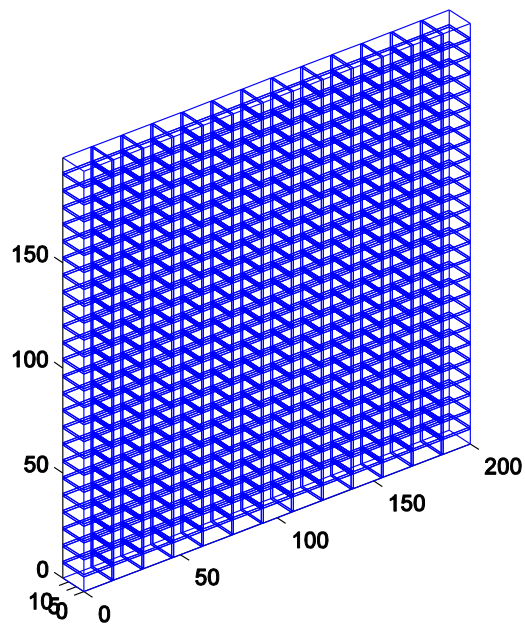


Figura 4.37: Mallado del muro en Matlab.

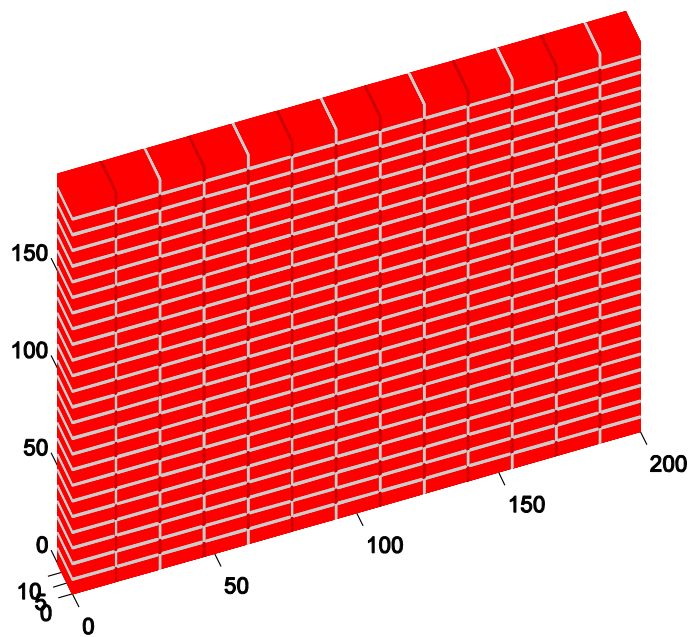


Figura 4.38: Materiales y sus propiedades asignadas en Matlab.

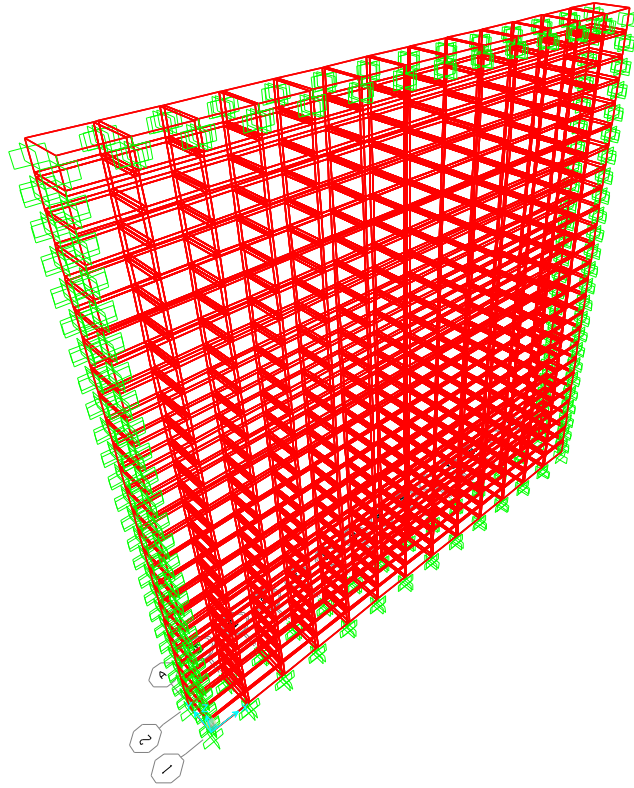


Figura 4.39: Mallado y condiciones de borde en SAP 2000.

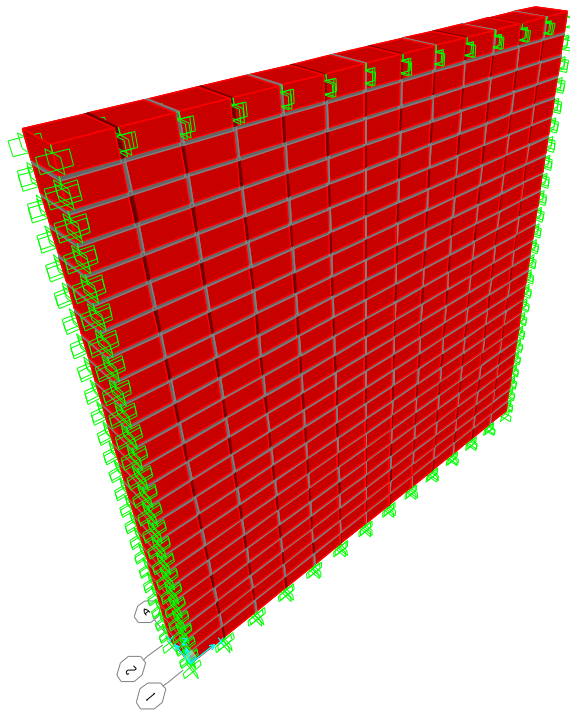


Figura 4.40: Materiales y sus propiedades asignadas en SAP 2000.

4.2.5.1. Resumen de resultados

Tabla 11: Matriz de modos con mayor correlación de datos, según MAC:

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.00%	0.00%	0.00%	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
SAP 2	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.02%	0.00%	0.03%
SAP 3	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%
SAP 4	0.00%	0.00%	0.00%	99.99%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 5	0.00%	0.00%	0.00%	0.00%	99.98%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 6	0.00%	0.00%	0.00%	0.00%	0.00%	99.96%	0.00%	0.00%	0.00%	0.00%
SAP 7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.97%	0.00%	0.00%	0.00%
SAP 8	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.96%	0.00%	0.00%
SAP 9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%
SAP 10	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.94%

Tabla 12: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SAP		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	681.51	1	691.42	1	1	98.54%
2	1226.59	2	1259.36	2	2	97.32%
3	1427.19	3	1480.52	3	3	96.26%
4	1831.90	4	1919.64	4	4	95.21%
5	2045.60	5	2131.62	5	5	95.79%
6	2487.10	6	2658.25	6	6	93.11%
7	2516.48	7	2671.91	7	7	93.82%
8	2787.79	8	3005.39	8	8	92.19%
9	3017.99	9	3030.42	9	9	99.58%
10	3045.26	10	3224.75	10	10	94.10%

(A)

(B)

(C)

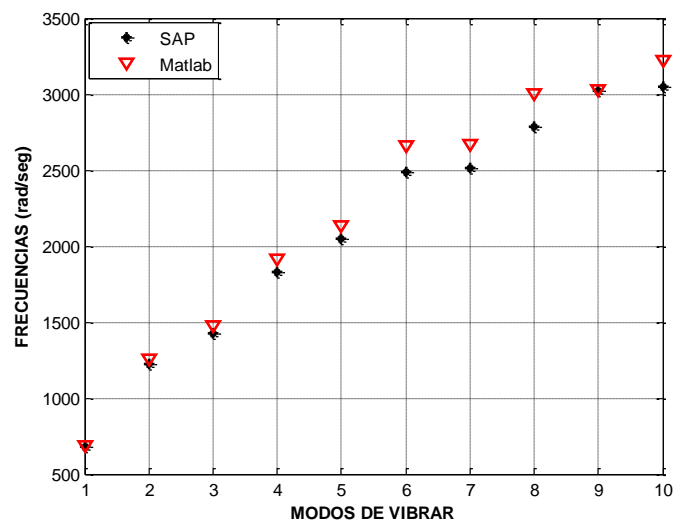


Figura 4.41: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.5.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

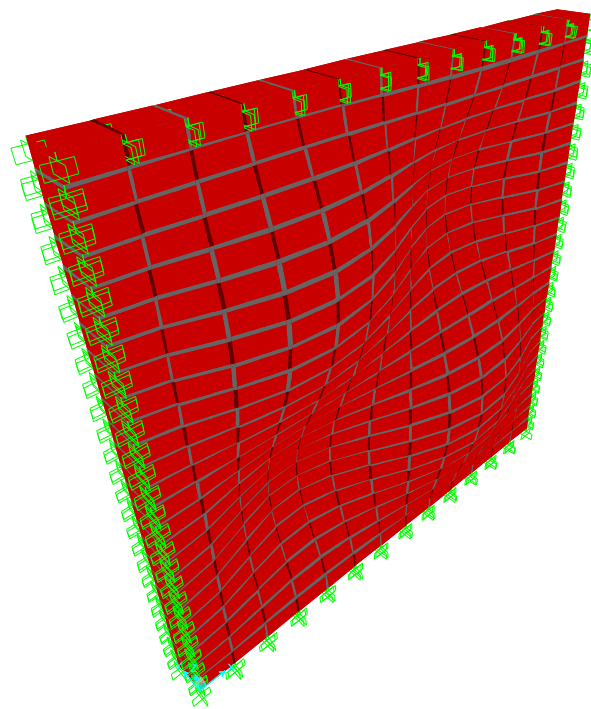


Figura 4.42: Modo 4 (SAP).

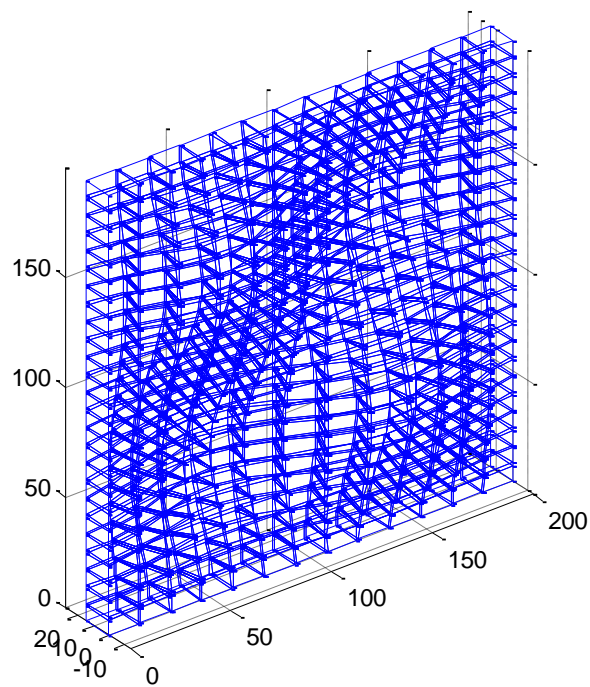


Figura 4.43: Modo 4 (Matlab).

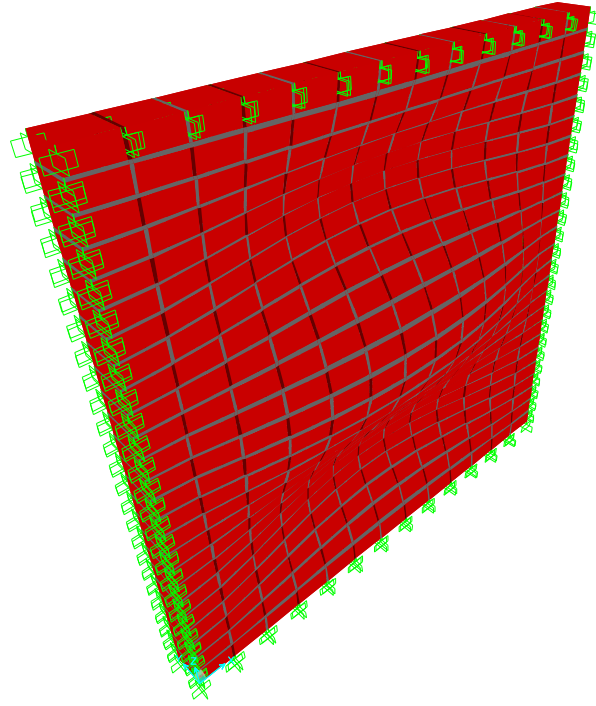


Figura 4.44: Modo 5 (SAP).

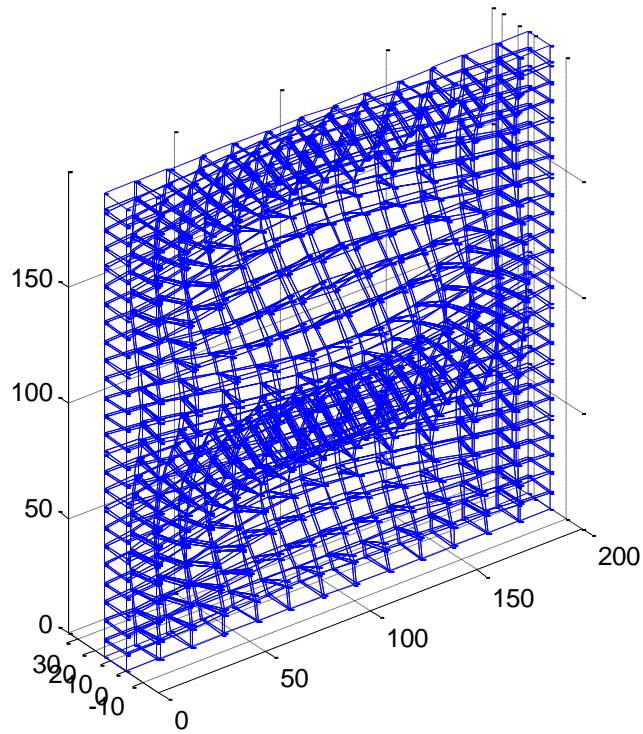


Figura 4.45: Modo 5 (Matlab).

En la diagonal principal de la Tabla 11, se lograron correlaciones sobre el 99.94%. El porcentaje de similitud de los modos con mayor correlación de datos, fue sobre el 92.19%, llegando en el mejor caso a un 99.59%. Los resultados entregados en la Tabla 11 y 12 son muy buenos. En la Figura 4.41 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.42-4-43 y 4.44-4.45, muestran los modos 6 y 4 respectivamente, de ambos softwares.

4.2.6. Modelo 6: Muro 2m x 2m, empotrado solamente en la base

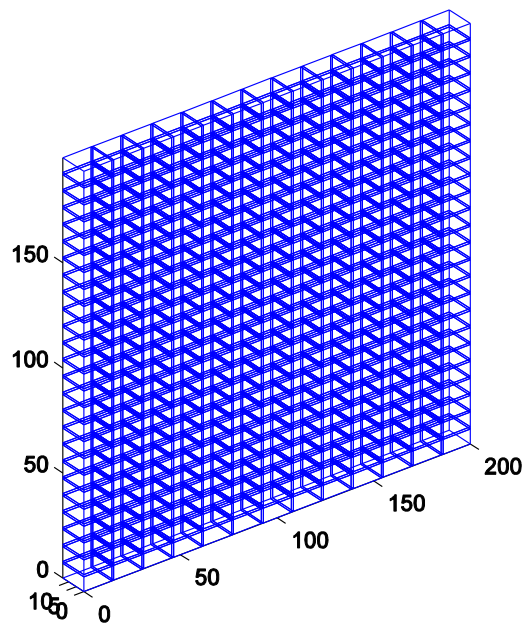


Figura 4.46: Mallado de muro en Matlab.

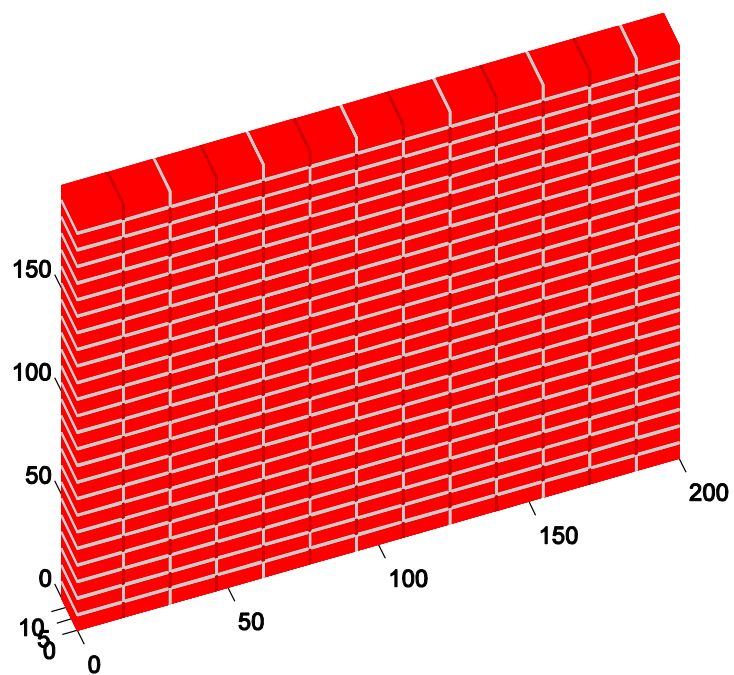


Figura 4.47: Materiales y sus propiedades asignadas en Matlab.

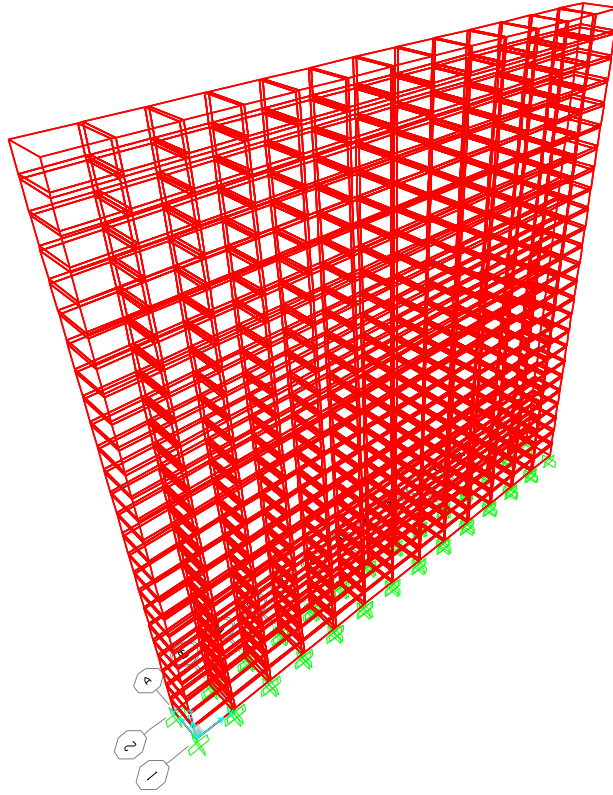


Figura 4.48: Mallado y condiciones de borde en SAP 2000.

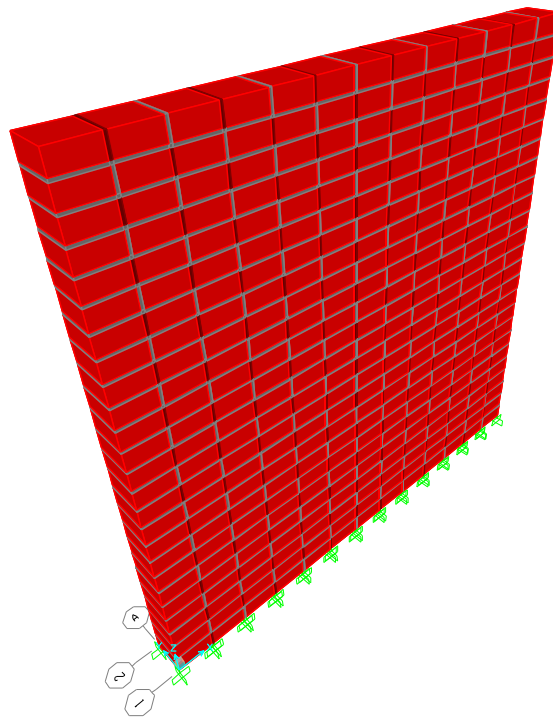


Figura 4.49: Materiales y sus propiedades asignadas en SAP 2000.

4.2.6.1. Resumen de resultados

Tabla 13: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.00%	0.03%	0.00%	0.00%	0.02%	0.00%	0.02%	0.00%	0.00%
SAP 2	0.00%	100.00%	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 3	0.01%	0.00%	99.99%	0.00%	0.00%	0.04%	0.01%	0.12%	0.00%	0.00%
SAP 4	0.00%	0.01%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.09%	0.00%
SAP 5	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 6	0.02%	0.00%	0.00%	0.00%	0.00%	99.98%	0.01%	0.00%	0.00%	0.00%
SAP 7	0.00%	0.00%	0.01%	0.00%	0.00%	0.01%	99.99%	0.00%	0.00%	0.00%
SAP 8	0.01%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	99.92%	0.00%	0.00%
SAP 9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.76%	0.00%
SAP 10	0.00%	0.05%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.15%	100.00%

Tabla 14: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SAP		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	62.83	1	62.97	1	1	99.78%
2	140.93	2	142.50	2	2	98.89%
3	377.61	3	383.62	3	3	98.41%
4	510.78	4	522.56	4	4	97.69%
5	524.82	5	525.15	5	5	99.93%
6	530.20	6	550.91	6	6	96.09%
7	888.25	7	930.67	7	7	95.23%
8	1015.86	8	1049.32	8	8	96.70%
9	1120.42	9	1167.98	9	9	95.75%
10	1305.25	10	1305.40	10	10	99.98%

(A)

(B)

(C)

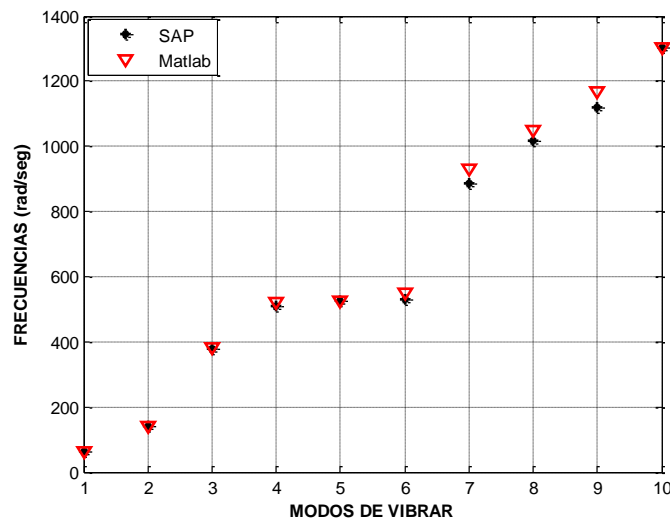


Figura 4.50: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.6.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

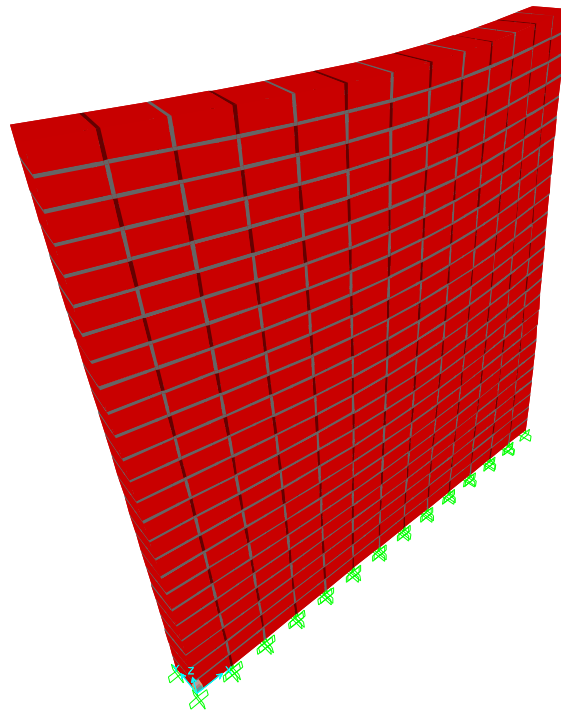


Figura 4.51: Modo 6 (SAP).

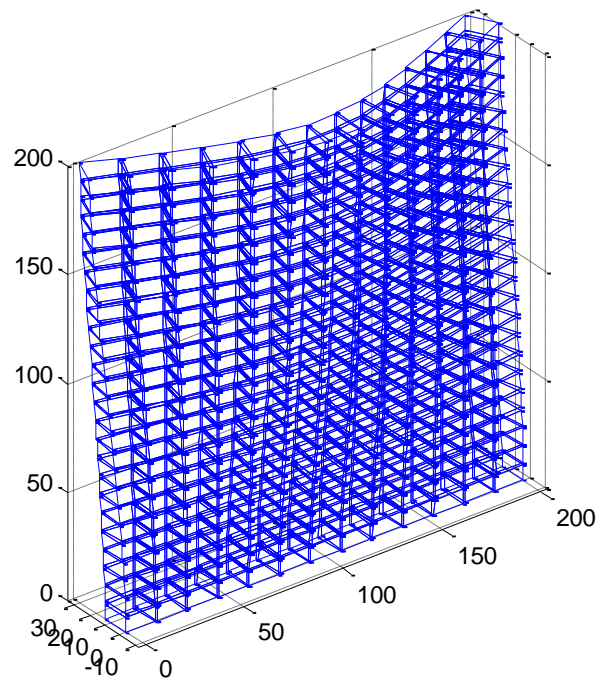


Figura 4.52: Modo 6 (Matlab).

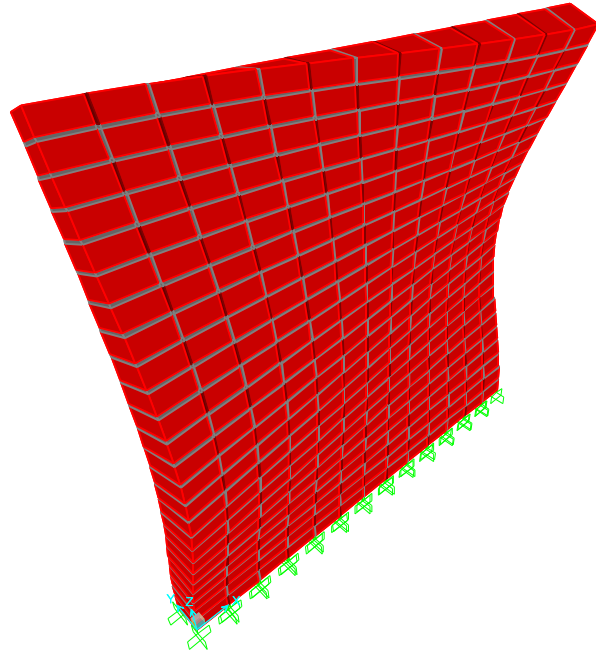


Figura 4.53: Modo 4 (SAP).

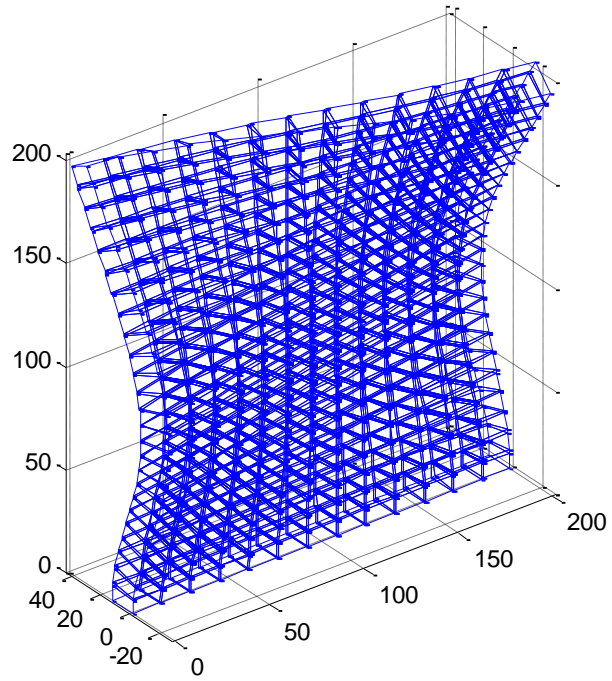


Figura 4.54: Modo 4 (Matlab).

En la diagonal principal de la Tabla 13, se lograron correlaciones sobre 99.76%, llegando al 100% en varios modos. El porcentaje de similitud de los modos con mayor correlación de datos, fue sobre el 95.23%, llegando en el mejor caso a un 99.98%. Los resultados entregados en la Tabla 13 y 14 son muy buenos. . En la Figura 4.50 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.51-4.52 y 4.53-4.54, muestran los modos 6 y 4 respectivamente, de ambos softwares.

4.2.7. Modelo 7: Muro 2m x 3m, empotrado en todos sus extremos

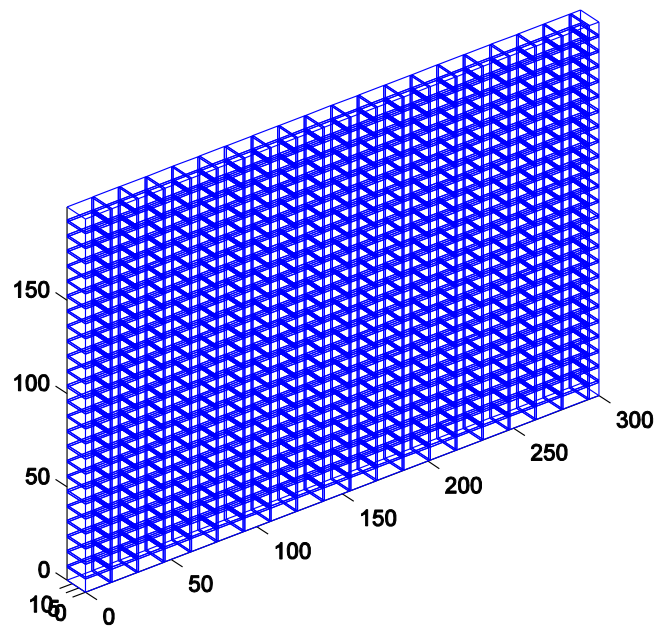


Figura 4.55: Mallado del muro en Matlab.

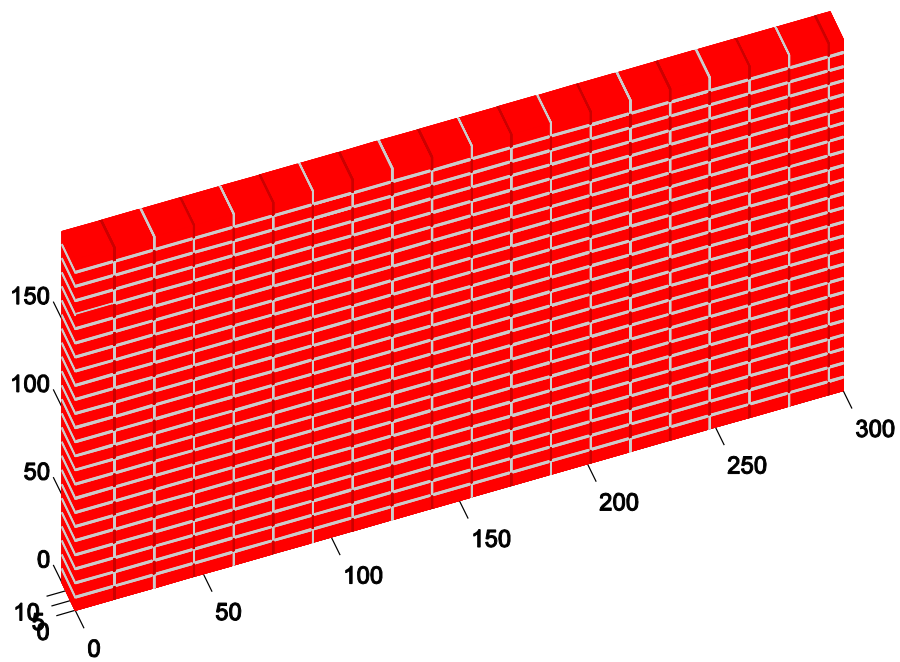


Figura 4.56: Materiales y sus propiedades en Matlab.

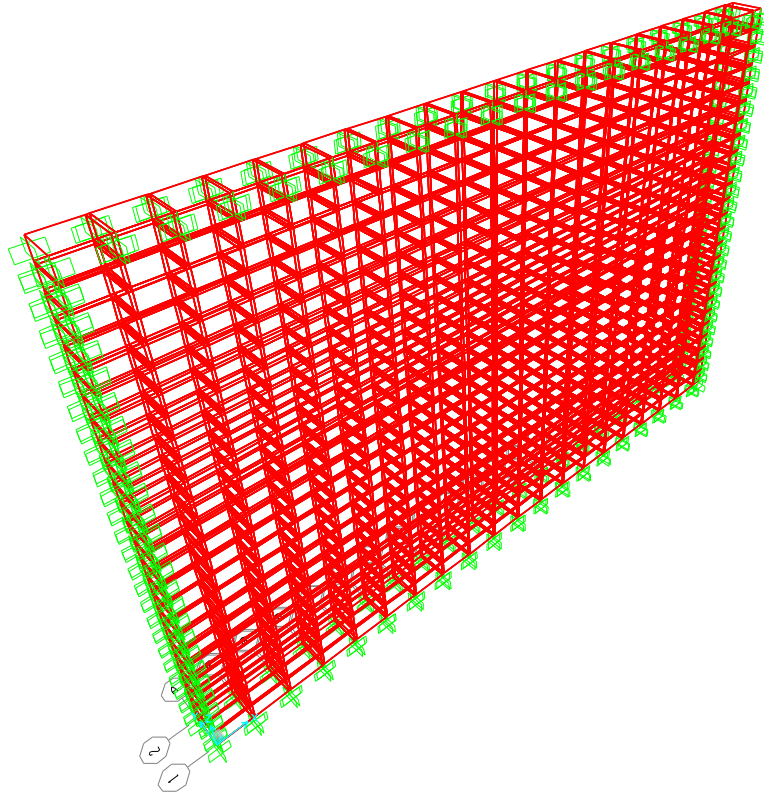


Figura 4.57: Mallado y condiciones de borde en SAP 2000.

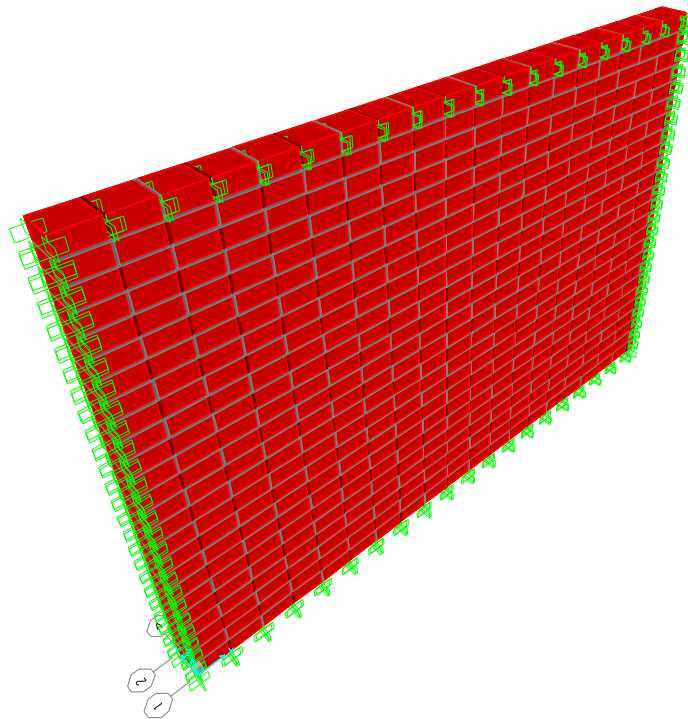


Figura 4.58: Materiales y sus propiedades asignadas en SAP 2000.

4.2.7.1. Resumen de resultados

Tabla 15: Matriz de modos con mayor correlación de datos, según MAC.

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%
SAP 2	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.02%	0.00%	0.01%	0.00%
SAP 3	0.00%	0.00%	100.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
SAP 4	0.00%	0.00%	0.00%	99.99%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
SAP 5	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.02%
SAP 6	0.00%	0.00%	0.00%	0.00%	0.00%	99.99%	0.00%	0.00%	0.00%	0.00%
SAP 7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.97%	0.00%	0.00%	0.00%
SAP 8	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.98%	0.00%	0.00%
SAP 9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.98%	0.00%
SAP 10	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	99.97%

Tabla 16: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SAP		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	482.43	1	487.05	1	1	99.04%
2	776.00	2	791.79	2	2	97.96%
3	1097.56	3	1121.94	3	3	97.77%
4	1263.01	4	1308.83	4	4	96.37%
5	1314.67	5	1356.78	5	5	96.79%
6	1708.86	6	1789.14	6	6	95.30%
7	1898.38	7	2005.53	7	7	94.35%
8	1953.34	8	2027.39	8	8	96.20%
9	2121.82	9	2221.44	9	9	95.30%
10	2265.60	10	2415.36	10	10	93.39%

(A)

(B)

(C)

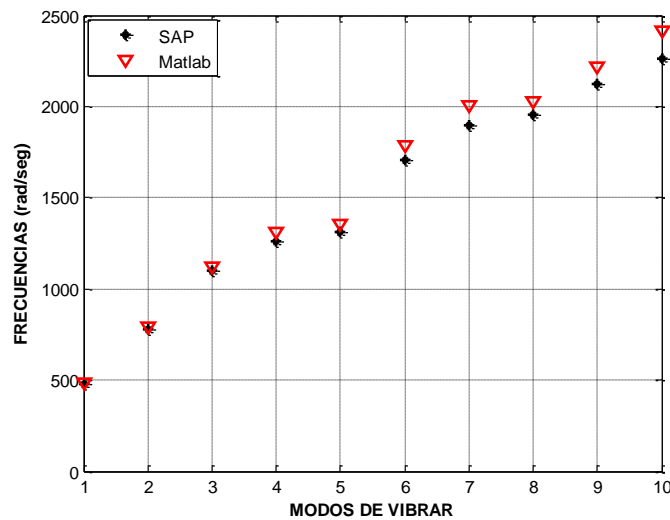


Figura 4.59: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.7.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

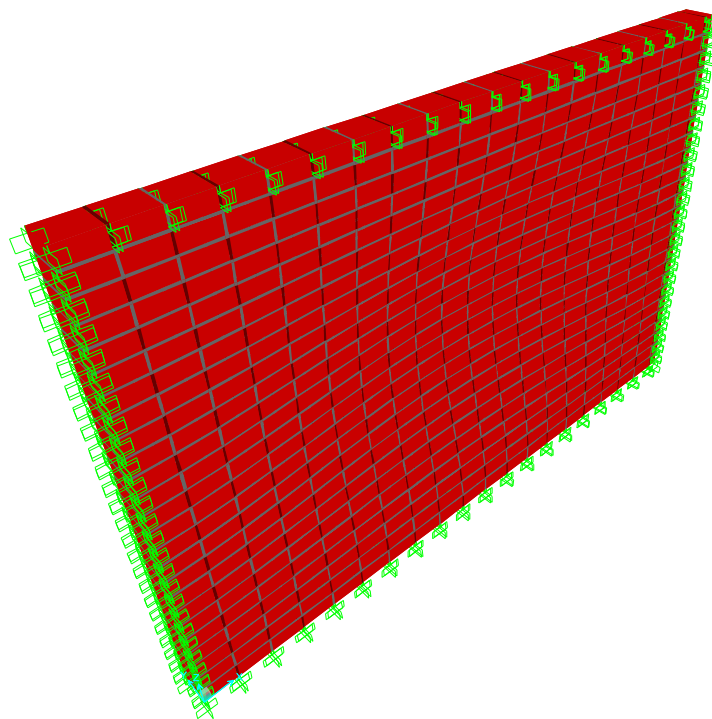


Figura 4.60: Modo 1 (SAP).

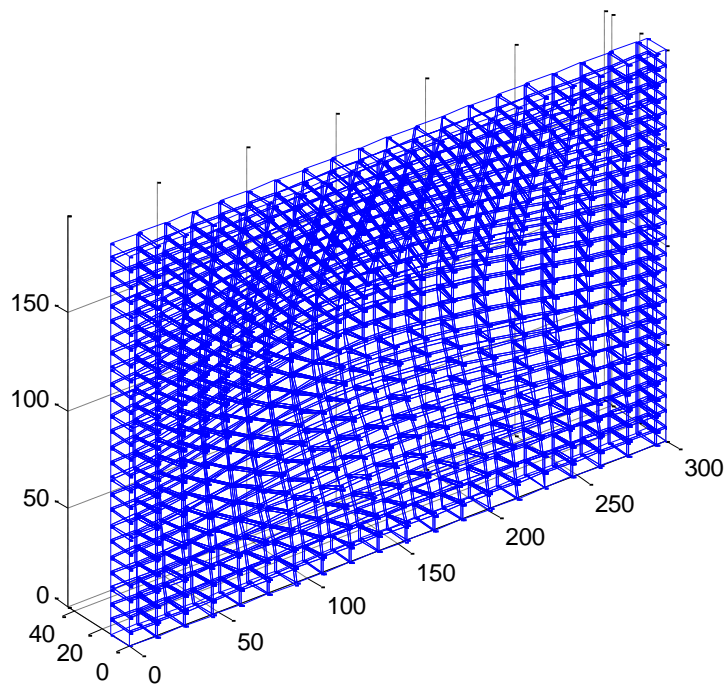


Figura 4.61: Modo 1 (Matlab).

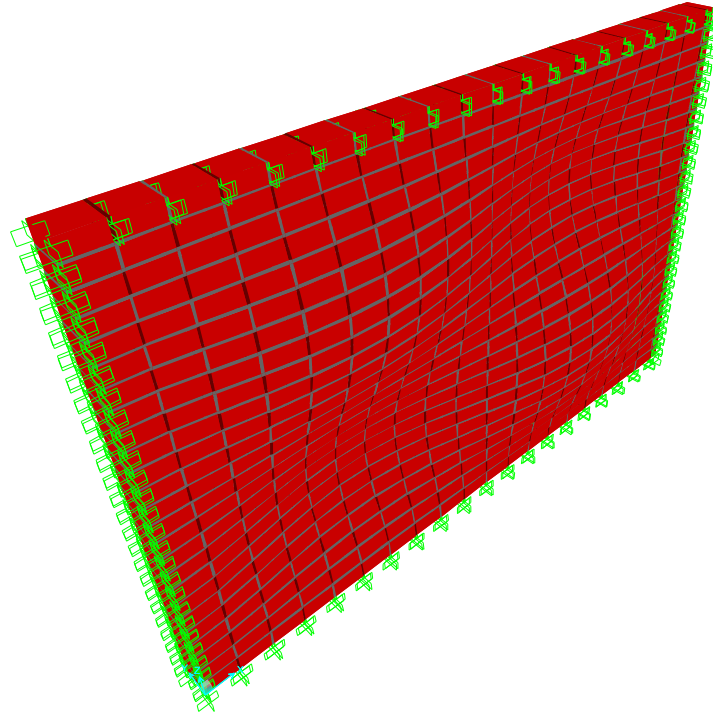


Figura 4.62: Modo 5 (SAP).

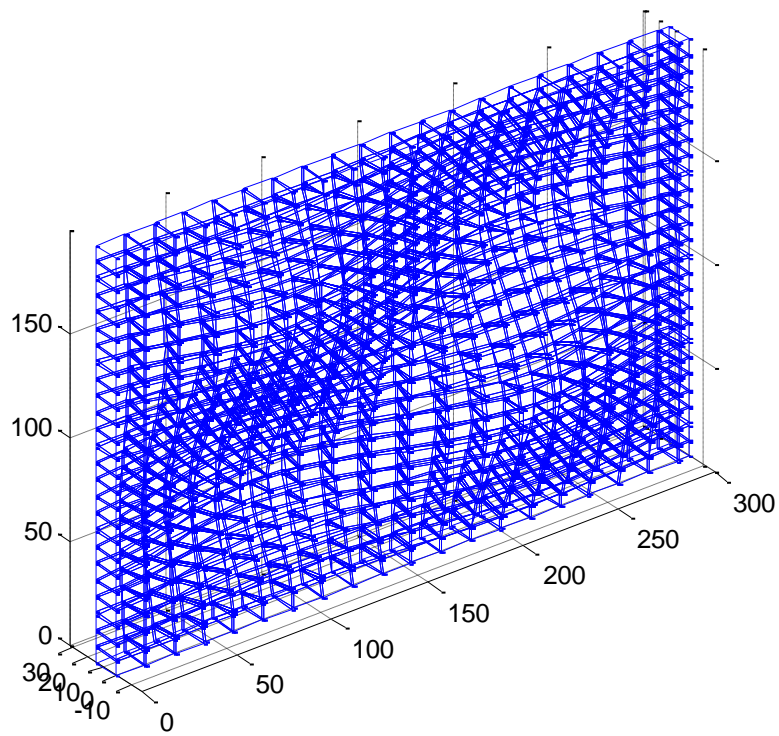


Figura 4.63: Modo 5 (Matlab).

En la diagonal principal de la Tabla 15, se lograron correlaciones sobre 99.97%, llegando al 100% en varios modos. El porcentaje de similitud de los modos con mayor correlación de datos, fue sobre el 93.39%, llegando en el mejor caso a un 99.04%. Los resultados entregados en la Tabla 15 y 16 son muy buenos. . En la Figura 4.59 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.60-4-61 y 4.62-4.69, muestran los modos 6 y 4 respectivamente, de ambos softwares.

4.2.8. Modelo 8: Muro 2m x 3m, empotrado solamente en la base

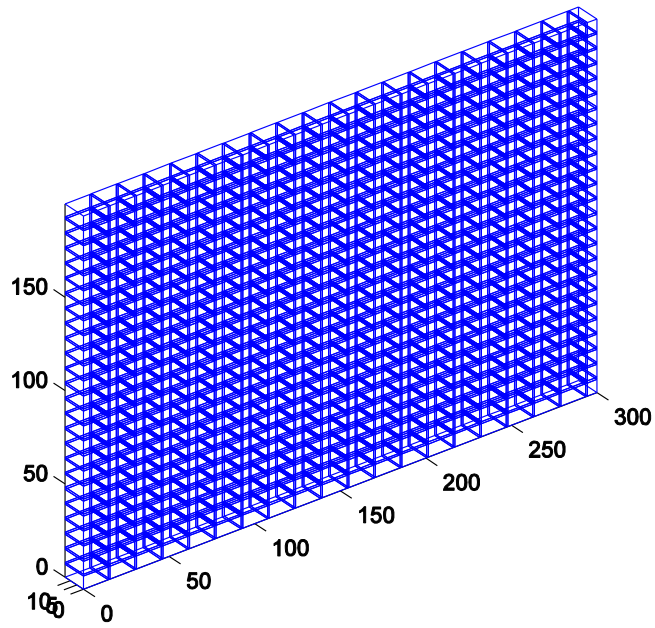


Figura 4.64: Mallado del muro en Matlab.

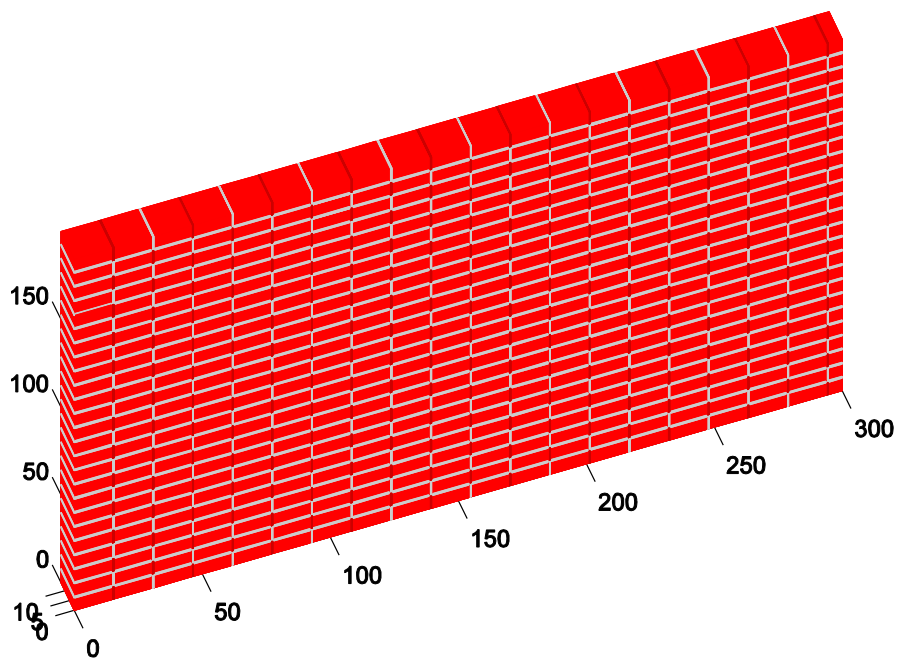


Figura 4.65: Materiales y sus propiedades en Matlab.

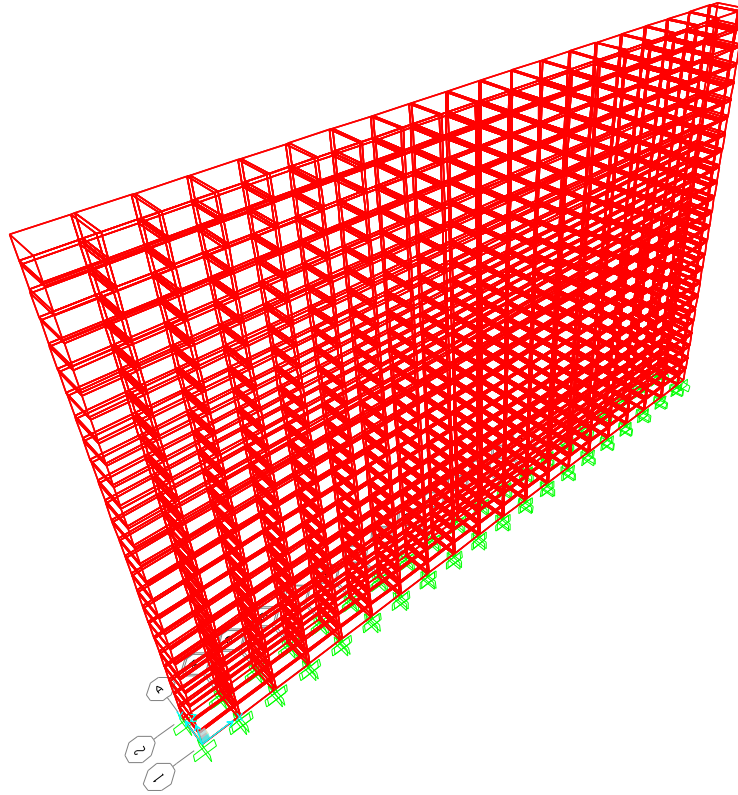


Figura 4.66: Mallado y condiciones de borde en SAP 2000.

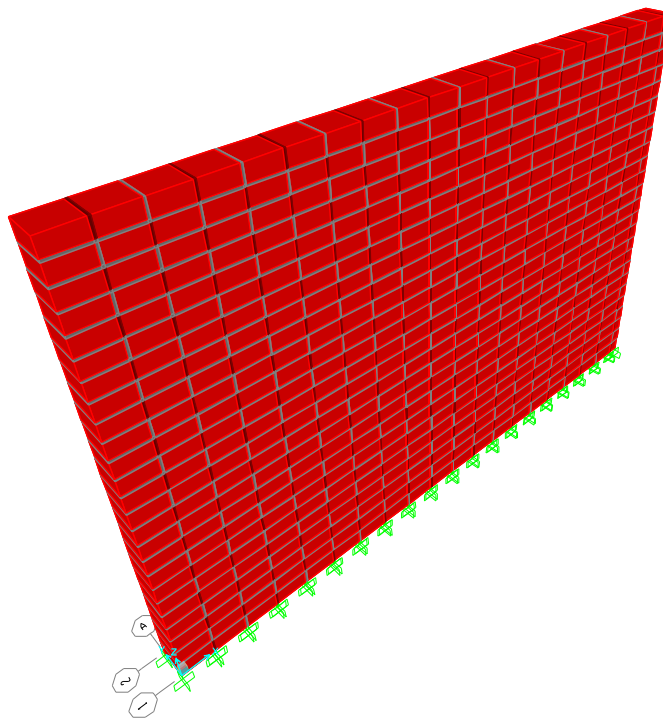


Figura 4.67: Materiales y sus propiedades asignadas en SAP 2000.

4.2.8.1. Resumen de resultados

Tabla 17: Matriz de modos con mayor correlación de datos, según MAC:

Modos	MATLAB 1	MATLAB 2	MATLAB 3	MATLAB 4	MATLAB 5	MATLAB 6	MATLAB 7	MATLAB 8	MATLAB 9	MATLAB 10
SAP 1	100.00%	0.21%	0.28%	0.06%	0.00%	0.00%	0.04%	0.20%	0.01%	0.01%
SAP 2	0.21%	100.00%	0.70%	0.00%	0.00%	0.00%	0.11%	0.86%	0.01%	0.01%
SAP 3	0.30%	0.70%	100.00%	0.00%	0.04%	0.00%	0.08%	0.72%	0.00%	0.03%
SAP 4	0.03%	0.00%	0.00%	99.99%	0.22%	0.00%	0.33%	0.02%	0.25%	0.11%
SAP 5	0.00%	0.00%	0.05%	0.22%	99.99%	0.00%	0.42%	0.23%	0.70%	0.00%
SAP 6	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%
SAP 7	0.23%	0.81%	0.81%	0.01%	0.00%	0.00%	10.61%	89.81%	0.05%	0.01%
SAP 8	0.00%	0.00%	0.00%	0.33%	0.59%	0.00%	89.55%	10.00%	0.70%	0.00%
SAP 9	0.01%	0.01%	0.00%	0.26%	0.67%	0.00%	0.74%	0.01%	99.99%	0.02%
SAP 10	0.00%	0.01%	0.01%	0.00%	0.00%	0.00%	0.01%	0.01%	0.05%	99.85%

Tabla 18: Porcentaje de similitud de frecuencias entre modos con mayor correlación.

Frecuencias (rad/s)		Frecuencias (rad/s)		SIMILITUD		
Modo	SAP	Modo	MATLAB	SAP	MATLAB	SIMILITUD
1	62.90	1	63.04	1	1	99.78%
2	108.13	2	108.77	2	2	99.40%
3	276.32	3	281.46	3	3	98.14%
4	381.73	4	387.57	4	4	98.47%
5	442.01	5	450.46	5	5	98.08%
6	598.99	6	599.21	6	6	99.96%
7	626.83	7	650.19	8	7	97.11%
8	631.93	8	650.48	7	8	96.36%
9	948.48	9	992.74	9	9	95.33%
10	1012.25	10	1046.11	10	10	96.65%

(A)

(B)

(C)

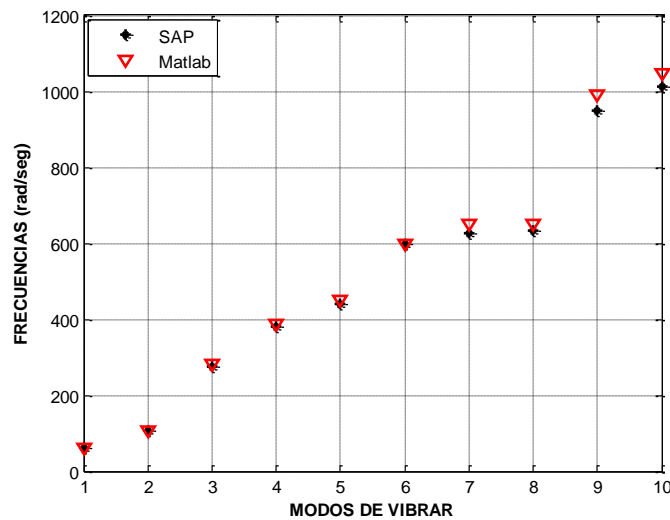


Figura 4.68: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

4.2.8.1.1. Comparación visual entre modos de vibrar entregados por SAP y Matlab

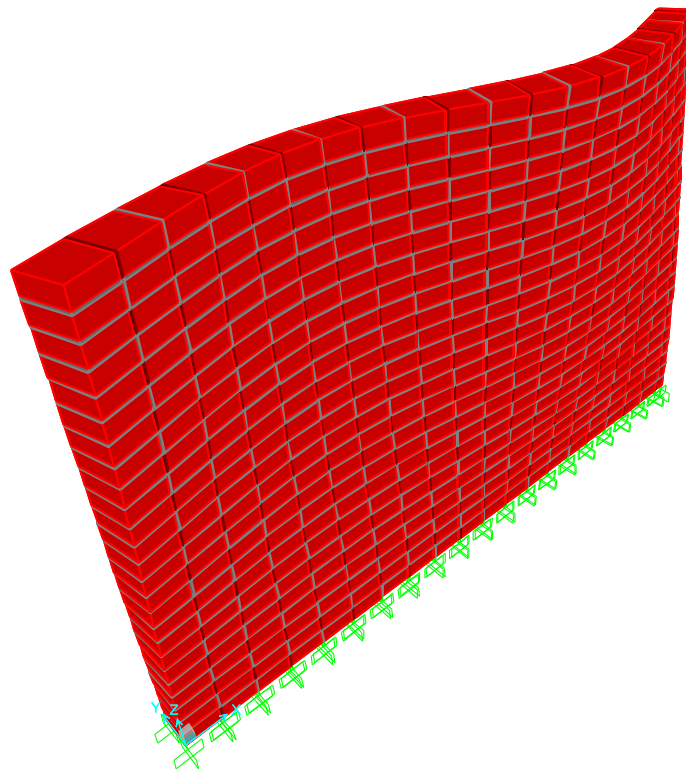


Figura 4.69: Modo 7 (SAP).

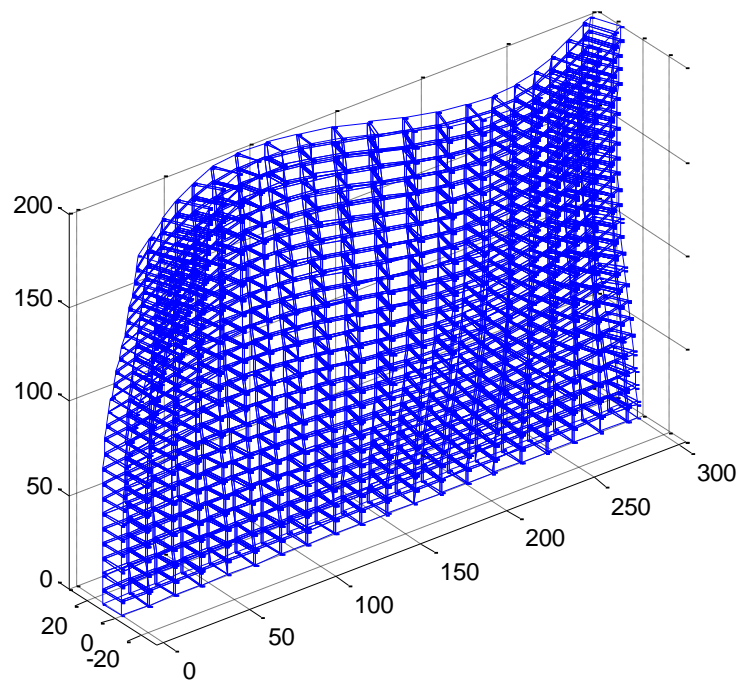


Figura 4.70: Modo 8 (Matlab).

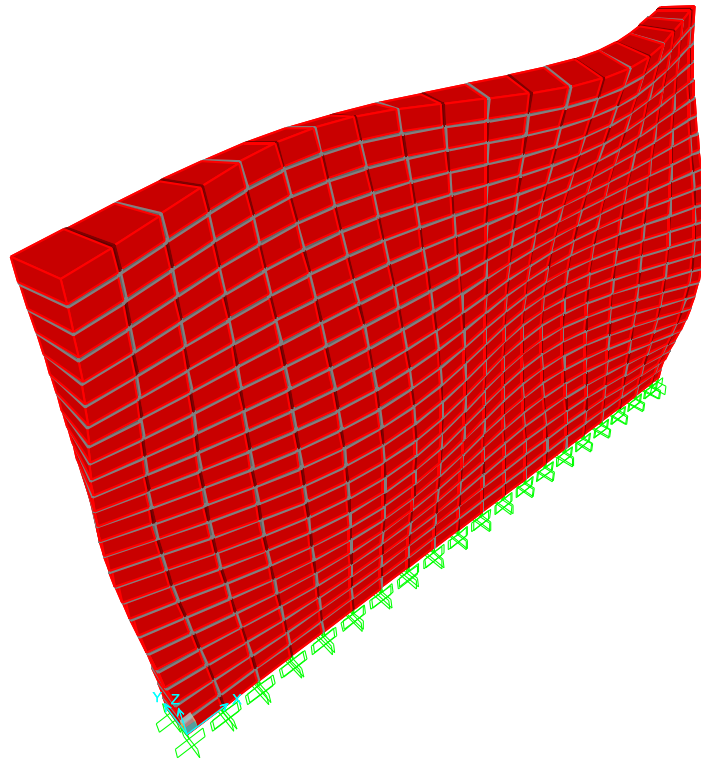


Figura 4.71: Modo 9 (SAP).

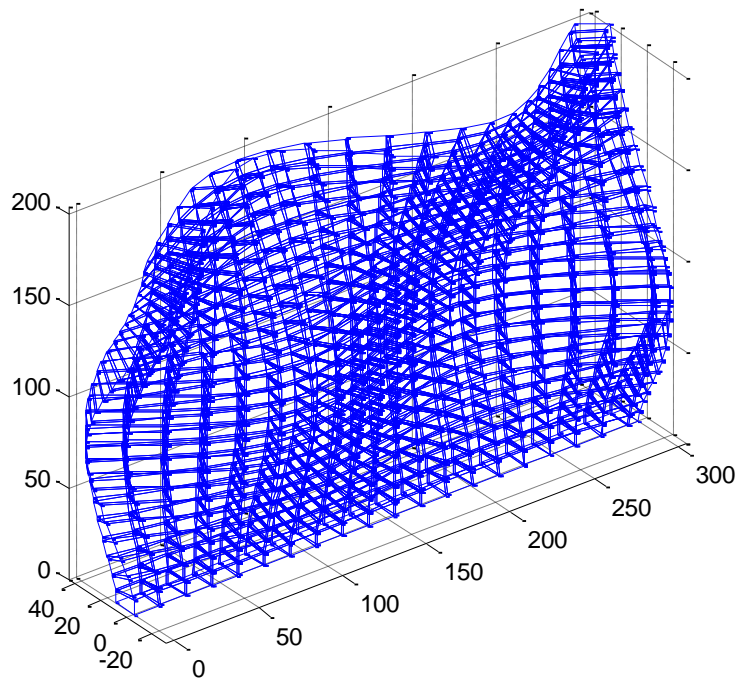


Figura 4.72: Modo 9 (Matlab).

Se puede observar en la Tabla 17 que las correlaciones de los datos de cada modo son muy buenas. El porcentaje de similitud de los modos con mayor correlación de datos, fue sobre el 91.4%, llegando en el mejor caso a un 99.96%. Los resultados entregados en la Tabla 15 y 16 son muy buenos. En la Figura 4.68 se puede observar la diferencia entre las frecuencias asociadas a los modos con mayor correlación.

Las Figuras 4.69 y 4.70, muestran una comparación visual del modo 7 y 8 en Matlab y SAP respectivamente, que son los modos relacionados entre ambos software. También, se puede ver en las Figuras 4.71 y 4.72, el modo 9 en Matlab y SAP respectivamente.

Debido a que en algunos casos hay correlaciones entre las frecuencias obtenidas por SAP y Matlab en torno al 80%, se repite el análisis anterior considerando mallados más finos. Particularmente, los resultados de un muro de 1m x 1m se muestra a continuación.

Modelo 2: Muro 1m x 1m, empotrado solamente en la base, discretizado en 299, 2392 y 19136 elementos

4.2.8.2. Resumen de resultados

Tabla 19: Porcentaje de similitud de frecuencias entre modos con mayor correlación. n=0 →299 elementos, n=1 →2392 elementos y n=2 → 19136 elementos.

Modo	SAP			MATLAB			SIMILITUD		
	Frecuencias (rad/seg)						n=0	n=1	n=2
	n=0	n=1	n=2	n=0	n=1	n=2			
1	234.56	219.89	223.83	236.67	227.52	223.94	99.11%	96.65%	99.95%
2	496.79	495.44	501.65	517.48	509.56	502.96	96.00%	97.23%	99.74%
3	998.55	984.09	989.89	1001.05	993.96	990.02	99.75%	99.01%	99.99%
4	1288.37	1253.34	1277.31	1357.65	1306.31	1281.68	94.90%	95.95%	99.66%
5	1683.59	1707.35	1733.71	1815.84	1772.64	1742.24	92.72%	96.32%	99.51%
6	1806.15	1782.34	1766.75	2044.48	1848.12	1781.80	88.34%	96.44%	99.16%
7	2470.13	2431.49	2452.04	2471.48	2460.67	2452.13	99.95%	98.81%	99.99%
8	2641.31	2638.52	2643.46	2675.41	2656.46	2645.66	98.73%	99.32%	99.92%
9	2766.32	2896.50	2929.94	3191.56	3035.23	2958.56	86.68%	95.43%	99.03%
10	3165.04	3178.25	3232.00	3459.00	3325.75	3251.53	91.50%	95.57%	99.40%

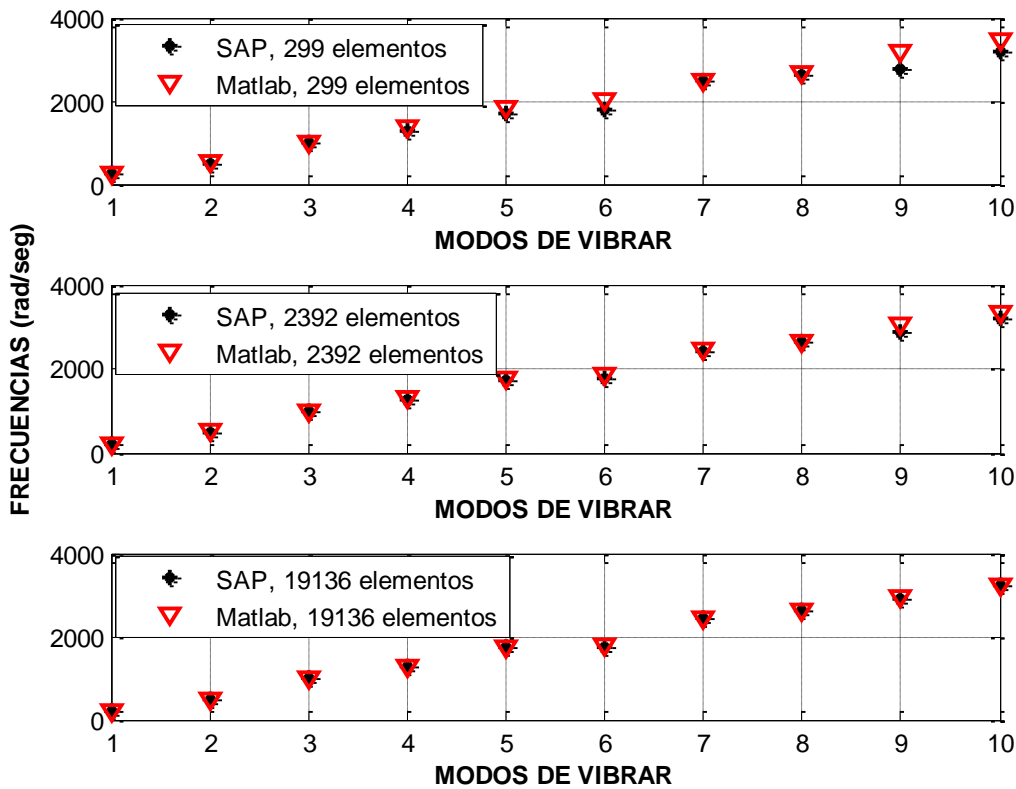


Figura 4.73: Frecuencias de los modos con mayor correlación entre SAP y Matlab.

En la Tabla 19 y Figura 4.73 se puede ver que a medida que se refinó la malla en ambos software, los resultados, en este caso las frecuencias, tendieron a un similitud del 100%, lo que es muy bueno. Para el caso en que el muro se discretizó en 299 elementos, el porcentaje de similitud varió entre 86.68% y 99.95%. Al ser discretizado el muro en 2392 elementos, el porcentaje de similitud mejoró, variando entre 95.43% y 99.32%. Finalmente, al ser discretizado el muro en 19136 elementos, el porcentaje de similitud mejoró notablemente, variando entre 99.03% y 99.99 %, lo que es un resultado excelente.

Debido a que SAP2000 es una “caja negra” (ya que según los datos de entrada que recibe, posterior a un análisis, el programa entrega resultados, desconociendo su funcionamiento interno), se hace difícil o casi imposible encontrar las causas de las pequeñas diferencias entre los resultados de ambos softwares. Por lo mostrado anteriormente, estas diferencias desaparecen en mallas más finas.

5. Conclusiones

Con los resultados entregados por la rutina en el software Matlab y el software SAP2000, de los ocho modelos analizados, se pueden sacar varias conclusiones y recomendaciones que se nombrarán a continuación.

1. Se programó una rutina en el software Matlab capaz de generar un mallado en 3D para muros de albañilería, considerando todas las posibles restricciones de geometría, disposición de materiales y condiciones de borde, con tan solo ingresar las dimensiones del ladrillo, muro y propiedades mecánicas de los materiales. Este código está basado en el método de elementos finitos y es capaz de hacer un análisis dinámico del muro, entregando como resultados, la matriz de modos de vibración, frecuencias naturales y animaciones de los modos de vibrar.
2. Se analizaron distintos modelos con cuatro geometrías distintas (1m x 1m, 2m x 1m, 2m x 2m y 2m x 3m) y dos tipos de condiciones de borde (empotrado en todos sus extremos y solamente en el extremo inferior). Se comparó la matriz de formas modales obtenidas mediante ambos software considerando deformaciones en los tres ejes del panel. Estas matrices fueron comparadas bajo el criterio de certeza modal (MAC). Los resultados de esta comparación basada en el factor MAC fueron buenos, ya que en la diagonal principal de la matriz MAC se obtuvo correlaciones casi perfectas (100%) en la gran mayoría de los casos.
3. Para el caso de las frecuencias naturales, se compararon los modos con mayor correlación según MAC, obteniendo un porcentaje de similitud de frecuencias entre dichos modos. Para los ocho modelos analizados, los porcentajes de similitud de frecuencias varían entre 84.01% y 99.99%. Por ejemplo para el modelo 1, que es donde se registró la similitud más baja, los porcentajes varían entre 84.01% y 98.35%. Por el contrario, el modelo 6 registró las similitudes más cercanas al 100%, variando entre 95.23% y 99.98%. El modelo 2 (1m x 1m y empotrado solamente en la base) fue discretizado en 299, 2392 y 19136 elementos en ambos software, mejorando notablemente el porcentaje de similitud entre las frecuencias, variando entre 86.68% - 99.95%, 95.43% - 99.32% y 99.03% - 99.99 % para 299, 2392 y 19136 elementos respectivamente. En general los resultados fueron buenos para todos los modelos analizados.
4. Por lo tanto, se recomienda usar un $n \geq 2$ en la generación del mallado, la cual subdivide cada elemento del muro en 8^n elementos.
5. Por último, se hace interesante realizar un ensayo experimental de un muro de albañilería y conocer las propiedades mecánicas de cada material a través del modelo. Esto se lograría corriendo el modelo para un set de datos de entrada, y ver para qué combinación de datos los

resultados experimentales y del modelo convergen al mismo valor. Luego de encontrar bajo que parámetros los resultados son más sensibles y en el rango que se encuentran, se espera modificar este o estos parámetros hasta llegar al mismo resultado entre el modelo numérico y el modelo experimental.

6. Referencias

Chávez, M. (2013). “Caracterización de las propiedades mecánicas de la albañilería basado en ensayos vibratorios no destructivos.” Informe de Proyecto de Título para al Título de Ingeniero Civil. Departamento de Ingeniería Civil, Facultad de Ingeniería, Universidad católica de la Santísima Concepción.

Galdames, A. (2014). “Desarrollo de una herramienta numérica para el modelamiento de paneles de albañilería.” Informe de Proyecto de Título para al Título de Ingeniero Civil. Departamento de Ingeniería Civil, Facultad de Ingeniería, Universidad católica de la Santísima Concepción.

Norma Chilena Oficial NCh 2123. Albañilería confinada – Requisitos de diseño y cálculo. Instituto Nacional de Normalización. INN Chile.

Norma Chilena Oficial NCh 1928. Of 1993, modificada en 2009. Albañilería armada – Requisitos de diseño y cálculo. Instituto Nacional de Normalización. INN Chile.

Manual del albañil, Instituto del cemento y del hormigón de Chile, primera edición 2010.

Astroza, M, Schmidt, A. (2004). “Capacidad de deformación de muros de albañilería confinada para distintos niveles de desempeño.” Revista de Ingeniería sísmica N° 70, 59-75

Becerra, M. (2016). “Un método de elementos finitos libre de bloque numérico para resolver vigas isotrópicas y funcionalmente graduadas, utilizando la teoría de alto orden de Reddy-Bickford.” Informe de Proyecto de Título para al Título de Ingeniero Civil. Departamento de Ingeniería Civil, Facultad de Ingeniería, Universidad católica de la Santísima Concepción.

Chaves, E, Mínguez, R. (2010). “Mecánica computacional en la ingeniería con aplicaciones en Matlab.” Escuela Técnica superior de Ingenieros de caminos, Canales y Puertos. Universidad de Castilla-La Mancha.

Celigüeta, J. (2011). “Método de los Elementos Finitos para Análisis Estructural.” Pamplona: Universidad de Navarra, Servicio de Publicaciones.

Gambarotta, L; Lagomarsino, S. (1997a). “Damage models for the seismic response of brick masonry shear walls. Part I: The mortar joint model and its applications.” Earthquake Engineering and Structural Dynamics, Vol. 26, 423-439.

Gambarotta, L; Lagomarsino, S. (1997b). "Damage models for the seismic response of brick masonry shear walls. Part II: The continuum model and its applications." *Earthquake Engineering and Structural Dynamics*, Vol. 26, 441-462.

Oñate, E. (1992). "Cálculo de Estructuras por el Método de Elementos Finitos." Centro Internacional de Métodos de Numéricos en Ingeniería, Primera Edición. España.

7. Anexos

7.1. Código programada en Matlab,basado en el método de elementos finitos para analizar el comportamiento dinámico de un muro de albañilería

```
clear all
close all
clc
```

7.1.1. Datos de entrada

```
%%----- INGRESO DE DATOS-----
-
% DIMENSIÓN DEL LADRILLO EN cm-----
largo=30;alto=7;prof=14;
% ESPESOR MORTERO EN cm-----
emortero=1;
% DIMENSIONES DEL MURO EN cm-----
distx=300;distz=200;disty=prof; %(altura)%(largo)%(profundidad)
% n ASOCIADO AL MALLADO-----
n=0;
%% PROPIEDADES
% LADRILLO, MORTERO, INTERFACE
E=9.8066499206*[896942380.935387 107070202.362683 244731891.114703]; N/m2
v=[0.22 0.3 0.22];
Densidad=[1800 2000 1800]; %kg/m3;
```

7.1.2. Mallado en 3D para un muro de albañilería

```
%%-----MALLADO MURO DE ALBAÑILERÍA-----
nladx=floor(distx/(largo+emortero));
difx=distx-(nladx*(largo+emortero));
if difx==0
    emorterox=emortero+(emortero/(nladx-1));
elseif difx<largo*0.2
    emorterox=emortero+((difx+emortero)/(nladx-1));
elseif largo*0.5<difx & difx<largo*0.7
    syms a b
    a=(emortero+(difx-(largo*0.5-b*0.5))/nladx)-b;
    emorterox=solve(a,b);
else
    emorterox=emortero;
end
interface=emorterox;
nladz=floor(distz/(alto+emortero));
difz=distz-alto*nladz;
emorteroz=difz/(nladz-1);
%% MALLADO 3D
coordx=zeros(nladx*4,1);
if difx<largo*0.2
for i=1:nladx*2-1;
    coordx(2*i,1)=coordx((2*i-1),1)+(largo/2)-(interface/2);
    coordx((2*i-1)+2,1)=coordx(2*i,1)+interface;
    coordx(nladx*4,1)=coordx(nladx*4-1,1)+(largo/2)-(interface/2);
end
elseif difx>=largo*0.7
    for j=1:nladx*2;
```

```

    coordx(2*j,1)=coordx((2*j-1),1)+(largo/2)-(interface/2);
    coordx((2*j-1)+2,1)=coordx(2*j,1)+interface;
    coordx(4*nladx+2,1)= coordx(4*nladx,1)+(largo-emortero)/2+emortero;
    coordx(4*nladx+3,1)= coordx(4*nladx+2,1)+emortero;
    coordx(4*nladx+4,1)= coordx(4*nladx+3,1)+difx-emorterox*0.5-
largo*0.5;
    end
elseif largo*0.5<difx & difx<largo*0.7
    for k=1:nladx*2;
        coordx(2*k,1)=coordx((2*k-1),1)+(largo/2)-(interface/2);
        coordx((2*k-1)+2,1)=coordx(2*k,1)+interface;
        coordx(4*nladx+2,1)= coordx(4*nladx,1)+emorterox+largo*0.5-
emorterox*0.5;
    end
else
    for l=1:nladx*2;
        coordx(2*l,1)=coordx((2*l-1),1)+(largo/2)-(interface/2);
        coordx((2*l-1)+2,1)=coordx(2*l,1)+interface;
        coordx(4*nladx+2,1)= coordx(4*nladx,1)+emorterox+difx;
    end
end
coordsapx=coordx;
nx=(length(coordx))+(length(coordx)-1)*(2^n-1); % número de nodos en x
para n cualquiera.
nxo=(length(coordx)); % número de nodos en x para n=0.
maux=zeros(length(coordx)-1,(2^n)+1);
for i=1:length(coordx)-1;
    maux(i,:)=linspace(coordx(i),coordx(i+1),(2^n)+1);
end
maux=maux(:,[2:end]);
coordx=[0 reshape(maux',1,prod(size(maux)))];
coordsapy=[0 prof];
coordz=zeros(1,2*nladz);
for i=1:nladz-1;
    coordz(1,2*i)=coordz(1,2*i-1)+alto;
    coordz(1,(2*i+1))=coordz(1,2*i)+emorteroz;
    coordz(1,nladz*2)=coordz(1,nladz*2-1)+alto;
end
coordsapz=coordz';
nyo=length(coordsapy);
nzo=length(coordsapz);
maux=zeros(length(coordz)-1,(2^n)+1);
for i=1:length(coordz)-1;
    maux(i,:)=linspace(coordz(i),coordz(i+1),(2^n)+1);
end
maux=maux(:,[2:end]);
coordz=[0 reshape(maux',1,prod(size(maux)))];
nz=length(coordz);
coordy=linspace(0,prof,(2^n)+1);
ny=length(coordy);
%
coordy=repmat(coordy,length(coordx)*length(coordz),1);
coordx=repmat(coordx,((2^n)+1)*length(coordz),1);
coordz=repmat(coordz,nx,((2^n)+1));
coordz=reshape(coordz,prod(size(coordz)),1);
coordy=reshape(coordy,prod(size(coordy)),1);
coord=[coordx coordy coordz]; % cm

```

```

coordsapxx=coord([1:nx],1);
coordsapyy=linspace(0,prof,(2^n)+1)';
coordsapzz=coord([1:nx:nx*nz],3);
n1=[];
n2=[];
n5=[];
n6=[];
for i=1:(nz-1);
    n1=[n1 ((1+(nx*(i-1))):(nx-1)+nx*(i-1))];
    n2=[n2 (2+(nx*(i-1))):(nx)+nx*(i-1)];
    n5=[n5 ((1+nx)+(nx*(i-1))):(2*nx-1)+nx*(i-1)];
    n6=[n6 ((2+nx)+(nx*(i-1))):(2*nx)+nx*(i-1)];
end
n1=repmat(n1,ny-1,1)+repmat([0:nx*nz:nx*nz*(ny-2)],length(n1),1);
n1=reshape(n1,prod(size(n1)),1);
n2=repmat(n2,ny-1,1)+repmat([0:nx*nz:nx*nz*(ny-2)],length(n2),1);
n2=reshape(n2,prod(size(n2)),1);
n3=[ones(length(n1),1)*nx*nz+n2];
n4=[ones(length(n1),1)*nx*nz+n1];
n5=repmat(n5,ny-1,1)+repmat([0:nx*nz:nx*nz*(ny-2)],length(n5),1);
n5=reshape(n5,prod(size(n5)),1);
n6=repmat(n6,ny-1,1)+repmat([0:nx*nz:nx*nz*(ny-2)],length(n6),1);
n6=reshape(n6,prod(size(n6)),1);
n7=[ones(length(n5),1)*nx*nz+n6];
n8=[ones(length(n5),1)*nx*nz+n5];
node=[n1 n2 n3 n4 n5 n6 n7 n8];
%% -----CLASIFICACIÓN DE ELEMENTOS SEGÚN MATERIAL-----
%LADRILLO
lad=[];
for i=0:nxo*0.5-1;
    lad=[lad linspace(1+(2^(n+1))*i,2^n+(2^(n+1))*i,2^n)];
end
lad=reshape(repmat(lad,1,2^n)',length(lad),2^n);
for i=1:2^n-1;
    lad(:,i+1)=lad(:,i)+(nx-1);
end
lad=repmat(reshape(lad,prod(size(lad)),1),1,nladz);
for i=1:nladz-1;
    lad(:,i+1)=lad(:,i)+2*lad(end,1);
end
lad=repmat(reshape(lad,prod(size(lad)),1),1,2^n);
for i=1:2^n-1;
    lad(:,i+1)=lad(:,i)+lad(end,1);
end
lad=reshape(lad,prod(size(lad)),1);

% INTER
inter=[];
for i=1:nxo/4;
    inter=[inter linspace((2^(n)+1)+2^(n+2)*(i-1),(2^(n+1))+2^(n+2)*(i-1),2^n)];
end
inter=repmat(inter',1,2^n);
for i=1:2^n-1;
    inter(:,i+1)=inter(:,i)+nx-1;
end
inter=reshape(inter,prod(size(inter)),1);

```

```

if largo*0.5<difx & difx<largo*0.7
    inter2=zeros(1,length(inter));
elseif largo*0.2<=difx & difx<=largo*0.5
    inter2=zeros(1,length(inter));
else
    inter2=zeros(1,length(inter)-(2^(2*n)));
end
inter2=[];
if largo*0.5<difx & difx<largo*0.7
for i=1:(nxo-2)/4;
    inter2=[inter2
linspace((1+2*(inter(end)+3*(2^n))+(2^n)*3)+2^(n+2)*(i-
1),((2^n))+2*(inter(end)+3*(2^n))+(2^n)*3)+2^(n+2)*(i-1),2^n)];
end
elseif largo*0.2<=difx & difx<=largo*0.5
for i=1:(nxo-2)/4;
    inter2=[inter2
linspace((1+2*(inter(end)+3*(2^n))+(2^n)*3)+2^(n+2)*(i-
1),((2^n))+2*(inter(end)+3*(2^n))+(2^n)*3)+2^(n+2)*(i-1),2^n)];
end
else
for i=1:(nxo/4)-1;
inter2=[inter2 linspace((1+2*(inter(end)+2^n)+(2^n)*3)+2^(n+2)*(i-
1),((2^n))+2*(inter(end)+2^n)+(2^n)*3)+2^(n+2)*(i-1),2^n)];
end
end
inter2= repmat(inter2',1,2^n);
for i=1:2^n-1;
    inter2(:,i+1)=inter2(:,i)+nx-1;
end
inter2=reshape(inter2,prod(size(inter2)),1);
inter=repmat(inter,1,ceil(nladz*0.5));
inter2=repmat(inter2,1,nladz-ceil(nladz*0.5));
if largo*0.5<difx & difx<largo*0.7
for i=1:size(inter,2)-1;
    inter(:,i+1)=inter(:,i)+((inter(end,1)+3*(2^n))*4);
end
elseif largo*0.2<=difx & difx<=largo*0.5
for i=1:size(inter,2)-1;
    inter(:,i+1)=inter(:,i)+((inter(end,1)+3*(2^n))*4);
end
else
for i=1:size(inter,2)-1;
    inter(:,i+1)=inter(:,i)+((inter(end,1)+(2^n))*4);
end
end
if size(inter2,2)==1
    inter2=inter2;
elseif largo*0.5<difx & difx<largo*0.7
for i=1:size(inter2,2)-1;
    inter2(:,i+1)=inter2(:,i)+((inter(end,1)+(3*(2^n)))*4);
end
elseif largo*0.2<=difx & difx<=largo*0.5
for i=1:size(inter2,2)-1;
    inter2(:,i+1)=inter2(:,i)+((inter(end,1)+(3*(2^n)))*4);
end
else

```

```

    for i=1:size(inter2,2)-1;
        inter2(:,i+1)=inter2(:,i)+((inter(end,1)+((2^n))*4);
    end
end

inter=reshape(inter,prod(size(inter)),1);
inter2=reshape(inter2,prod(size(inter2)),1);
inter=repmat(inter,1,2^n);
inter2=repmat(inter2,1,2^n);
for i=1:size(inter,2)-1;
    inter(:,i+1)=inter(:,i)+(lad(end)/(ny-1));
end
if size(inter2,2)==1
    inter2=inter2;
else
for i=1:size(inter,2)-1;
    inter2(:,i+1)=inter2(:,i)+lad(end)/(ny-1);
end
end
inter=reshape(inter,prod(size(inter)),1);
inter2=reshape(inter2,prod(size(inter2)),1);
inter=sort([inter' inter2'],'ascend');
elem=[1:length(node)]';
%MORTERO
mort=setdiff(elem,[lad' inter']');

```

7.1.3. Ploteo malla 3D y muro a color

```

%% -----PLOTEO MALLA Y MURO A COLOR-----
figure (13)
plot_malla3D( coord,node)
axis equal
figure (14)
cx=coord(:,1);
cy=coord(:,2);
cz=coord(:,3);
nodelad=node(lad,:);
nodeinter=node(inter,:);
nodemort=node(mort,:);
aux=[nodelad(:,1:4);nodelad(:,5:8);nodelad(:,1:2) nodelad(:,6)
nodelad(:,5);nodelad(:,4) nodelad(:,3) nodelad(:,7:8); nodelad(:,1)
nodelad(:,4) nodelad(:,8) nodelad(:,5); nodelad(:,2:3) nodelad(:,7)
nodelad(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),'r');
set(h,'EdgeColor','r')
view(-20.5, 50)
hold on
aux=[nodemort(:,1:4);nodemort(:,5:8);nodemort(:,1:2) nodemort(:,6)
nodemort(:,5);nodemort(:,4) nodemort(:,3) nodemort(:,7:8); nodemort(:,1)
nodemort(:,4) nodemort(:,8) nodemort(:,5); nodemort(:,2:3) nodemort(:,7)
nodemort(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.8);
set(h,'EdgeColor',[1 1 1]*0.8)
aux=[nodeinter(:,1:4);nodeinter(:,5:8);nodeinter(:,1:2) nodeinter(:,6)
nodeinter(:,5);nodeinter(:,4) nodeinter(:,3) nodeinter(:,7:8);
nodeinter(:,1) nodeinter(:,4) nodeinter(:,8) nodeinter(:,5);
nodeinter(:,2:3) nodeinter(:,7) nodeinter(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 0 0]*0.8);

```

```
set(h, 'EdgeColor', [1 0 0]*0.8)
axis equal
```

7.1.4. Clasificación de elementos según sus dimensiones y material

```
% -----CLASIFICACIÓN DE ELEMENTOS SEGÚN DIMENSIONES Y MATERIAL-----
if largo*0.2<=difx & difx<=largo*0.5;
    lad2=linspace((nx-2^n),(nx-1),2^n);
    lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,2^n);
    for i=1:2^n-1;
        lad2(:,i+1)=lad2(:,i)+(nx-1);
    end
    lad2=reshape(lad2,prod(size(lad2)),1);
    lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,nladz);
for i=1:nladz-1;
    lad2(:,i+1)=lad2(:,i)+2*lad2(end,1);
end
lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,2^n);
for i=1:2^n-1;
    lad2(:,i+1)=lad2(:,i)+lad2(end,1);
end
lad2=reshape(lad2,prod(size(lad2)),1);
lad1=setdiff(lad,lad2);
elseif difx>=largo*0.7;
    lad2=linspace((nx-2^n),(nx-1),2^n);
    lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,2^n);
    for i=1:2^n-1;
        lad2(:,i+1)=lad2(:,i)+(nx-1);
    end
    lad2=reshape(lad2,prod(size(lad2)),1);
    lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,nladz);
for i=1:nladz-1;
    lad2(:,i+1)=lad2(:,i)+2*lad2(end,1);
end
lad2= repmat(reshape(lad2,prod(size(lad2)),1),1,2^n);
for i=1:2^n-1;
    lad2(:,i+1)=lad2(:,i)+lad2(end,1);
end
lad2=reshape(lad2,prod(size(lad2)),1);
lad1=setdiff(lad,lad2);
else
    lad=lad;
end
%MORTERO
%MORTERO 4
if largo*0.2<=difx & difx<=largo*0.5;
    mort4=linspace((nx-2^n),(nx-1),2^n);
    mort4=mort4+(nx-1)*2^n);
    mort4= repmat(reshape(mort4,prod(size(mort4)),1),1,2^n);
    for i=1:2^n-1;
        mort4(:,i+1)=mort4(:,i)+(nx-1);
    end
    mort4=reshape(mort4,prod(size(mort4)),1);
    mort4= repmat(reshape(mort4,prod(size(mort4)),1),1,nladz-1);
for i=1:nladz-2;
    mort4(:,i+1)=mort4(:,i)+mort4(end,1);
end
mort4= repmat(reshape(mort4,prod(size(mort4)),1),1,2^n);
```

```

for i=1:2^n-1;
    mort4(:,i+1)=mort4(:,i)+(nx-1)*(nz-1);
end
mort4=reshape(mort4,prod(size(mort4)),1);
elseif difx>=largo*0.7;
    mort4=linspace((nx-2^n),(nx-1),2^n);
    mort4=mort4+(nx-1)*2^n;
    mort4=repmat(reshape(mort4,prod(size(mort4)),1),1,2^n);
    for i=1:2^n-1;
        mort4(:,i+1)=mort4(:,i)+(nx-1);
    end
    mort4=reshape(mort4,prod(size(mort4)),1);
    mort4=repmat(reshape(mort4,prod(size(mort4)),1),1,nladz-1);
for i=1:nladz-2;
    mort4(:,i+1)=mort4(:,i)+mort4(end,1);
end
mort4=repmat(reshape(mort4,prod(size(mort4)),1),1,2^n);
for i=1:2^n-1;
    mort4(:,i+1)=mort4(:,i)+(nx-1)*(nz-1);
end
mort4=reshape(mort4,prod(size(mort4)),1);
else
    mort=mort;
end
%MORTERO 2 Y MORTERO 3
if difx>=largo*0.7
    mort2=repmat(linspace((1+(nx-1)*2^n),(1+(nx-1)*2^n)+2^(n)-
1,2^n)',1,nladx*2);
    for i=1:nladx*2;
        mort2(:,i+1)=mort2(:,i)+2^(n+1);
    end
elseif largo*0.2<=difx & difx<=largo*0.5;
    mort2=repmat(linspace((1+(nx-1)*2^n),(1+(nx-1)*2^n)+2^(n)-
1,2^n)',1,nladx*2);
    for i=1:nladx*2-1;
        mort2(:,i+1)=mort2(:,i)+2^(n+1);
    end
elseif difx<largo*0.2
    mort2=repmat(linspace((1+(nx-1)*2^n),(1+(nx-1)*2^n)+2^(n)-
1,2^n)',1,nladx*2);
    for i=1:nladx*2-1;
        mort2(:,i+1)=mort2(:,i)+2^(n+1);
    end
else
    mort2=repmat(linspace((1+(nx-1)*2^n),(1+(nx-1)*2^n)+2^(n)-
1,2^n)',1,nladx*2);
    for i=1:nladx*2;
        mort2(:,i+1)=mort2(:,i)+2^(n+1);
    end
end
mort2=reshape(mort2,prod(size(mort2)),1);
if difx>=largo*0.7;
    mort3=[(1+(nx-1)*2^n):(nx-1)*(2^n+1)-3*(2^n)+2^(n+1)]';
else
    mort3=[(1+(nx-1)*2^n):(nx-1)*(2^n+1)-2^n]';
end
mort3=setdiff(mort3,mort2);

```

```

mort3=repmat (mort3,1,2^n);
mort2=repmat (mort2,1,2^n);
for i=1:2^n-1;
    mort2(:,i+1)=mort2(:,i)+(nx-1);
    mort3(:,i+1)=mort3(:,i)+(nx-1);
end
mort2=reshape (mort2,prod (size (mort2)),1);
mort2=repmat (reshape (mort2,prod (size (mort2)),1),1,nladz-1);
mort3=reshape (mort3,prod (size (mort3)),1);
mort3=repmat (reshape (mort3,prod (size (mort3)),1),1,nladz-1);
for i=1:nladz-2;
    mort2(:,i+1)=mort2(:,i)+(nx-1)*2^(n+1);
    mort3(:,i+1)=mort3(:,i)+(nx-1)*2^(n+1);
end
mort2=repmat (reshape (mort2,prod (size (mort2)),1),1,2^n);
mort3=repmat (reshape (mort3,prod (size (mort3)),1),1,2^n);
for i=1:2^n-1;
    mort2(:,i+1)=mort2(:,i)+(nx-1)*(nz-1);
    mort3(:,i+1)=mort3(:,i)+(nx-1)*(nz-1);
end
mort2=reshape (mort2,prod (size (mort2)),1);
mort3=reshape (mort3,prod (size (mort3)),1);
if largo*0.2<=difx & difx<=largo*0.5;
mort1=setdiff (mort,[mort2' mort3' mort4'])';
elseif difx>=largo*0.7;
mort1=setdiff (mort,[mort2' mort3' mort4'])';
elseif largo*0.5<difx & difx<largo*0.7;
mort1=setdiff (mort,[mort2' mort3' ])' ;
else
mort1=setdiff (mort,[mort2' mort3'])';
end
if largo*0.2<=difx & difx<=largo*0.5;
cx=coord (:,1);
cy=coord (:,2);
cz=coord (:,3);
nodelad1=node (lad1,:);
nodelad2=node (lad2,:);
nodeinter=node (inter,:);
nodemort1=node (mort1,:);
nodemort2=node (mort2,:);
nodemort3=node (mort3,:);
nodemort4=node (mort4,:);
aux=[nodelad1 (:,1:4);nodelad1 (:,5:8);nodelad1 (:,1:2) nodelad1 (:,6)
nodelad1 (:,5);nodelad1 (:,4) nodelad1 (:,3) nodelad1 (:,7:8); nodelad1 (:,1)
nodelad1 (:,4) nodelad1 (:,8) nodelad1 (:,5); nodelad1 (:,2:3) nodelad1 (:,7)
nodelad1 (:,6)];
figure (15)
h=patch (cx (aux'),cy (aux'),cz (aux'),'r');
set (h,'EdgeColor','r')
view (-20.5, 50)
hold on
aux=[nodelad2 (:,1:4);nodelad2 (:,5:8);nodelad2 (:,1:2) nodelad2 (:,6)
nodelad2 (:,5);nodelad2 (:,4) nodelad2 (:,3) nodelad2 (:,7:8); nodelad2 (:,1)
nodelad2 (:,4) nodelad2 (:,8) nodelad2 (:,5); nodelad2 (:,2:3) nodelad2 (:,7)
nodelad2 (:,6)];
h=patch (cx (aux'),cy (aux'),cz (aux'),[1 1 0]*0.5);
set (h,'EdgeColor',[1 1 0]*0.5)

```

```

aux=[nodemort1(:,1:4);nodemort1(:,5:8);nodemort1(:,1:2) nodemort1(:,6)
nodemort1(:,5);nodemort1(:,4) nodemort1(:,3) nodemort1(:,7:8);
nodemort1(:,1) nodemort1(:,4) nodemort1(:,8) nodemort1(:,5);
nodemort1(:,2:3) nodemort1(:,7) nodemort1(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.8);
set(h,'EdgeColor',[1 1 1]*0.8)
aux=[nodemort2(:,1:4);nodemort2(:,5:8);nodemort2(:,1:2) nodemort2(:,6)
nodemort2(:,5);nodemort2(:,4) nodemort2(:,3) nodemort2(:,7:8);
nodemort2(:,1) nodemort2(:,4) nodemort2(:,8) nodemort2(:,5);
nodemort2(:,2:3) nodemort2(:,7) nodemort2(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.5);
set(h,'EdgeColor',[1 1 1]*0.5)
aux=[nodemort3(:,1:4);nodemort3(:,5:8);nodemort3(:,1:2) nodemort3(:,6)
nodemort3(:,5);nodemort3(:,4) nodemort3(:,3) nodemort3(:,7:8);
nodemort3(:,1) nodemort3(:,4) nodemort3(:,8) nodemort3(:,5);
nodemort3(:,2:3) nodemort3(:,7) nodemort3(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.2);
set(h,'EdgeColor',[1 1 1]*0.2)
aux=[nodemort4(:,1:4);nodemort4(:,5:8);nodemort4(:,1:2) nodemort4(:,6)
nodemort4(:,5);nodemort4(:,4) nodemort4(:,3) nodemort4(:,7:8);
nodemort4(:,1) nodemort4(:,4) nodemort4(:,8) nodemort4(:,5);
nodemort4(:,2:3) nodemort4(:,7) nodemort4(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.95);
set(h,'EdgeColor',[1 1 1]*0.95)
aux=[nodeinter(:,1:4);nodeinter(:,5:8);nodeinter(:,1:2) nodeinter(:,6)
nodeinter(:,5);nodeinter(:,4) nodeinter(:,3) nodeinter(:,7:8);
nodeinter(:,1) nodeinter(:,4) nodeinter(:,8) nodeinter(:,5);
nodeinter(:,2:3) nodeinter(:,7) nodeinter(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 0 0]*0.8);
set(h,'EdgeColor',[1 0 0]*0.8)
axis equal
elseif difx>=0.7*largo;
cx=coord(:,1);
cy=coord(:,2);
cz=coord(:,3);
nodelad1=node(lad1,:);
nodelad2=node(lad2,:);
nodeinter=node(inter,:);
nodemort1=node(mort1,:);
nodemort2=node(mort2,:);
nodemort3=node(mort3,:);
nodemort4=node(mort4,:);
aux=[nodelad1(:,1:4);nodelad1(:,5:8);nodelad1(:,1:2) nodelad1(:,6)
nodelad1(:,5);nodelad1(:,4) nodelad1(:,3) nodelad1(:,7:8); nodelad1(:,1)
nodelad1(:,4) nodelad1(:,8) nodelad1(:,5); nodelad1(:,2:3) nodelad1(:,7)
nodelad1(:,6)];
figure (15)
h=patch(cx(aux'),cy(aux'),cz(aux'),'r');
set(h,'EdgeColor','r')
view(-20.5, 50)
hold on
aux=[nodelad2(:,1:4);nodelad2(:,5:8);nodelad2(:,1:2) nodelad2(:,6)
nodelad2(:,5);nodelad2(:,4) nodelad2(:,3) nodelad2(:,7:8); nodelad2(:,1)
nodelad2(:,4) nodelad2(:,8) nodelad2(:,5); nodelad2(:,2:3) nodelad2(:,7)
nodelad2(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 0]*0.5);
set(h,'EdgeColor',[1 1 0]*0.5)

```

```

aux=[nodemort1(:,1:4);nodemort1(:,5:8);nodemort1(:,1:2) nodemort1(:,6)
nodemort1(:,5);nodemort1(:,4) nodemort1(:,3) nodemort1(:,7:8);
nodemort1(:,1) nodemort1(:,4) nodemort1(:,8) nodemort1(:,5);
nodemort1(:,2:3) nodemort1(:,7) nodemort1(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.8);
set(h,'EdgeColor',[1 1 1]*0.8)
aux=[nodemort2(:,1:4);nodemort2(:,5:8);nodemort2(:,1:2) nodemort2(:,6)
nodemort2(:,5);nodemort2(:,4) nodemort2(:,3) nodemort2(:,7:8);
nodemort2(:,1) nodemort2(:,4) nodemort2(:,8) nodemort2(:,5);
nodemort2(:,2:3) nodemort2(:,7) nodemort2(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.5);
set(h,'EdgeColor',[1 1 1]*0.5)
aux=[nodemort3(:,1:4);nodemort3(:,5:8);nodemort3(:,1:2) nodemort3(:,6)
nodemort3(:,5);nodemort3(:,4) nodemort3(:,3) nodemort3(:,7:8);
nodemort3(:,1) nodemort3(:,4) nodemort3(:,8) nodemort3(:,5);
nodemort3(:,2:3) nodemort3(:,7) nodemort3(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.2);
set(h,'EdgeColor',[1 1 1]*0.2)
aux=[nodemort4(:,1:4);nodemort4(:,5:8);nodemort4(:,1:2) nodemort4(:,6)
nodemort4(:,5);nodemort4(:,4) nodemort4(:,3) nodemort4(:,7:8);
nodemort4(:,1) nodemort4(:,4) nodemort4(:,8) nodemort4(:,5);
nodemort4(:,2:3) nodemort4(:,7) nodemort4(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.95);
set(h,'EdgeColor',[1 1 1]*0.95)
aux=[nodeinter(:,1:4);nodeinter(:,5:8);nodeinter(:,1:2) nodeinter(:,6)
nodeinter(:,5);nodeinter(:,4) nodeinter(:,3) nodeinter(:,7:8);
nodeinter(:,1) nodeinter(:,4) nodeinter(:,8) nodeinter(:,5);
nodeinter(:,2:3) nodeinter(:,7) nodeinter(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 0 0]*0.8);
set(h,'EdgeColor',[1 0 0]*0.8)
axis equal
else
cx=coord(:,1);
cy=coord(:,2);
cz=coord(:,3);
nodelad=node(lad,:);
nodeinter=node(inter,:);
nodemort1=node(mort1,:);
nodemort2=node(mort2,:);
nodemort3=node(mort3,:);
aux=[nodelad(:,1:4);nodelad(:,5:8);nodelad(:,1:2) nodelad(:,6)
nodelad(:,5);nodelad(:,4) nodelad(:,3) nodelad(:,7:8); nodelad(:,1)
nodelad(:,4) nodelad(:,8) nodelad(:,5); nodelad(:,2:3) nodelad(:,7)
nodelad(:,6)];
figure (15)
h=patch(cx(aux'),cy(aux'),cz(aux'),'r');
set(h,'EdgeColor','r')
view(-20.5, 50)
hold on
aux=[nodemort1(:,1:4);nodemort1(:,5:8);nodemort1(:,1:2) nodemort1(:,6)
nodemort1(:,5);nodemort1(:,4) nodemort1(:,3) nodemort1(:,7:8);
nodemort1(:,1) nodemort1(:,4) nodemort1(:,8) nodemort1(:,5);
nodemort1(:,2:3) nodemort1(:,7) nodemort1(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.8);
set(h,'EdgeColor',[1 1 1]*0.8)
aux=[nodemort2(:,1:4);nodemort2(:,5:8);nodemort2(:,1:2) nodemort2(:,6)
nodemort2(:,5);nodemort2(:,4) nodemort2(:,3) nodemort2(:,7:8);

```

```

nodemort2(:,1) nodemort2(:,4) nodemort2(:,8) nodemort2(:,5);
nodemort2(:,2:3) nodemort2(:,7) nodemort2(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.6);
set(h,'EdgeColor',[1 1 1]*0.6)
aux=[nodemort3(:,1:4);nodemort3(:,5:8);nodemort3(:,1:2) nodemort3(:,6)
nodemort3(:,5);nodemort3(:,4) nodemort3(:,3) nodemort3(:,7:8);
nodemort3(:,1) nodemort3(:,4) nodemort3(:,8) nodemort3(:,5);
nodemort3(:,2:3) nodemort3(:,7) nodemort3(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 1 1]*0.4);
set(h,'EdgeColor',[1 1 1]*0.4)
aux=[nodeinter(:,1:4);nodeinter(:,5:8);nodeinter(:,1:2) nodeinter(:,6)
nodeinter(:,5);nodeinter(:,4) nodeinter(:,3) nodeinter(:,7:8);
nodeinter(:,1) nodeinter(:,4) nodeinter(:,8) nodeinter(:,5);
nodeinter(:,2:3) nodeinter(:,7) nodeinter(:,6)];
h=patch(cx(aux'),cy(aux'),cz(aux'),[1 0 0]*0.8);
set(h,'EdgeColor',[1 0 0]*0.8)
axis equal
end
%% DIMENSIONES LOCALES DE LOS ELEMENTOS
if largo*0.2<=difx & difx<=largo*0.5;

aux=[node(lad1(1),:);node(lad2(1),:);node(mort1(1),:);node(mort2(1),:);no
de(mort3(1),:);node(mort4(1),:);node(inter(1),:)];
elseif difx>=0.7*largo;

aux=[node(lad1(1),:);node(lad2(1),:);node(mort1(1),:);node(mort2(1),:);no
de(mort3(1),:);node(mort4(1),:);node(inter(1),:)];
else

aux=[node(lad(1),:);node(mort1(1),:);node(mort2(1),:);node(mort3(1),:);no
de(inter(1),:)];
end
aux1=cx(aux)/100;
a=aux1(:,2)-aux1(:,1);
aux2=cy(aux)/100;
b=aux2(:,3)-aux2(:,2);
aux3=cz(aux)/100;
c=aux3(:,5)-aux3(:,1);

```

7.1.5. Cálculo de Klocal y Mlocal

```

%%-----Cálculo N, B Klocal y Mlocal-----
D=[];
for i=1:3;
e1=v(i)/(1-v(i));
e2=(1-2*v(i))/(2-2*v(i));
D=[D (E(i)*(1-v(i))/((1+v(i))*(1-2*v(i))))*[1 e1 e1 0 0 0;e1 1 e1 0 0
0;e1 e1 1 0 0 0;0 0 0 e2 0 0;0 0 0 0 e2 0;0 0 0 0 0 e2]];
coordloc=[-1 -1 -1;1 -1 -1;1 1 -1;-1 1 -1;-1 -1 1;1 -1 1;1 1 1;-1 1 1];
end
B=[];
N=sym(zeros(3,24));
aux=1:3:24;
syms x y z real
for j=1:length(a);
for i=1:8
Ni=(1/8)*[(1+coordloc(i,1)*x)*(1+coordloc(i,2)*y)*(1+coordloc(i,3)*z)];

```

```

        Nii=[Ni 0 0;0 Ni 0;0 0 Ni];
        N(:,(aux(i):aux(i)+2))=Nii;
        B=[B [diff(Ni,x)/(a(j)/2) 0 0; 0 diff(Ni,y)/(b(j)/2) 0; 0 0
diff(Ni,z)/(c(j)/2);diff(Ni,y)/(b(j)/2) diff(Ni,x)/(a(j)/2)
0;diff(Ni,z)/(c(j)/2) 0 diff(Ni,x)/(a(j)/2); 0 diff(Ni,z)/(c(j)/2)
diff(Ni,y)/(b(j)/2)]];

end
end
if largo*0.2<=difx & difx<=largo*0.5;
    B1=B(:,1:24);
    B2=B(:,25:48);
    B3=B(:,49:72);
    B4=B(:,73:96);
    B5=B(:,97:120);
    B6=B(:,121:144);
    B7=B(:,145:168);

Klocal1=vpa(int(int(int(B1'*D(:,1:6)*B1*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal2=vpa(int(int(int(B2'*D(:,1:6)*B2*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal3=vpa(int(int(int(B3'*D(:,7:12)*B3*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal4=vpa(int(int(int(B4'*D(:,7:12)*B4*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal5=vpa(int(int(int(B5'*D(:,7:12)*B5*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal6=vpa(int(int(int(B6'*D(:,7:12)*B6*(a(6)/2)*(b(6)/2)*(c(6)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal7=vpa(int(int(int(B7'*D(:,13:18)*B7*(a(7)/2)*(b(7)/2)*(c(7)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Mlocal1=vpa(Densidad(1)*(int(int(int(N'*N*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal2=vpa(Densidad(1)*(int(int(int(N'*N*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal3=vpa(Densidad(2)*(int(int(int(N'*N*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal4=vpa(Densidad(2)*(int(int(int(N'*N*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal5=vpa(Densidad(2)*(int(int(int(N'*N*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal6=vpa(Densidad(2)*(int(int(int(N'*N*(a(6)/2)*(b(6)/2)*(c(6)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

```

```

Mlocal7=vpa(Densidad(3)*(int(int(int(N'*N*(a(7)/2)*(b(7)/2)*(c(7)/2),x,-
1,1),y,-1,1),z,-1,1)),2);
elseif difx>=0.7*largo;
    B1=B(:,1:24);
    B2=B(:,25:48);
    B3=B(:,49:72);
    B4=B(:,73:96);
    B5=B(:,97:120);
    B6=B(:,121:144);
    B7=B(:,145:168);

Klocal1=vpa(int(int(int(B1'*D(:,1:6)*B1*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal2=vpa(int(int(int(B2'*D(:,1:6)*B2*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal3=vpa(int(int(int(B3'*D(:,7:12)*B3*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal4=vpa(int(int(int(B4'*D(:,7:12)*B4*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal5=vpa(int(int(int(B5'*D(:,7:12)*B5*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal6=vpa(int(int(int(B6'*D(:,7:12)*B6*(a(6)/2)*(b(6)/2)*(c(6)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal7=vpa(int(int(int(B7'*D(:,13:18)*B7*(a(7)/2)*(b(7)/2)*(c(7)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Mlocal1=vpa(Densidad(1)*(int(int(int(N'*N*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal2=vpa(Densidad(1)*(int(int(int(N'*N*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal3=vpa(Densidad(2)*(int(int(int(N'*N*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal4=vpa(Densidad(2)*(int(int(int(N'*N*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal5=vpa(Densidad(2)*(int(int(int(N'*N*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal6=vpa(Densidad(2)*(int(int(int(N'*N*(a(6)/2)*(b(6)/2)*(c(6)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal7=vpa(Densidad(3)*(int(int(int(N'*N*(a(7)/2)*(b(7)/2)*(c(7)/2),x,-
1,1),y,-1,1),z,-1,1)),2);
else
    B1=B(:,1:24);
    B2=B(:,25:48);
    B3=B(:,49:72);

```

```

B4=B(:,73:96);
B5=B(:,97:120);

Klocal1=vpa(int(int(int(B1'*D(:,1:6)*B1*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal2=vpa(int(int(int(B2'*D(:,7:12)*B2*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal3=vpa(int(int(int(B3'*D(:,7:12)*B3*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal4=vpa(int(int(int(B4'*D(:,7:12)*B4*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Klocal5=vpa(int(int(int(B5'*D(:,13:18)*B5*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1),3);

Mlocal1=vpa(Densidad(1)*(int(int(int(N'*N*(a(1)/2)*(b(1)/2)*(c(1)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal2=vpa(Densidad(2)*(int(int(int(N'*N*(a(2)/2)*(b(2)/2)*(c(2)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal3=vpa(Densidad(2)*(int(int(int(N'*N*(a(3)/2)*(b(3)/2)*(c(3)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal4=vpa(Densidad(2)*(int(int(int(N'*N*(a(4)/2)*(b(4)/2)*(c(4)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

Mlocal5=vpa(Densidad(3)*(int(int(int(N'*N*(a(5)/2)*(b(5)/2)*(c(5)/2),x,-
1,1),y,-1,1),z,-1,1)),2);

```

7.1.6. Ensamble de Klocal y Mlocal

```

end%%-----ENSAMBLE-----
Kglobal=sparse(3*length(coord),3*length(coord));
Mglobal=sparse(3*length(coord),3*length(coord));
gdl=[1:3:(3*nx*ny*nz)-2]',[2:3:(3*nx*ny*nz)-1]',[3:3:(3*nx*ny*nz)]'];
if largo*0.2<=difx & difx<=largo*0.5;
    for i=1:size(lad1,1);
        gdl2=reshape(gdl(node(lad1(i),:),:)',1,24);
        Kglobal(gdl2,gdl2)=Kglobal(gdl2,gdl2)+ Klocal1;
        Mglobal(gdl2,gdl2)=Mglobal(gdl2,gdl2)+ Mlocal1;
    end
    for i=1:size(lad2,1);
        gdl2=reshape(gdl(node(lad2(i),:),:)',1,24);
        Kglobal(gdl2,gdl2)=Kglobal(gdl2,gdl2)+ Klocal2;
        Mglobal(gdl2,gdl2)=Mglobal(gdl2,gdl2)+ Mlocal2;
    end
    for i=1:size(mort1,1);
        gdl2=reshape(gdl(node(mort1(i),:),:)',1,24);
        Kglobal(gdl2,gdl2)=Kglobal(gdl2,gdl2)+ Klocal3;
        Mglobal(gdl2,gdl2)=Mglobal(gdl2,gdl2)+ Mlocal3;
    end
    for i=1:size(mort2,1);
        gdl2=reshape(gdl(node(mort2(i),:),:)',1,24);
        Kglobal(gdl2,gdl2)=Kglobal(gdl2,gdl2)+ Klocal4;

```

```

Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal4;
end
for i=1:size(mort3,1);
gdl2=reshape (gdl (node (mort3 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal5;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal5;
end
for i=1:size(mort4,1);
gdl2=reshape (gdl (node (mort4 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal6;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal6;
end
for i=1:size(inter,1);
gdl2=reshape (gdl (node (inter (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal7;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal7;
end
elseif difx>=0.7*largo;
for i=1:size(lad1,1);
gdl2=reshape (gdl (node (lad1 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal1;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal1;
end
for i=1:size(lad2,1);
gdl2=reshape (gdl (node (lad2 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal2;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal2;
end
for i=1:size(mort1,1);
gdl2=reshape (gdl (node (mort1 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal3;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal3;
end
for i=1:size(mort2,1);
gdl2=reshape (gdl (node (mort2 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal4;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal4;
end
for i=1:size(mort3,1);
gdl2=reshape (gdl (node (mort3 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal5;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal5;
end
for i=1:size(mort4,1);
gdl2=reshape (gdl (node (mort4 (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal6;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal6;
end
for i=1:size(inter,1);
gdl2=reshape (gdl (node (inter (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal7;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal7;
end
else
for i=1:size(lad,1);
gdl2=reshape (gdl (node (lad (i) , :), :)' ,1,24);
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal1;

```

```

Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal1;
end
for i=1:size(mort1,1);
gdl2=reshape (gdl (node (mort1 (i) , : ) , : ) ' , 1 , 24 ) ;
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal2;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal2;
end
for i=1:size(mort2,1);
gdl2=reshape (gdl (node (mort2 (i) , : ) , : ) ' , 1 , 24 ) ;
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal3;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal3;
end
for i=1:size(mort3,1);
gdl2=reshape (gdl (node (mort3 (i) , : ) , : ) ' , 1 , 24 ) ;
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal4;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal4;
end
for i=1:size(inter,1);
gdl2=reshape (gdl (node (inter (i) , : ) , : ) ' , 1 , 24 ) ;
Kglobal (gdl2,gdl2)=Kglobal (gdl2,gdl2)+ Klocal5;
Mglobal (gdl2,gdl2)=Mglobal (gdl2,gdl2)+ Mlocal5;
end
end

```

7.1.7. Condiciones de borde asignadas

```

%%-----CONDICIONES DE BORDE-----
c1=repmat ([1:nx], 2^n+1, 1) ' ;
for i=1:2^n;
    c1 (:, i+1)=c1 (:, i)+nx*nz;
end
c1=reshape (c1, prod (size (c1)) , 1) ;
c1=gdl (c1, : ) ;
c1=reshape (c1, prod (size (c1)) , 1) ' ; % extremo inferior fijo
c2=repmat ([nx*nz-(nx-1) : nx*nz], 2^n+1, 1) ' ;
for i=1:2^n;
    c2 (:, i+1)=c2 (:, i)+nx*nz;
end
c2=reshape (c2, prod (size (c2)) , 1) ;
c2=gdl (c2, : ) ;
c2=reshape (c2, prod (size (c2)) , 1) ' ; % extremo superior fijo
c3=repmat (nx, 1, nz) ;
for i=1:length (c3)-1;
    c3 (1, i+1)=c3 (1, i)+nx;
end
c3=repmat (c3, 2^n+1, 1) ' ;
for i=1:2^n;
    c3 (:, i+1)=c3 (:, i)+nx*nz;
end
c3=reshape (c3, prod (size (c3)) , 1) ;
c3=gdl (c3, : ) ;
c3=reshape (c3, prod (size (c3)) , 1) ' ; % extremo derecho fijo
c4=repmat (1, 1, nz) ;
for i=1:length (c4)-1;
    c4 (1, i+1)=c4 (1, i)+nx;
end
c4=repmat (c4, 2^n+1, 1) ' ;
for i=1:2^n;

```

```

        c4(:,i+1)=c4(:,i)+nx*nz;
end
c4=reshape(c4,prod(size(c4)),1);
c4=gdl(c4,:);
c4=reshape(c4,prod(size(c4)),1)';% extremo izquierdo fijo

u=zeros(3*nx*ny*nz,1);
F=sort([c1 ],'ascend');
uF=zeros(length(F),1);
L=setdiff([1:3*nx*ny*nz],F);

Kglobal(F,:)=[];
Kglobal(:,F)=[];
Mglobal(F,:)=[];
Mglobal(:,F)=[];
Frecuencias=sort(vpa(sqrt(eigs(Kglobal,Mglobal,12,'sm'))),'ascend');
Periodonatural=sort(vpa((2*pi())./Frecuencias),'descend');
FRE=1./Periodonatural;

```

7.1.8. Cálculo de Frecuencias y matriz de formas modales

```

%-----valores y vectores propios-----
[V,D]=eigs(Kglobal,Mglobal,12,'sm');
u=zeros(3*nx*ny*nz,1);
for i=1:12
figure(i)

% modo=V(:,i);
u(L)=V(:,i);
% for t=0:.5:50;
%     coord=coord+10*sin(t)*reshape(u,3,nx*ny*nz)';
%     plot_malla3D(coord,node);
%     axis([-210 210 -25 25 -210 210])
%     1
%     pause(0.01)
%     cla
%     grid on
% end

coord2=coord+500*reshape(u,3,nx*ny*nz)';
plot_malla3D(coord2,node);
axis equal
end

```

7.2. Código programado en Matlab, para calcular las correlaciones de datos de la matriz de modos ocupando el criterio de certeza modal MAC

```

clear all; clc
x=xlsread('Modos de vibrar sap.xlsx');
coordsap=xlsread('Modos de vibrar sap.xlsx','hoja1');
coordsap2=coordsap+0.0001;
coordsap1=coordsap-0.0001;
load('100x100E.mat','V','coord','F','L','L','nx','ny','nz','node');
coord=coord*0.01;
P=[];
for i=1:length(coord);
    for j=1:length(coord);

```

```

    if coord(i,:) <= coordsap2(j,:) & coordsap1(j,:) <= coord(i,:);
        P=[P j];
    else
        Po=0;
    end
end
end
x=x(P,:);
VS=[];
for i=1:12;
    a=reshape(x(:, [(3*i-2):3*i])', prod(size(x(:, [(3*i-2):3*i]))), 1);
    a=a(setdiff([1:length(a)], find(a==0)));
    VS=[VS a];
end
gdll=[ [1:3:(3*length(coord))-2]', [2:3:(3*length(coord))-1]', [3:3:(3*length(coord))]'];
mac=zeros(12,12);
for i=1:12;
    for j=1:12;
        mac(i,j)=mac(i,j)+(abs(V(:,13-j)'*VS(:,i))^2/((V(:,13-j)'*(V(:,13-j)))*VS(:,i)'*VS(:,i))));
    end
end
u=zeros(3*nx*ny*nz,1);
for i=1:12
figure(i)

% modo=V(:,i);
u(L)=VS(:,i);
% for t=0:.5:50;
%     coord=coord+10*sin(t)*reshape(u,3,nx*ny*nz)';
%     plot_malla3D(coord,node);
%     axis([-210 210 -25 25 -210 210])
%     1
%     pause(0.01)
%     cla
%     grid on
% end

coord2=coord+reshape(u,3,nx*ny*nz)';
plot_malla3D(coord2,node);
axis equal
end

```