

Desarrollo de un programa didáctico de elementos finitos para problemas elásticos lineales en tres dimensiones

Alejandro Robayo Zapata

Universidad Nacional de Colombia
Facultad de Ingeniería, Departamento de Ingeniería Civil y Agrícola
Bogotá, Colombia
2020

Desarrollo de un programa didáctico de elementos finitos para problemas elásticos lineales en tres dimensiones

Alejandro Robayo Zapata

Trabajo de grado presentado como requisito parcial para optar al título de:
Magíster en Ingeniería - Estructuras

Director:

PhD Dorian Luis Linero Segrera

Codirector:

PhD Martín Estrada Mejía

Línea de Investigación:

Análisis Estructural

Grupo de Investigación:

GIEs. Análisis, Diseño y Materiales

Universidad Nacional de Colombia

Facultad de Ingeniería, Departamento de ingeniería Civil y Agrícola

Bogotá, Colombia

2020

Dedico este trabajo a mi hijo Manuel,
por enseñarme tantas cosas.

Mis más sinceros agradecimientos a las siguientes personas:

Al profesor Dorian Linero, director del trabajo de grado,
por su dedicación y paciencia para enseñarme los conceptos
y las herramientas que me permitieron realizar este trabajo.

Al profesor Martín Estrada, codirector del trabajo de grado,
por sus valiosos consejos y oportunos aportes.

A mi esposa Mery, mi hija Mariana, mi sobrino Felipe
y a mi mamá por su confianza y apoyo incondicional.

Resumen

Desarrollo de un programa didáctico de elementos finitos para problemas elásticos lineales en tres dimensiones

En el presente trabajo final de maestría se desarrolla y documenta un programa didáctico de elementos finitos a código abierto escrito en pseudo lenguaje matemático, capaz de simular el proceso de análisis de un problema mecánico elástico lineal en tres dimensiones bajo cargas estáticas. Particularmente, se implementó el elemento finito hexaédrico de ocho nudos y el elemento tipo prisma de 6 nudos.

La finalidad de este trabajo es apoyar el aprendizaje, la investigación y la profundización en el análisis estructural de sólidos tridimensionales mediante el método de los elementos finitos.

El programa desarrollado es una extensión del programa de elementos finitos a código abierto PEFiCA 2.0, en el cual se implementaron nuevas rutinas en GNU-Octave, relacionadas con el proceso de cálculo y con el pre y postproceso gráfico realizado con GMSH.

Se simularon dos ejemplos de aplicación utilizando el programa desarrollado, cuyos resultados como el desplazamiento máximo y los esfuerzos principales, fueron similares a los obtenidos con el programa comercial ANSYS. Lo anterior verifica el correcto funcionamiento del programa desarrollado.

Palabras Clave

Método de elementos finitos, Análisis mecánico estático elástico tridimensional, Programación didáctica a código abierto, Elemento finito hexaédrico y sus simplificaciones.

Abstract

Development of a finite element didactic program for linear elastic problems in three dimensions

In this final master's work, an open source finite element didactic program written in pseudo-mathematical language is developed and documented, capable of simulating the analysis process of a linear mechanical elastic problem in three dimensions under static loads. In particular, the eight-node hexahedral finite element and the six-node prism-type element were implemented.

The purpose of this work is to support learning, research and deepening in the structural analysis of three-dimensional solids through the finite element method. The developed program is an extension of the open source finite element program PEFiCA 2.0, in which new routines were implemented in GNU-Octave, related to the calculation process and the graphic pre- and post-processing made with GMSH.

Two application examples were simulated using the program developed, whose results, such as maximum displacement and main stresses, were similar to those obtained with the commercial program ANSYS. This verifies the correct operation of the developed program.

Keywords

Finite element method, Three-dimensional static-elastic mechanical analysis, Open source didactic programming, Hexahedral finite element and its simplifications.

Índice general

1. Introducción	1
1.1. Motivación	1
1.2. Objetivo General	2
1.3. Objetivos Específicos	3
1.4. Metodología general	3
2. Antecedentes	5
2.1. Estructura básica de los programas de elementos finitos	5
2.1.1. Preproceso	5
2.1.2. Proceso	5
2.1.3. Postproceso	6
2.2. Programas didácticos y de aprendizaje	6
2.2.1. MATFEM	6
2.2.2. CALFEM	7
2.2.3. PEFiCA	7
3. Marco teórico	9
3.1. Generalidades del método de los elementos finitos	9
3.1.1. Funciones de aproximación y funciones de forma	10
3.1.2. Principio del Trabajo Virtual	10
3.2. Formulación del método de los elementos finitos para problemas de la mecánica estructural lineales tridimensionales	11
3.2.1. Matriz de rigidez y vector de fuerzas equivalentes en el sólido	12
3.2.2. Metodología para el cálculo del vector de desplazamientos nodales en el sólido	12
3.2.3. Vector de fuerzas totales y reacciones en el sólido	13
3.2.4. Extracción de los componentes de desplazamiento, deformación y esfuerzo en el interior de cada elemento	14
3.3. Valores característicos de esfuerzo y de deformación	14

3.4.	Criterio de von Mises	15
3.5.	Valores promedio de esfuerzo y deformación en los nudos de la malla .	15
3.6.	Formulación del método de elementos finitos para el elemento hexaédrico de 8 nudos	16
3.6.1.	Numeración de los nudos	18
3.6.2.	Coordenadas naturales	18
3.6.3.	Funciones de forma	19
3.6.4.	Matriz de operadores diferenciales con respecto al sistema natural actuando sobre funciones de forma	23
3.6.5.	Matriz Jacobiana y su inversa	24
3.6.6.	Matriz de operadores diferenciales actuando sobre las funciones de forma	26
3.6.7.	Matriz constitutiva del material elástico lineal isotrópico	28
3.6.8.	Matriz de rigidez del hexaedro	28
3.6.9.	Vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento	30
3.6.10.	Vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global	32
3.6.11.	Vector de fuerzas equivalentes a la acción de una presión uniforme normal a una cara del elemento	34
3.6.12.	Vector de fuerzas equivalentes a la acción de una presión hidrostática	36
3.6.13.	Vector de fuerzas aplicadas directamente en los nudos de la malla	37
3.7.	Formulación del método de los elementos finitos para el elemento prisma o cuña	37
3.7.1.	Numeración de los nudos	37
3.7.2.	Coordenadas naturales	38
3.7.3.	Funciones de forma	39
3.7.4.	Matriz de operadores diferenciales con respecto al sistema coordenado natural actuando sobre funciones de forma	41
3.7.5.	Matriz Jacobiana y su inversa	42
3.7.6.	Matriz de operadores diferenciales actuando sobre las funciones de forma	42
3.7.7.	Matriz de rigidez del elemento finito	43
3.7.8.	Vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento finito	45
3.7.9.	Vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global	45

3.7.10. Vector de fuerzas equivalentes a la acción de una presión uniforme normal a una cara del elemento	47
3.7.11. Vector de fuerzas equivalentes a la acción de una presión hidrostática	48
4. Implementación del programa de computador	49
4.1. Descripción de la estructura general del programa desarrollado a partir de PEFiCA 2.0	49
4.2. Alcance del programa implementado	52
4.3. Construcción del archivo de datos de entrada mediante el entorno gráfico de GMSH	54
4.4. Lectura del archivo de datos de entrada generado en el preproceso	54
4.5. Rutinas modificadas de PEFiCA 2.0 en la etapa de proceso	56
4.5.1. Función PEFiCA	57
4.5.2. Función KELEME	58
4.5.3. Función BELEME	60
4.5.3.1. Función FELEMC	62
4.5.4. Función FELEMS	64
4.5.5. Función PBNUEL	67
4.5.6. Función PBPGAU	67
4.6. Rutinas nuevas en la etapa de proceso	69
4.6.1. Función JELEME	69
4.6.2. Función NELEME	70
4.7. Creación del archivo de postproceso	70
5. Ejemplos de aplicación y análisis de resultados	73
5.1. Ejemplo 1: Presa de bóveda sometida a la presión hidráulica y su peso propio	74
5.1.1. Creación de las mallas de elementos finitos	74
5.1.2. Solución del problema utilizando el programa desarrollado	77
5.1.3. Resultados	77
5.1.4. Análisis de resultados	79
5.1.5. Análisis de sensibilidad	80
5.2. Ejemplo 2: Ensayo de resistencia al arrancamiento del concreto	81
5.2.1. Creación de las mallas de elementos finitos	83
5.2.2. Solución del problema	84
5.2.3. Resultados	84
5.2.4. Análisis de resultados	88

5.2.5.	Análisis de sensibilidad	89
5.3.	Análisis de costo computacional	89
5.3.1.	Comparación del costo computacional por tipo de elemento	89
5.3.2.	Comparación del costo computacional de acuerdo al método de solución de ecuaciones simultáneas	91
6.	Conclusiones y recomendaciones	93
6.1.	Conclusiones	93
6.2.	Recomendaciones para futuros trabajos	94
7.	Manual del usuario del programa	97
7.1.	Creación de la geometría y el enmallado en el programa GMSH	97
7.1.1.	Consideraciones generales	100
7.1.2.	Construcción de entidades geométricas	100
7.1.2.1.	Construcción de puntos	101
7.1.2.2.	Construcción de líneas	102
7.1.2.3.	Construcción de superficies	103
7.1.2.4.	Construcción de volúmenes	103
7.1.3.	Construcción de formas comunes	104
7.1.3.1.	Creación de un rectángulo	104
7.1.3.2.	Creación de un círculo o elipse	104
7.1.3.3.	Paralelepípedo regular	105
7.1.3.4.	Creación de un cilindro	105
7.1.4.	Construcción de entidades geométricas mediante extrusión	106
7.1.4.1.	Creación de volúmenes a partir de la extrusión por traslación de superficies	106
7.1.4.2.	Creación de volúmenes a partir de la extrusión por rotación de superficies	106
7.1.5.	Asignación de entidades físicas para el proceso en PEFiCA	107
7.1.5.1.	Propiedades mecánicas de los materiales	107
7.1.5.2.	Definición de desplazamientos	108
7.1.5.3.	Aplicación de cargas puntuales	109
7.1.5.4.	Aplicación de presiones	110
7.1.6.	Preparación de entidades geométricas para el enmallado	111
7.1.6.1.	Preparación de las líneas	111
7.1.6.2.	Preparación de las superficies	112
7.1.6.3.	Preparación de los volúmenes	113
7.1.6.4.	Preparación de los volúmenes	113

7.1.7.	Generación de la malla	114
7.1.8.	Creación del archivo <code>.msh</code>	114
7.2.	Ejecución del programa en GNU-OCTAVE o MATLAB	115
7.3.	Lectura de resultados en el programa GMSH	116
7.3.1.	Desplazamientos, condición deformada y fuerzas en los nudos .	117
7.3.2.	Componentes de esfuerzo y deformación	118

Capítulo 1

Introducción

Este documento describe el trabajo final de maestría titulado "Desarrollo de un programa de elementos finitos para problemas elásticos lineales en tres dimensiones". El primer capítulo contiene la introducción general y los objetivos propuestos. En el segundo capítulo se presentan los antecedentes, en donde se explican brevemente algunos programas didácticos de elementos finitos similares. En el tercer capítulo se describe el marco teórico, indicando la formulación del método de los elementos finitos para el elemento hexaédrico trilinear y para el elemento tipo prisma.

En el cuarto capítulo se detalla la metodología con la que se realiza cada una de las etapas del presente trabajo. En el quinto capítulo se explica la implementación del programa. En el capítulo sexto se muestran ejemplos de aplicación del programa desarrollado, junto con la comparación de resultados con el programa comercial Ansys. En el capítulo séptimo se presentan las conclusiones y recomendaciones para futuros trabajos. Finalmente el capítulo octavo es el manual de usuario del programa desarrollado.

El programa desarrollado, los ejemplos de aplicación y el documento de este trabajo final de maestría se puede descargar en el siguiente vínculo [Trabajo final](#).

1.1. Motivación

El método de elementos finitos aplicado al análisis estructural ha adquirido cada vez más importancia, gracias al aumento de la capacidad de procesamiento de los computadores desde su aparición en la década de los setentas (Linero, 2010). Por lo anterior, los programas académicos de Ingeniería Civil y Mecánica de muchas universidades han incluido el método de los elementos finitos entre sus asignaturas.

Asimismo, algunas instituciones educativas y de investigación han desarrollado programas didácticos de elementos finitos a código abierto, diseñados con el objetivo

de mejorar el aprendizaje del método en los estudiantes y a su vez, servir como herramienta para futuras investigaciones. En la [Tabla 1.1](#) se muestran algunos programas de este tipo creadas por diferentes universidades e instituciones en varios países, de los cuales se indica el año de su última versión.

Tabla 1.1: Programas didácticos de elementos finitos

Programa	Institución	Año	Lenguaje
Z88	University of Bayreuth (Alemania)	1985	Fortran y C
FreeFem	Universidad Pierre y Marie Curie (Paris, Francia)	1987	C++
CodeAster	Electricité de France (EDF)	2001	Fortran y Python
MatFem	CIMNE - Universidad Politécnica de Cataluña (Barcelona, España)	2006	Matlab
Feap	University of California (Berkeley, EEUU)	2020	Fortran
Su2	Stanford University (EEUU)	2012	C++, Python
Calfem	Universidad de Lund (Suecia),	2004	Matlab
PEFiCA	Universidad Nacional de Colombia	2020	Octave o Matlab
2.0			

Existen varios programas comerciales de simulación numérica de problemas físicos mediante el método de los elementos finitos como ANSYS, Abaqus y NASTRAN, entre otros. Por otra parte, la mayoría de los programas comerciales de análisis de estructuras reticulares como SAP2000, ETABS o STAAD, han incluido procedimientos de simulación de placas y sólidos mediante el método de los elementos finitos.

En este trabajo se produce y documenta un programa didáctico a código abierto para solucionar problemas elásticos lineales en tres dimensiones, utilizando para ello programación estructurada y modular, la cual permite una mejor interpretación del código dentro de cada módulo (Molina, [2006](#)).

Este programa es una ayuda didáctica para el aprendizaje del método de los elementos finitos en tres dimensiones y también una herramienta para estudios que requieran apoyarse con este tipo de análisis.

1.2. Objetivo General

Crear y documentar un programa didáctico de elementos finitos a código abierto escrito en pseudo lenguaje matemático, capaz de simular el proceso de análisis de un

problema mecánico elástico lineal en tres dimensiones bajo cargas estáticas.

1.3. Objetivos Específicos

- Generar los algoritmos de conexión para el programa de preproceso, que permita vincular los datos de la geometría y enmallado, con el programa de proceso creado.
- Crear las rutinas de proceso en pseudo lenguaje matemático utilizando programación estructurada, que permitan calcular el desplazamiento, la deformación y el esfuerzo para cada punto dentro del sólido mediante el método de elementos finitos.
- Generar un archivo de salida, que contenga los resultados obtenidos en un formato que permita su visualización en un programa de postprocesamiento.
- Verificar los resultados de algunas simulaciones del programa desarrollado con un programa comercial de elementos finitos.
- Documentar de forma didáctica el programa presentando ejemplos de aplicación.

1.4. Metodología general

El programa de computador desarrollado en este trabajo final de maestría implementa su metodología de análisis en lenguaje interpretado GNU-OCTAVE, una plataforma de programación con base en un lenguaje interpretado de alto nivel, orientado al cálculo numérico (Eaton *et al.*, 2017). Las rutinas realizadas en GNU-OCTAVE serán compatibles con MATLAB.

La estrategia de representación gráfica de los parámetros geométricos y físicos del problema y de sus resultados, se realizó con el entorno de pre y post proceso del programa GMSH, el cual está disponible para varios sistemas operativos y su licencia es libre y gratuita (Geuzaine & Remacle, 2019).

Las actividades realizadas en este trabajo final de maestría se indican a continuación:

- Inicialmente se definen las rutinas de lectura en GNU-OCTAVE para que la información de la geometría del problema, cargas, condiciones de borde y enmallado que se generan desde GMSH pueda ser identificada por las rutinas del procedimiento principal.

- En las rutinas de proceso se analiza el problema en tres dimensiones mediante elementos finitos hexaédricos y su caso especial, el elemento pentaédrico o prisma, de acuerdo al enmallado realizado en GMSH. Con estas rutinas de proceso se escribe un archivo con los resultados del problema, es decir los desplazamientos, esfuerzos y deformaciones para los nudos y los elementos finitos.
- Para el post proceso se definen las rutinas de escritura en GNU-OCTAVE, de tal forma que permitan mostrar gráficamente sus resultados en GMSH.
- Se realizan simulaciones de cuatro problemas tridimensionales comunes con el programa propuesto y con el programa comercial ANSYS, con el fin de verificar sus resultados.
- Se escriben comentarios en las subrutinas de forma amigable y didáctica explicando su función y facilitando su uso. Por ejemplo, cada subrutina creada está encabezada con comentarios que identifiquen su objetivo y sus argumentos. Además el nombre de la subrutina está asociado a su función y tiene un número común de caracteres.
- Se escribe la documentación del programa, la cual incluye: la descripción general, la estructura y su manejo, los tipos de datos y las matrices utilizadas y la descripción de las instrucciones programadas. Asimismo se elabora una guía con algunos ejemplos de aplicación.

Capítulo 2

Antecedentes

Para la enseñanza de la ingeniería estructural por el método de los elementos finitos el uso de programas didácticos es cada vez más frecuente, debido a que éstos son difundidos con mayor facilidad (Duque & Mejía, 2007). Varios de los programas mostrados en la [Tabla 1.1](#) se encuentran además en constante evolución.

2.1. Estructura básica de los programas de elementos finitos

Los programas de elementos finitos tienen usualmente tres partes fundamentales denominadas el preproceso, el proceso y el postproceso (Duque & Mejía, 2007; Linero *et al.*, 2020). A continuación se describen tales etapas particularmente para la solución de problemas mecánicos.

2.1.1. Preproceso

La etapa de preproceso permite al usuario ingresar la información del problema de la mecánica estructural y usualmente está compuesto por: (i) un entorno de dibujo de las entidades geométricas como puntos, líneas, superficies y volúmenes, (ii) un procedimiento de asignación de condiciones de borde, acciones aplicadas y propiedades mecánicas, y (iii) los procedimientos de generación de la malla de elementos finitos.

2.1.2. Proceso

En la etapa de proceso se transforma toda la información ingresada en la etapa previa a matrices y tablas. A continuación se realizan las operaciones matriciales definidas por el método de los elementos finitos, obteniendo como resultados del problema arreglos que contienen: desplazamientos y fuerzas en los nudos, y deformaciones y esfuerzos en el interior de los elementos.

2.1.3. Postproceso

La etapa de postproceso convierte los arreglos obtenidos en la etapa de proceso, en distribución de colores o tonos sobre el sólido, curvas de relación y animaciones que permiten la representación gráfica de los resultados.

2.2. Programas didácticos y de aprendizaje

Los programas didácticos tienen características bien definidas para facilitar su eficacia en el aprendizaje (Marqués, 1995), como por ejemplo la versatilidad y la posibilidad de gestionar fácilmente la información que ellos proporcionan (Marqués, 2002). De esta manera la implementación de un programa didáctico de elementos finitos de forma estructurada, mediante la creación de funciones con un objetivo definido, favorece la caracterización de las mismas y se puede identificar mejor el proceso analítico de cada una.

Además, esta programación estructurada da versatilidad a los procesos vinculados al cálculo principal, con la posibilidad de modificar sus rutinas originales, de permitir la adición de nuevos procedimientos y eliminar otros, pensando en futuras mejoras y cambios en el alcance del programa.

A continuación se describen brevemente otros programas didácticos desarrollados para el análisis de estructuras por el método de elementos finitos.

2.2.1. MATFEM

MATFEM es un programa escrito en MATLAB, diseñado para introducir a los usuarios en el uso práctico del método de elementos finitos para sólidos y problemas de campo gobernados por las ecuaciones de Poisson y Laplace en espacios bi- y tridimensionales (mecánica, transferencia de calor, acústica, infiltración, electromagnetismo, etc.). Está vinculado con el programa gráfico GID (International Center for Numerical Methods in Engineering (CIMNE), 1987), para el preproceso y postproceso de la información.

MATFEM contiene algunos archivos en donde se guardan las rutinas completas para los análisis a partir de los archivos generados desde GID. Es un programa que aprovecha la capacidad de pre y post procesamiento de GID y numerosas herramientas predefinidas de MATLAB (Link, 2005).

2.2.2. CALFEM

CALFEM es un programa escrito en MATLAB para enseñar el método de elementos finitos. El nombre CALFEM es una abreviatura de *Computer Aided Learning of the Finite Element Method*. El programa se puede utilizar para diferentes tipos de problemas de mecánica estructural y problemas de campo.

CALFEM ha sido el resultado de varios desarrollos realizados por la división de mecánica estructural en la Universidad de Lund en Suecia. Contiene una serie de procedimientos dispuestos en varios archivos de MATLAB, con amplia documentación. Su alcance está limitado a problemas en dos dimensiones (Dahlblom *et al.*, 1986).

2.2.3. PEFiCA

En la Universidad Nacional de Colombia, sede Bogotá se ha utilizado el programa de elementos finitos a código abierto PEFiCA (Linero, 2010) en las asignaturas análisis matricial avanzado y análisis estructural por elementos finitos, con el cual los estudiantes aplican los conceptos adquiridos en las clases mediante ejemplos de aplicación con el programa incentivando el aprendizaje de los temas estudiados.

PEFiCA y su posterior actualización PEFiCA 2.0 (Linero *et al.*, 2020), han facilitado a los estudiantes la comprensión de la formulación y la práctica del método de los elementos finitos, brindando una solución a problemas mecánicos uni-, bi- y tri-dimensionales con cargas estáticas, deformaciones infinitesimales y de material elástico lineal isotrópico. El código de dicho programa coincide con los procedimientos básicos implementados en programas comerciales de análisis estructural y elementos finitos, favoreciendo un mejor desempeño en la práctica profesional.

En este trabajo final de maestría se extiende el programa PEFiCA 2.0, al análisis de sólidos en tres dimensiones con elementos finitos de orden superior, conservando las características relacionadas con su estructura básica y su objetivo didáctico.

Capítulo 3

Marco teórico

La formulación del método de los elementos finitos para la solución de un problema mecánico elástico lineal tridimensional ha sido descrita en varias referencias (Felippa, 2017a; Linero *et al.*, 2013; Zienkiewicz, 2013). En este capítulo se presenta la formulación particular del elemento hexaédrico de ocho nudos y de su simplificación al elemento prisma de 6 nudos, los cuales son implementados como parte del presente trabajo.

3.1. Generalidades del método de los elementos finitos

El método de los elementos finitos permite obtener una solución numérica aproximada a problemas del medio continuo, sobre los que se definen ciertas ecuaciones diferenciales o integrales que caracterizan el comportamiento físico del problema. Particularmente en este trabajo se analiza una solución a problemas mecánicos estructurales tridimensionales, por lo que las ecuaciones de gobierno vinculan la cinemática, el modelo constitutivo del material y las condiciones de equilibrio. Asimismo, las cantidades de interés son el campo del desplazamiento, de la deformación y del esfuerzo en el sólido (Linero *et al.*, 2013).

En el método de elementos finitos, el medio continuo se divide en un número finito de subdominios llamados elementos finitos, los cuales responden a las ecuaciones de la mecánica y se encuentran vinculados entre sí por nudos (Zienkiewicz, 2013). La forma de los elementos finitos depende del dominio en el cual se define el problema. En este caso como el dominio es tridimensional los elementos finitos son también tridimensionales.

3.1.1. Funciones de aproximación y funciones de forma

La variable que se desea obtener a directamente de las ecuaciones de gobierno en términos de la posición material en el dominio del problema, se denomina función de aproximación. Tal función es continua en el interior de cada elemento finito, y es compatible en los nudos de la malla.

Particularmente en problemas mecánicos de sólidos, la función de aproximación corresponde al campo vectorial del desplazamiento $\mathbf{u}(x, y, z)$ y en consecuencia, los valores de sus componentes en los nudos o también llamados valores nodales son los grados de libertad del problema.

En el interior de cada elemento finito e , se define una función de aproximación escalar por cada componente del vector del desplazamiento en las direcciones x , y y z , de la forma:

$$u^{(e)}(x, y, z) = N_1^{(e)}(x, y, z)u_1^{(e)} + \cdots + N_n^{(e)}(x, y, z)u_n^{(e)} \quad (3.1a)$$

$$v^{(e)}(x, y, z) = N_1^{(e)}(x, y, z)v_1^{(e)} + \cdots + N_n^{(e)}(x, y, z)v_n^{(e)} \quad (3.1b)$$

$$w^{(e)}(x, y, z) = N_1^{(e)}(x, y, z)w_1^{(e)} + \cdots + N_n^{(e)}(x, y, z)w_n^{(e)} \quad (3.1c)$$

En las expresiones anteriores, $u_i^{(e)}$, $v_i^{(e)}$ y $w_i^{(e)}$, son los valores de las componentes de desplazamiento en el nudo i del elemento e .

Las funciones de forma $N_i^{(e)}$ en el nudo i de un elemento finito e , son polinomios en términos de la posición (x, y, z) , que determinan el orden de la aproximación y que permiten interpolar una componente de desplazamiento a partir de sus valores en los nudos.

3.1.2. Principio del Trabajo Virtual

Se considera un sólido cuyo volumen es V , el cual está sometido a acciones externas y tiene restricciones debidas a sus apoyos. Este sólido está representado mediante una malla de elementos finitos tridimensionales. En los vértices o nudos de cada elemento finito se definen las componentes de desplazamiento, las cuales corresponden a los grados de libertad.

Las fuerzas reales producen un trabajo virtual el cual se divide en dos partes: el trabajo virtual interno el cual lo realizan las fuerzas internas y el trabajo virtual externo, realizado por las fuerzas externas.

Las fuerzas a las que el sólido compuesto por ne elementos finitos de volumen V_e está sometido son:

- fuerzas por unidad de volumen en cada elemento \mathbf{b}_e ,

- fuerzas por unidad de área \mathbf{p}_e en el contorno $\Gamma_p^e \subset \Gamma_p$ y
- fuerzas puntuales $\mathbf{f}(\mathbf{x}_n)$ aplicadas sobre r nudos.

Las condiciones de frontera son los valores conocidos del desplazamiento real $\mathbf{u}(\mathbf{x}) = [u \ v \ w]^T = \mathbf{u}^*$ en los puntos que hacen parte de un contorno definido Γ_u (Linero *et al.*, 2020).

El campo del desplazamiento virtual $\delta\mathbf{u}(\mathbf{x}) = [\delta u \ \delta v \ \delta w]^T$, corresponde a una función que cumple con la condición de borde de la forma:

$$\delta\mathbf{u}(\mathbf{x}) = \mathbf{0} \quad \forall \mathbf{x} \in \Gamma_u \quad (3.2)$$

El trabajo virtual externo en cada elemento e se realiza por las fuerzas reales externas $\mathbf{b}(\mathbf{e})$, $\mathbf{p}(\mathbf{e})$ y $\mathbf{f}(\mathbf{x}_n)$ mientras se presenta un desplazamiento virtual $\delta\mathbf{u}(\mathbf{e})$.

Por otro lado, el trabajo virtual interno es producido por el esfuerzo σ_e mientras ocurren deformaciones generadas por desplazamientos virtuales $\tilde{\nabla}\delta\mathbf{u}_e$, por lo tanto es igual a la densidad de energía de deformación $(\tilde{\nabla}\delta\mathbf{u}_e)^T \sigma_e$ integrada en el volumen del elemento finito.

De esta manera, se dice que un cuerpo está en equilibrio si el trabajo virtual realizado por las acciones externas e internas es nulo, por lo tanto:

$$\sum_{e=1}^{ne} \int_{V^{(e)}} (\tilde{\nabla}\delta u_e)^T \sigma_e dV = \sum_{e=1}^{ne} \left(\int_{\Gamma_p^e} \delta\mathbf{u}_e^T \mathbf{p}_e dS + \int_{V^e} \delta\mathbf{u}_e^T \mathbf{b}_e dV \right) + \sum_{n=1}^r \delta\mathbf{u}(\mathbf{x}_n)^T \mathbf{f}(\mathbf{x}_n) \quad (3.3)$$

Los vectores de fuerzas de superficie \mathbf{p}_e , fuerzas de cuerpo \mathbf{b}_e en los elementos y fuerzas puntuales en los nudos $\mathbf{f}(\mathbf{x}_n)$, son de la forma:

$$\mathbf{p}_e = [p_x^e \ p_y^e \ p_z^e]^T \quad \mathbf{b}_e = [b_x^e \ b_y^e \ b_z^e]^T \quad \mathbf{f}(\mathbf{x}_n) = [f_x \ f_y \ f_z]^T \quad (3.4)$$

3.2. Formulación del método de los elementos finitos para problemas de la mecánica estructural lineales tridimensionales

A continuación se describe brevemente la formulación del método de los elementos finitos aplicada a problemas mecánicos de sólidos elásticos lineales sometidos a cargas estáticas en el espacio tridimensional (Linero *et al.*, 2020). Tal formulación es común para cualquier tipo de elemento finito en tres dimensiones como los indicados en las secciones anteriores.

3.2.1. Matriz de rigidez y vector de fuerzas equivalentes en el sólido

La matriz de rigidez del sólido \mathbf{K} se obtiene al ensamblar las matrices de rigidez $\mathbf{K}^{(e)}$ de los ne elementos que conforman la malla. Esto equivale a la adición de acuerdo a los grados de libertad asociados al elemento (e), de cada término de la matriz de rigidez del elemento en la matriz de rigidez del sólido. Este procedimiento de ensamblaje se representa como:

$$\mathbf{K} = \bigoplus_{e=1}^{ne} [\mathbf{K}^{(e)}] \quad (3.5)$$

El vector de fuerzas nodales totales en la malla \mathbf{f} , resulta de la suma del vector de fuerzas aplicadas directamente a los nudos $\mathbf{f}^{(n)}$ con el vector $\tilde{\mathbf{f}}$, es decir:

$$\mathbf{f} = \mathbf{f}^{(n)} + \tilde{\mathbf{f}} \quad (3.6)$$

El vector $\tilde{\mathbf{f}}$ resulta del proceso de ensamblaje de los vectores $\tilde{\mathbf{f}}^{(e)}$ de cada elemento vistos en la [Ecuación 3.57](#), de acuerdo a los grados de libertad asociados a cada elemento, de forma similar a la matriz de rigidez del sólido. Este proceso de ensamblaje de los vectores se representa así:

$$\tilde{\mathbf{f}} = \bigoplus_{e=1}^{ne} [\tilde{\mathbf{f}}^{(e)}] \quad (3.7)$$

Por consiguiente, a partir de la [Ecuación 3.51](#), con la matriz de rigidez del sólido \mathbf{K} , el vector de fuerzas nodales equivalentes en la malla \mathbf{f} , el vector de desplazamientos nodales virtuales $\delta\mathbf{a}$ y el vector de desplazamientos nodales reales \mathbf{a} se deduce que:

$$\delta\mathbf{a}^T \mathbf{K}\mathbf{a} - \delta\mathbf{a}^T \mathbf{f} = 0 \quad (3.8a)$$

$$\delta\mathbf{a}^T (\mathbf{K}\mathbf{a} - \mathbf{f}) = 0 \quad (3.8b)$$

$$\mathbf{K}\mathbf{a} = \mathbf{f} \quad (3.8c)$$

3.2.2. Metodología para el cálculo del vector de desplazamientos nodales en el sólido

El vector de desplazamientos nodales del sólido \mathbf{a} está compuesto por los desplazamientos desconocidos \mathbf{a}_α y los conocidos \mathbf{a}_β , así:

$$\mathbf{a} = [\mathbf{a}_\alpha^T \quad \mathbf{a}_\beta^T]^T \quad (3.9)$$

estos desplazamientos conocidos son los valores nodales de desplazamiento impuestos en la malla \mathbf{a}^* . De esta manera la [Ecuación 3.8](#) se puede representar como:

$$\begin{bmatrix} \mathbf{K}_{\alpha\alpha} & \vdots & \mathbf{K}_{\alpha\beta} \\ \cdots & \cdots & \cdots \\ \mathbf{K}_{\beta\alpha} & \vdots & \mathbf{K}_{\beta\beta} \end{bmatrix} \begin{bmatrix} \mathbf{a}_\alpha \\ \cdots \\ \mathbf{a}_\beta \end{bmatrix} - \begin{bmatrix} \mathbf{f}_\alpha \\ \cdots \\ \mathbf{f}_\beta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \cdots \\ \mathbf{0} \end{bmatrix} \quad (3.10)$$

De acuerdo con el número de desplazamientos nodales conocidos \mathbf{a}_α y desconocidos \mathbf{a}_β , la matriz de rigidez \mathbf{K} se particiona en cuatro submatrices $\mathbf{K}_{\alpha\alpha}$, $\mathbf{K}_{\alpha\beta}$, $\mathbf{K}_{\beta\alpha}$ y $\mathbf{K}_{\beta\beta}$. Asimismo el vector de fuerzas nodales \mathbf{f} se particiona en los vectores \mathbf{f}_α y \mathbf{f}_β .

Con esta partición según los grados de libertad conocidos o desconocidos se generan las siguientes ecuaciones matriciales:

$$\mathbf{K}_{\alpha\alpha}\mathbf{a}_\alpha + \mathbf{K}_{\alpha\beta}\mathbf{a}_\beta - \mathbf{f}_\alpha = \mathbf{0} \quad (3.11a)$$

$$\mathbf{K}_{\beta\alpha}\mathbf{a}_\alpha + \mathbf{K}_{\beta\beta}\mathbf{a}_\beta - \mathbf{f}_\beta = \mathbf{0} \quad (3.11b)$$

De la [Ecuación 3.11a](#) se calcula el vector de desplazamientos nodales desconocidos \mathbf{a}_α :

$$\mathbf{a}_\alpha = \mathbf{K}_{\alpha\alpha}^{-1}(\mathbf{f}_\alpha - \mathbf{K}_{\alpha\beta}\mathbf{a}_\beta) \quad (3.12)$$

3.2.3. Vector de fuerzas totales y reacciones en el sólido

A partir de la [Ecuación 3.11b](#) se despeja el vector \mathbf{f}_β , así:

$$\mathbf{f}_\beta = \mathbf{K}_{\beta\alpha}\mathbf{a}_\alpha + \mathbf{K}_{\beta\beta}\mathbf{a}_\beta \quad (3.13)$$

El vector \mathbf{f}_β de acuerdo con la ecuación [3.6](#) equivale a:

$$\mathbf{f}_\beta = \tilde{\mathbf{f}}_\beta + \mathbf{f}_\beta^{(n)} \quad (3.14)$$

Los vectores $\tilde{\mathbf{f}}_\beta$ y $\mathbf{f}_\beta^{(n)}$ son desconocidos y están asociados con los desplazamientos nodales conocidos. El vector $\mathbf{f}_\beta^{(n)}$ corresponde a las reacciones en los apoyos del sólido y se obtiene de la siguiente forma:

$$\mathbf{f}_\beta^{(n)} = \mathbf{f}_\beta - \tilde{\mathbf{f}}_\beta \quad (3.15a)$$

$$\mathbf{f}_\beta^{(n)} = \mathbf{K}_{\beta\alpha}\mathbf{a}_\alpha + \mathbf{K}_{\beta\beta}\mathbf{a}_\beta - \mathbf{f}_\beta \quad (3.15b)$$

3.2.4. Extracción de los componentes de desplazamiento, deformación y esfuerzo en el interior de cada elemento

El vector $\mathbf{a} = [\mathbf{a}_\alpha^T \ \mathbf{a}_\beta^T]^T$ corresponde a las componentes de desplazamiento en todos los nudos de la malla de acuerdo a la numeración de los grados de libertad. Cada elemento finito tiene asociado un vector de desplazamientos nodales $\mathbf{a}^{(e)}$, el cual corresponde a las componentes de desplazamiento para los nudos del elemento. Este vector $\mathbf{a}^{(e)}$ se obtiene a partir de la extracción de los coeficientes correspondientes a los grados de libertad del elemento e . Tal procedimiento se representa como:

$$\mathbf{a}^{(e)} = \underset{(e)}{\mathbb{I}\mathbb{E}} [\mathbf{a}] \quad (3.16)$$

Con el vector de desplazamientos en los nudos del elemento $\mathbf{a}^{(e)}$ se calculan los desplazamientos en un punto en el interior del elemento utilizando la [Ecuación 3.27](#). Asimismo, se obtiene la matriz de componentes de deformación del elemento $\boldsymbol{\varepsilon}^{(e)}$, a través de la matriz $\mathbf{B}^{(e)}(\xi, \eta, \mu)$ mediante la [Ecuación 3.46](#), cuyo resultado es:

$$\boldsymbol{\varepsilon}^{(e)}(\xi, \eta, \mu) = [\varepsilon_{xx} \ \varepsilon_{yy} \ \varepsilon_{zz} \ \gamma_{xy} \ \gamma_{xz} \ \gamma_{yz}]^T \quad (3.17)$$

Finalmente, utilizando la matriz constitutiva elástica $\mathbf{D}^{(e)}$ definida en la [Subsección 3.6.7](#) y $\boldsymbol{\varepsilon}^{(e)}$, se calculan las componentes de la matriz de esfuerzo $\boldsymbol{\sigma}^{(e)}$ utilizando la siguiente ecuación:

$$\boldsymbol{\sigma}^{(e)}(\xi, \eta, \mu) = \mathbf{D}^{(e)} \boldsymbol{\varepsilon}^{(e)} \quad (3.18)$$

Las componentes de la matriz columna $\boldsymbol{\sigma}^{(e)}$ con respecto al sistema coordenado global son:

$$\boldsymbol{\sigma}^{(e)}(\xi, \eta, \mu) = [\sigma_{xx} \ \sigma_{yy} \ \sigma_{zz} \ \sigma_{xy} \ \sigma_{xz} \ \sigma_{yz}]^T \quad (3.19)$$

3.3. Valores característicos de esfuerzo y de deformación

El estado de esfuerzos dado por sus componentes con respecto al sistema coordenado global de la [Ecuación 3.19](#), permite obtener los esfuerzos principales σ_1 , σ_2 y σ_3 , los cuales incluyen a los valores máximos y mínimos de esfuerzo normal en un plano. Los esfuerzos principales corresponden a los tres valores propios σ_p obtenidos como solución de la siguiente ecuación matricial (Linero & Garzón, 2010)

$$\left(\begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \sigma_p \right) \begin{bmatrix} n_x^{(p)} \\ n_y^{(p)} \\ n_z^{(p)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.20)$$

El vector de coeficientes $\mathbf{n}^{(p)} = [n_x^{(p)}, n_y^{(p)}, n_z^{(p)}]^T$ corresponde a los vectores en las direcciones principales obtenidos como los vectores propios de la ecuación anterior.

Asimismo, el estado de deformaciones está definido por seis componentes diferentes, con respecto al sistema coordenado global xyz , como se muestra en la [Ecuación 3.17](#). A partir de este estado de deformaciones se determinan las deformaciones principales ε_p , de las cuales dos son aquellas deformaciones longitudinales de valor máximo y mínimo, que pueden actuar sobre una superficie (Linero & Garzón, 2010).

Estas deformaciones principales ε_p y sus respectivas direcciones $\mathbf{m}^p = [m_x^{(p)}, m_y^{(p)}, m_z^{(p)}]^T$ se obtienen al resolver el siguiente problema de valores y vectores propios:

$$\left(\begin{bmatrix} \varepsilon_{xx} & \frac{1}{2}\gamma_{xy} & \frac{1}{2}\gamma_{xz} \\ \frac{1}{2}\gamma_{xy} & \varepsilon_{yy} & \frac{1}{2}\gamma_{yz} \\ \frac{1}{2}\gamma_{xz} & \frac{1}{2}\gamma_{yz} & \varepsilon_{zz} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \varepsilon_p \right) \begin{bmatrix} m_x^{(p)} \\ m_y^{(p)} \\ m_z^{(p)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.21)$$

3.4. Criterio de von Mises

El criterio de fallo de von Mises establece que el comportamiento elástico de un punto del sólido se mantiene si la densidad de energía de deformación por distorsión actuante es menor que la densidad de energía admisible del material (Linero *et al.*, 2020).

Si el material es dúctil y su fluencia coincide con el límite de su rango elástico, se puede afirmar que el material tiene un comportamiento elástico siempre que $\sigma_{vm} \leq \sigma_y$, donde σ_y el esfuerzo de fluencia del material y σ_{vm} es el esfuerzo de von Mises actuante que se obtiene de la forma:

$$\sigma_{vm} = \frac{1}{2} \sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_1 - \sigma_3)^2 + (\sigma_2 - \sigma_3)^2} \quad (3.22)$$

3.5. Valores promedio de esfuerzo y deformación en los nudos de la malla

Con el objetivo de mejorar la visualización de los resultados de esfuerzos y deformaciones en la malla de elementos finitos, se busca evitar grandes variaciones entre los valores de los esfuerzos y deformaciones de los elementos consecutivos que tienen nudos en común, dentro de la malla de elementos.

A partir de los esfuerzos y las deformaciones calculados en el interior del elemento, utilizando las funciones de forma se obtienen los valores de esfuerzos y deformaciones evaluados en los nudos de cada elemento.

Cada nudo de la malla que es compartido por más de un elemento finito, va a tener unos valores de esfuerzos y deformaciones provenientes de cada uno de los elementos que lo contienen. Estos valores de esfuerzos y deformaciones en el nudo i , que provienen de los elementos adyacentes son promediados, para luego designar el resultado de este promedio como valor de esfuerzo o deformación en el nudo.

Posteriormente se calculan los esfuerzos o deformaciones al interior del elemento mediante las funciones de forma utilizando para ello los valores nodales promedio previamente calculados. Este es un proceso que se conoce como suavizado y permite representar los valores de esfuerzo o deformación entre elementos adyacentes de forma continua en toda la malla de elementos finitos.

En el siguiente ejemplo se considera que un nudo i es común a los elementos finitos 1, 2 y 3, y que la deformación normal $\varepsilon_{xx(i)}^{(1)}$, $\varepsilon_{xx(i)}^{(2)}$ y $\varepsilon_{xx(i)}^{(3)}$, es evaluada para la posición del nudo i en el interior de los elementos 1, 2 y 3, respectivamente. Primero se calcula la deformación normal promedio en el nudo i como:

$$\bar{\varepsilon}_{xx(i)} = \frac{1}{3}(\varepsilon_{xx(i)}^{(1)} + \varepsilon_{xx(i)}^{(2)} + \varepsilon_{xx(i)}^{(3)})$$

Este procedimiento se repite en todos los nudos de la malla de elementos finitos. Después se evalúa la deformación promedio $\bar{\varepsilon}_{xx}$ en el interior del elemento por la interpolación dada por sus funciones de forma $N_i^{(e)}$ y los valores de deformación promedio en los nudos $\bar{\varepsilon}_{xx(i)}$, así:

$$\bar{\varepsilon}_{xx} = \sum_{i=1}^{ni} N_i^{(e)} \bar{\varepsilon}_{xx(i)} \quad (3.23)$$

siendo ni el número de nudos del elemento finito.

3.6. Formulación del método de elementos finitos para el elemento hexaédrico de 8 nudos

Los conceptos presentados en esta sección están basados en el método de los elementos finitos aplicado a la mecánica elástica lineal utilizando el elemento hexaédrico irregular de 8 nudos (Felippa, 2017a; Linero *et al.*, 2020).

Así como los elementos tetraédricos lineales son una expansión a tres dimensiones de los triangulares lineales, los elementos hexaédricos de 8 nudos lo son a los cuadriláteros de 4 nudos. El elemento hexaédrico irregular de 8 nudos o simplemente elemento hexaédrico es un poliedro de 6 caras, 12 lados y 8 nudos ubicados en sus

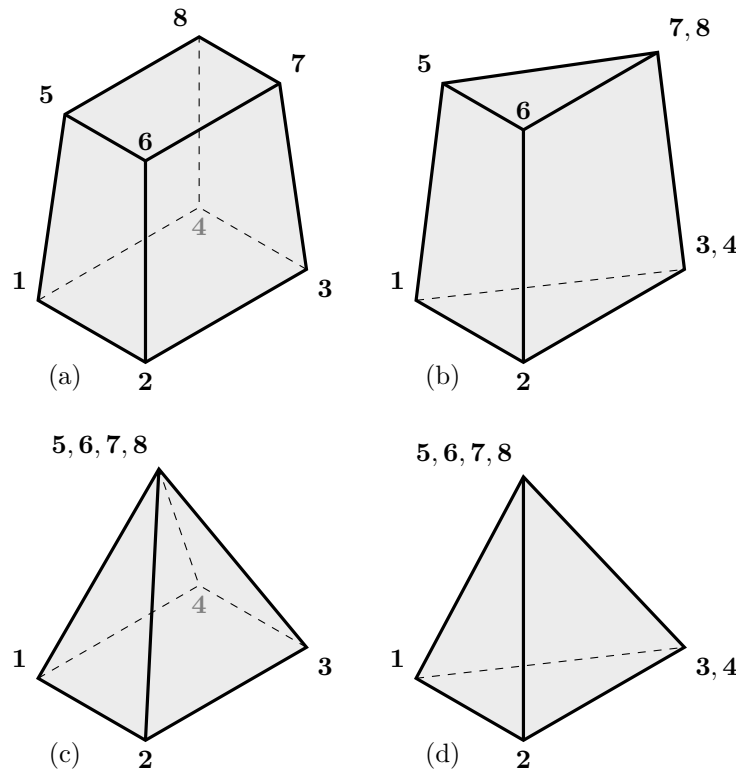


Figura 3.1: Esquema y numeración de nudos en el elemento hexaédrico y en sus formas simplificadas más comunes: (a) hexaedro irregular, (b) prisma, (c) pirámide y (d) tetraedro.

vértices como lo muestra la [Figura 3.1\(a\)](#). Este elemento finito es el más simple de la familia de los elementos isoparamétricos tridimensionales con forma de hexaedro.

Este tipo de elemento finito permite describir al menos diez tipos de volúmenes simplificados que resultan de modificar su geometría estándar (Dudley & Owen, 2014). Dentro de las formas más comunes del hexaedro simplificado se encuentra el prisma, el cual se obtiene superponiendo los nudos 3 y 4, y 7 y 8, como lo ilustra la [Figura 3.1\(b\)](#). También se puede lograr la forma de pirámide, superponiendo los nudos 5, 6, 7 y 8, como se indica en la [Figura 3.1\(c\)](#). Asimismo, se puede producir la forma de tetraedro superponiendo los nudos 3 y 4, y 5, 6, 7 y 8 como se indica en la [Figura 3.1\(d\)](#).

El elemento hexaédrico es usado ampliamente en simulación numérica de problemas con sólidos tridimensionales, siendo escogido en vez del tetraedro lineal en muchas situaciones (Dudley & Owen, 2014).

El número de nudos del elemento hexaédrico está determinado por el orden de

la función de aproximación en su interior. El hexaedro de 8 nudos tiene una función de aproximación de orden trilineal. En cambio, los elementos hexaédricos irregulares de 20 y 27 nudos, contienen funciones de aproximación de orden superior. Particularmente, este trabajo abarca el elemento finito de 8 nudos (Felippa, 2017a).

3.6.1. Numeración de los nudos

Los ocho vértices del hexaedro son numerados del 1 al 8, conservando una secuencia que garantice un valor positivo del volumen, utilizando el determinante del Jacobiano, como se muestra mas adelante. Esto se logra mediante las siguientes reglas de numeración:

- Tomar un vértice inicial y designarla como 1. Luego, seleccionar una cara a la cual pertenezca este vértice, es decir, cualquiera de las tres caras adyacentes a ella.
- Designar los otros tres vértices de la cara seleccionada en el ítem anterior con los números 2, 3 y 4, recorriendo los vértices en sentido anti-horario a un vector normal a la cara seleccionada entrando al elemento.
- Nombrar el nudo 5 en el vértice de la cara opuesta que se conecta con una línea al nudo 1.
- Escoger los demás vértices de la cara opuesta con los números 6, 7 y 8, recorriéndolos en sentido horario a un vector unitario a la cara opuesta entrando al elemento.

En la [Figura 3.1\(a\)](#) se muestra el resultado de la aplicación de las anteriores reglas de numeración.

3.6.2. Coordenadas naturales

Las coordenadas naturales para el elemento hexaédrico se definen con las letras griegas ξ , η y μ . Estas coordenadas van de una cara a la otra en cada uno de los tres sentidos del elemento. Los valores que toman las coordenadas naturales son desde -1 en una cara, hasta 1 en la cara opuesta, a lo largo del elemento. Para el elemento del ejemplo anterior las coordenadas naturales se muestran en la [Figura 3.2](#) y quedarían así:

- ξ toma el valor de -1 en el centro de la cara 1485 y $+1$ en el centro de la cara 2376.

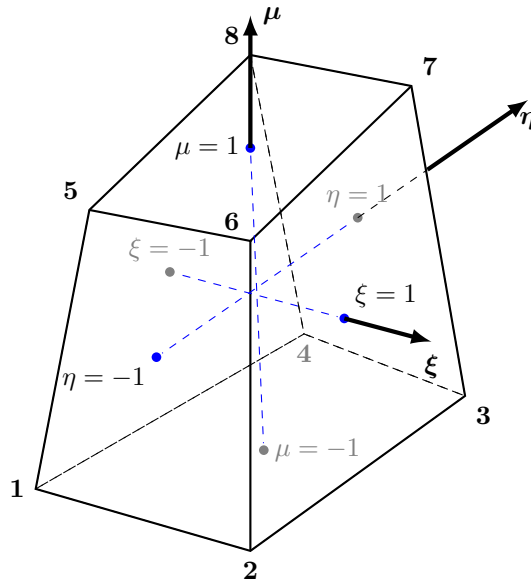


Figura 3.2: Coordenadas naturales en el hexaedro

- η toma el valor de -1 en el centro de la cara 1265 y $+1$ en el centro de la cara 3487.
- μ toma el valor de -1 en el centro de la cara 1234 y $+1$ en el centro de la cara 5678.

Las coordenadas naturales para los vértices, lados y caras del elemento finito se muestran en la [Tabla 3.1](#) (Felippa, 2017a).

3.6.3. Funciones de forma

Particularmente el elemento hexaédrico de ocho nudos está conformado por un nudo en cada uno de sus ocho vértices, de coordenadas (x_i, y_i, z_i) y tres grados de libertad por cada nudo, asociados a las componentes de desplazamiento $u_i^{(e)}$, $v_i^{(e)}$ y $w_i^{(e)}$. De esta manera el elemento presentado tiene en total 24 grados de libertad, como se muestra en la [Figura 3.3](#).

La matriz de coordenadas globales de los nudos del hexaedro $\mathbf{c}^{(e)}$, donde (x_i, y_i, z_i) es la coordenada de cada uno de los nudos del elemento, corresponde a:

nudo	ξ	η	μ	lado	coordenada	cara	coordenada
1	-1	-1	-1	1-2	$\eta = -1, \mu = -1$	1-4-8-5	$\xi = -1$
2	+1	-1	-1	2-3	$\xi = +1, \mu = -1$	1-2-6-5	$\eta = -1$
3	+1	+1	-1	3-4	$\eta = +1, \mu = -1$	1-2-3-4	$\mu = -1$
4	-1	+1	-1	1-4	$\xi = -1, \mu = -1$	2-3-7-6	$\xi = +1$
5	-1	-1	+1	1-5	$\xi = -1, \eta = -1$	3-7-8-4	$\eta = +1$
6	+1	-1	+1	2-6	$\xi = +1, \eta = -1$	5-6-7-8	$\mu = +1$
7	+1	+1	+1	3-7	$\xi = +1, \eta = +1$		
8	-1	+1	+1	4-8	$\xi = -1, \eta = +1$		
				5-6	$\eta = +1, \mu = +1$		
				6-7	$\xi = +1, \mu = +1$		
				7-8	$\eta = +1, \mu = +1$		
				5-8	$\xi = -1, \mu = +1$		

Tabla 3.1: Coordenadas naturales para el hexaedro: a la izquierda coordenadas de los nudos, al centro coordenadas de los lados y a la derecha coordenadas de las caras.

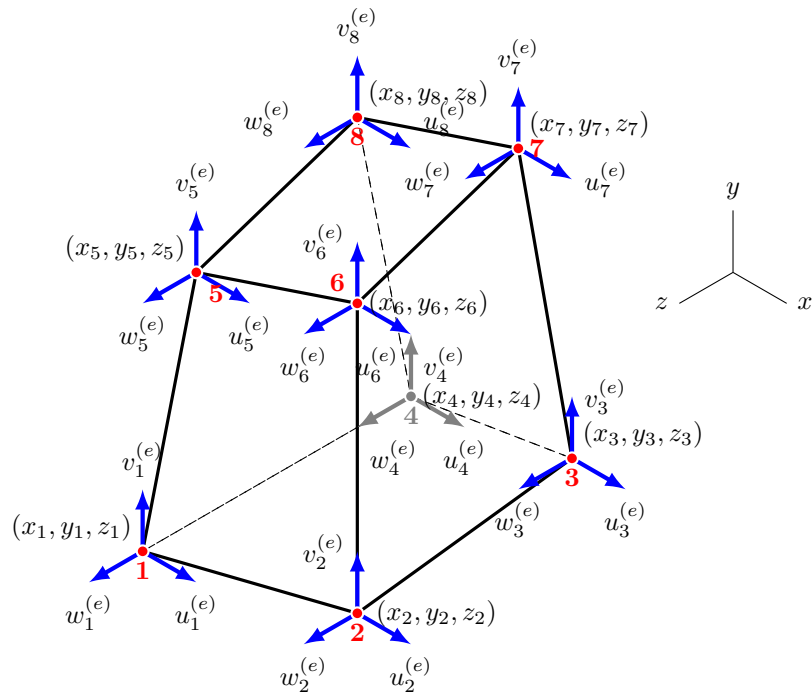


Figura 3.3: Grados de libertad del elemento hexaédrico

$$\mathbf{c}^{(e)} = \begin{bmatrix} x_1^{(e)} & y_1^{(e)} & z_1^{(e)} \\ x_2^{(e)} & y_2^{(e)} & z_2^{(e)} \\ x_3^{(e)} & y_3^{(e)} & z_3^{(e)} \\ x_4^{(e)} & y_4^{(e)} & z_4^{(e)} \\ x_5^{(e)} & y_5^{(e)} & z_5^{(e)} \\ x_6^{(e)} & y_6^{(e)} & z_6^{(e)} \\ x_7^{(e)} & y_7^{(e)} & z_7^{(e)} \\ x_8^{(e)} & y_8^{(e)} & z_8^{(e)} \end{bmatrix} \quad (3.24)$$

Se considera el vector de desplazamientos nodales del elemento $\mathbf{a}^{(e)}$ a la matriz columna que contiene los desplazamientos en los nudos, como se muestra:

$$\mathbf{a}^{(e)} = \left[\mathbf{a}_1^{(e)T} \quad \vdots \quad \mathbf{a}_2^{(e)T} \quad \vdots \quad \dots \quad \vdots \quad \mathbf{a}_8^{(e)T} \right]^T \quad (3.25a)$$

$$\mathbf{a}_i^{(e)} = \left[u_i^{(e)} \quad v_i^{(e)} \quad w_i^{(e)} \right]^T \quad (3.25b)$$

las componentes de desplazamiento $u^{(e)}$, $v^{(e)}$ y $w^{(e)}$, en una coordenada (x, y, z) del interior del hexaedro, se obtienen a partir de la interpolación de los valores nodales $u_i^{(e)}$, $v_i^{(e)}$ y $w_i^{(e)}$, de la forma:

$$u^{(e)}(x, y, z) = \mathbf{N}_1^{(e)} u_1^{(e)} + \mathbf{N}_2^{(e)} u_2^{(e)} + \dots + \mathbf{N}_8^{(e)} u_8^{(e)} \quad (3.26a)$$

$$v^{(e)}(x, y, z) = \mathbf{N}_1^{(e)} v_1^{(e)} + \mathbf{N}_2^{(e)} v_2^{(e)} + \dots + \mathbf{N}_8^{(e)} v_8^{(e)} \quad (3.26b)$$

$$w^{(e)}(x, y, z) = \mathbf{N}_1^{(e)} w_1^{(e)} + \mathbf{N}_2^{(e)} w_2^{(e)} + \dots + \mathbf{N}_8^{(e)} w_8^{(e)} \quad (3.26c)$$

donde $N_i^{(e)}(x, y, z)$ es la función de forma del nudo i del elemento e , evaluada en la posición (x, y, z) .

La forma matricial del vector del desplazamiento en un punto de coordenadas (x, y, z) en el interior del hexaedro puede obtenerse mediante:

$$\mathbf{u}^{(e)}(x, y, z) = \mathbf{N}^{(e)}(x, y, z) \mathbf{a}^{(e)} \quad (3.27)$$

La matriz de funciones de forma del elemento hexaédrico puede expresarse en términos de submatrices 3×3 , así:

$$\mathbf{N}^{(e)} = \left[\mathbf{N}_1^{(e)} \quad \mathbf{N}_2^{(e)} \quad \dots \quad \mathbf{N}_8^{(e)} \right] \quad (3.28a)$$

siendo,

$$\mathbf{N}_i^{(e)} = \begin{bmatrix} N_i & 0 & 0 \\ 0 & N_i & 0 \\ 0 & 0 & N_i \end{bmatrix} \quad (3.28b)$$

En consecuencia la matriz de funciones de forma tiene 3 filas por 24 columnas y corresponde a:

$$\mathbf{N}^{(e)} = \begin{bmatrix} N_1^{(e)} & 0 & 0 & N_2^{(e)} & 0 & 0 & \dots & N_8^{(e)} & 0 & 0 \\ 0 & N_1^{(e)} & 0 & 0 & N_2^{(e)} & 0 & \dots & 0 & N_8^{(e)} & 0 \\ 0 & 0 & N_1^{(e)} & 0 & 0 & N_2^{(e)} & \dots & 0 & 0 & N_8^{(e)} \end{bmatrix}_{3 \times 24} \quad (3.28c)$$

Las ocho funciones de forma asociadas a cada nudo del elemento $N_i^{(e)}$ son las siguientes:

$$N_1^{(e)} = \frac{1}{8}(1 - \xi)(1 - \eta)(1 - \mu) \quad (3.29a)$$

$$N_2^{(e)} = \frac{1}{8}(1 + \xi)(1 - \eta)(1 - \mu) \quad (3.29b)$$

$$N_3^{(e)} = \frac{1}{8}(1 + \xi)(1 + \eta)(1 - \mu) \quad (3.29c)$$

$$N_4^{(e)} = \frac{1}{8}(1 - \xi)(1 + \eta)(1 - \mu) \quad (3.29d)$$

$$N_5^{(e)} = \frac{1}{8}(1 - \xi)(1 - \eta)(1 + \mu) \quad (3.29e)$$

$$N_6^{(e)} = \frac{1}{8}(1 + \xi)(1 - \eta)(1 + \mu) \quad (3.29f)$$

$$N_7^{(e)} = \frac{1}{8}(1 + \xi)(1 + \eta)(1 + \mu) \quad (3.29g)$$

$$N_8^{(e)} = \frac{1}{8}(1 - \xi)(1 + \eta)(1 + \mu) \quad (3.29h)$$

Estas funciones de forma se pueden resumir en la siguiente expresión:

$$N_i^{(e)} = \frac{1}{8}(1 - \xi\xi_i)(1 + \eta\eta_i)(1 + \mu\mu_i) \quad (3.30)$$

donde ξ_i , η_i y μ_i , son las coordenadas naturales del i -ésimo nudo.

3.6.4. Matriz de operadores diferenciales con respecto al sistema natural actuando sobre funciones de forma

Las derivadas parciales de las funciones de forma del elemento hexaédrico con respecto a las coordenadas cartesianas, se obtienen utilizando la regla de la cadena. Dicha operación diferencial en el espacio tridimensional corresponde a:

$$\frac{\partial N_i^e}{\partial x}(\xi, \eta, \mu) = \frac{\partial N_i^e}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial N_i^e}{\partial \eta} \frac{\partial \eta}{\partial x} + \frac{\partial N_i^e}{\partial \mu} \frac{\partial \mu}{\partial x} \quad (3.31a)$$

$$\frac{\partial N_i^e}{\partial y}(\xi, \eta, \mu) = \frac{\partial N_i^e}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial N_i^e}{\partial \eta} \frac{\partial \eta}{\partial y} + \frac{\partial N_i^e}{\partial \mu} \frac{\partial \mu}{\partial y} \quad (3.31b)$$

$$\frac{\partial N_i^e}{\partial z}(\xi, \eta, \mu) = \frac{\partial N_i^e}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial N_i^e}{\partial \eta} \frac{\partial \eta}{\partial z} + \frac{\partial N_i^e}{\partial \mu} \frac{\partial \mu}{\partial z} \quad (3.31c)$$

Reescribiendo las ecuaciones anteriores en forma matricial y definiendo a $\mathbf{H}_{(i)}^{(e)}$ como la matriz columna que contiene las derivadas de las funciones de forma en sistema global para cada nudo i , se tiene que:

$$\mathbf{H}_{(i)}^{(e)} = \mathbf{J}^{-1} \hat{\mathbf{B}}_{(i)} \quad (3.32)$$

siendo:

$$\mathbf{H}_{(i)}^{(e)} = \begin{bmatrix} \frac{\partial N_i^e}{\partial x} \\ \frac{\partial N_i^e}{\partial y} \\ \frac{\partial N_i^e}{\partial z} \end{bmatrix} \quad \mathbf{J}^{-1} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} & \frac{\partial \mu}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} & \frac{\partial \mu}{\partial y} \\ \frac{\partial \xi}{\partial z} & \frac{\partial \eta}{\partial z} & \frac{\partial \mu}{\partial z} \end{bmatrix} \quad \hat{\mathbf{B}}_{(i)} = \begin{bmatrix} \frac{\partial N_i^e}{\partial \xi} \\ \frac{\partial N_i^e}{\partial \eta} \\ \frac{\partial N_i^e}{\partial \mu} \end{bmatrix} \quad (3.33)$$

Donde $\hat{\mathbf{B}}_{(i)}(\xi, \eta, \mu)$ es la matriz reducida de operadores diferenciales con respecto al sistema natural actuando sobre las funciones de forma del nudo i . La matriz \mathbf{J}^{-1} de tamaño 3×3 corresponde al inverso del Jacobiano de las coordenadas cartesianas (x, y, z) con respecto a las coordenadas naturales (ξ, η, μ) .

Asimismo, la matriz de 3×8 que contiene las derivadas de las funciones de forma en sistema global del elemento finito $\mathbf{H}^{(e)}$ esta conformada por las submatrices columna $\mathbf{H}_{(i)}^{(e)}$ de cada nudo, así:

$$\mathbf{H}^{(e)} = \begin{bmatrix} \mathbf{H}_{(1)}^{(e)} & \mathbf{H}_{(2)}^{(e)} & \cdots & \mathbf{H}_{(8)}^{(e)} \end{bmatrix} \quad (3.34)$$

De igual forma, la matriz de operadores diferenciales actuando sobre las funciones de forma en sistema coordinado natural $\hat{\mathbf{B}}^{(e)}$ para el elemento finito, tiene tamaño

3×8 y esta conformada por las submatrices columna $\hat{\mathbf{B}}^{(e)}_{(i)}$ de cada nudo, de la forma:

$$\hat{\mathbf{B}}^{(e)} = \left[\hat{\mathbf{B}}^{(e)}_{(1)} \quad \hat{\mathbf{B}}^{(e)}_{(2)} \quad \dots \quad \hat{\mathbf{B}}^{(e)}_{(8)} \right] \quad (3.35)$$

La matriz Jacobiana está definida como:

$$\mathbf{J} = \hat{\mathbf{B}}^{(e)} \mathbf{c}^{(e)} \quad (3.36)$$

siendo \mathbf{c} la matriz mostrada en la [Ecuación 3.24](#) y \mathbf{J} la matriz mostrada a continuación:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \mu} & \frac{\partial y}{\partial \mu} & \frac{\partial z}{\partial \mu} \end{bmatrix} \quad (3.37)$$

Para el cálculo de la matriz Jacobiana se requiere derivar parcialmente cada una de las 8 funciones de forma para ξ , η y μ , como se muestra de forma general en la [Ecuación 3.31a](#), conformando la matriz $\hat{\mathbf{B}}(\xi, \eta, \mu)$. Dicha matriz está conformada por los siguientes términos:

$$\hat{\mathbf{B}}^{(e)}(\xi, \eta, \mu) = \begin{bmatrix} \frac{\partial N_1(\xi, \eta, \mu)}{\partial \xi} & \dots & \frac{\partial N_8(\xi, \eta, \mu)}{\partial \xi} \\ \frac{\partial N_1(\xi, \eta, \mu)}{\partial \eta} & \dots & \frac{\partial N_8(\xi, \eta, \mu)}{\partial \eta} \\ \frac{\partial N_1(\xi, \eta, \mu)}{\partial \mu} & \dots & \frac{\partial N_8(\xi, \eta, \mu)}{\partial \mu} \end{bmatrix}_{3 \times 8} \quad (3.38a)$$

$$\hat{\mathbf{B}}^{(e)} = \frac{1}{8} \begin{bmatrix} -(1-\eta)(1-\mu) & (1-\eta)(1-\mu) & (1+\eta)(1-\mu) & -(1+\eta)(1-\mu) & & & & \\ -(1-\xi)(1-\mu) & -(1+\xi)(1-\mu) & (1+\xi)(1-\mu) & (1-\xi)(1-\mu) & \dots & & & \\ -(1-\xi)(1-\eta) & -(1+\xi)(1-\eta) & -(1+\xi)(1+\eta) & -(1-\xi)(1+\eta) & & & & \\ & & & & & & & \\ & -(1-\eta)(1+\mu) & (1-\eta)(1+\mu) & (1+\eta)(1+\mu) & -(1+\eta)(1+\mu) & & & \\ \dots & -(1-\xi)(1+\mu) & -(1+\xi)(1+\mu) & (1+\xi)(1+\mu) & (1-\xi)(1+\mu) & & & \\ & (1-\xi)(1-\eta) & (1+\xi)(1-\eta) & (1+\xi)(1+\eta) & (1-\xi)(1+\eta) & & & \end{bmatrix} \quad (3.38b)$$

3.6.5. Matriz Jacobiana y su inversa

La coordenada global de un punto (x, y, z) en el interior de un elemento finito isoparamétrico está definida por las coordenadas (x_i, y_i, z_i) de sus nudos $i = 1, \dots, nn$ de

la forma:

$$x = \sum_{i=1}^{nn} x_i N_i^e \quad y = \sum_{i=1}^{nn} y_i N_i^e \quad z = \sum_{i=1}^{nn} z_i N_i^e \quad (3.39)$$

En particular, el número de nudos del elemento hexaédrico es $nn = 8$.

De esta manera, al realizar el producto entre $\hat{\mathbf{B}}$ con la matriz de coordenadas de los nudos \mathbf{c} , se conforma la matriz \mathbf{J} , la cual se puede representar como:

$$\mathbf{J} = \begin{bmatrix} \sum_{i=1}^{nn} x_i \frac{\partial N_i^e}{\partial \xi} & \sum_{i=1}^{nn} y_i \frac{\partial N_i^e}{\partial \xi} & \sum_{i=1}^{nn} z_i \frac{\partial N_i^e}{\partial \xi} \\ \sum_{i=1}^{nn} x_i \frac{\partial N_i^e}{\partial \eta} & \sum_{i=1}^{nn} y_i \frac{\partial N_i^e}{\partial \eta} & \sum_{i=1}^{nn} z_i \frac{\partial N_i^e}{\partial \eta} \\ \sum_{i=1}^{nn} x_i \frac{\partial N_i^e}{\partial \mu} & \sum_{i=1}^{nn} y_i \frac{\partial N_i^e}{\partial \mu} & \sum_{i=1}^{nn} z_i \frac{\partial N_i^e}{\partial \mu} \end{bmatrix} \quad (3.40)$$

A continuación se muestra un procedimiento para calcular la inversa de la matriz \mathbf{J} , el cual es una simplificación del método de cofactores (Felippa, 2017a):

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad \mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|} \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & -A_{32} & A_{33} \end{bmatrix}^T \quad (3.41)$$

donde:

$$A_{11} = a_{22}a_{33} - a_{23}a_{32}, \quad (3.42a)$$

$$A_{22} = a_{33}a_{11} - a_{31}a_{13}, \quad (3.42b)$$

$$A_{33} = a_{11}a_{22} - a_{12}a_{21}, \quad (3.42c)$$

$$A_{12} = a_{23}a_{31} - a_{21}a_{33}, \quad (3.42d)$$

$$A_{23} = a_{31}a_{12} - a_{32}a_{11}, \quad (3.42e)$$

$$A_{31} = a_{12}a_{23} - a_{13}a_{22}, \quad (3.42f)$$

$$A_{21} = a_{32}a_{13} - a_{12}a_{33}, \quad (3.42g)$$

$$A_{32} = a_{13}a_{21} - a_{23}a_{11}, \quad (3.42h)$$

$$A_{13} = a_{21}a_{22} - a_{31}a_{22}, \quad (3.42i)$$

$$|\mathbf{A}| = a_{11}A_{11} + a_{12}A_{21} + a_{13}A_{31} \quad (3.42j)$$

Con la inversa de la matriz Jacobiana \mathbf{J}^{-1} y con la matriz $\hat{\mathbf{B}}_{(i)}$, utilizando la [Ecuación 3.32](#) se calcula la matriz $\mathbf{H}_{(i)}^{(e)}$

3.6.6. Matriz de operadores diferenciales actuando sobre las funciones de forma

La matriz de operadores diferenciales ∇ para problemas de la mecánica estructural tridimensionales con deformaciones infinitesimales, contiene las derivadas con respecto a los ejes coordenados globales y está definida como (Linero *et al.*, 2020):

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \end{bmatrix} \quad (3.43)$$

La matriz columna de la deformación $\boldsymbol{\varepsilon}^{(e)}$ en notación de ingeniería de un elemento finito tridimensional e , es de la forma:

$$\boldsymbol{\varepsilon}^{(e)} = [\varepsilon_{xx} \quad \varepsilon_{yy} \quad \varepsilon_{zz} \quad \gamma_{xy} \quad \gamma_{xz} \quad \gamma_{yz}]^T \quad (3.44)$$

La relación entre el campo del desplazamiento y el campo de la deformación en el interior del elemento finito es de la forma:

$$\boldsymbol{\varepsilon}^{(e)} = \nabla \mathbf{u}^{(e)} \quad (3.45)$$

Reemplazando la [Ecuación 3.25](#) que define a la matriz columna de desplazamientos nodales $\mathbf{a}^{(e)}$ en la expresión anterior, se obtiene que:

$$\boldsymbol{\varepsilon}^{(e)} = \nabla \mathbf{u}^{(e)} = \nabla \mathbf{N}^{(e)} \mathbf{a}^{(e)} = \mathbf{B}^{(e)} \mathbf{a}^{(e)} \quad (3.46)$$

La matriz de operadores diferenciales actuando sobre funciones de forma en coordenadas globales $\mathbf{B}^{(e)}$ para el elemento hexaédrico se expresa como:

$$\mathbf{B}^{(e)} = \begin{bmatrix} \frac{\partial N_1^e}{\partial x} & 0 & 0 & \frac{\partial N_2^e}{\partial x} & 0 & 0 & \dots & \frac{\partial N_8^e}{\partial x} & 0 & 0 \\ 0 & \frac{\partial N_1^e}{\partial y} & 0 & 0 & \frac{\partial N_2^e}{\partial y} & 0 & \dots & 0 & \frac{\partial N_8^e}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_1^e}{\partial z} & 0 & 0 & \frac{\partial N_2^e}{\partial z} & \dots & 0 & 0 & \frac{\partial N_8^e}{\partial z} \\ \frac{\partial N_1^e}{\partial y} & \frac{\partial N_1^e}{\partial x} & 0 & \frac{\partial N_2^e}{\partial y} & \frac{\partial N_2^e}{\partial x} & 0 & \dots & \frac{\partial N_n^e}{\partial y} & \frac{\partial N_n^e}{\partial x} & 0 \\ \frac{\partial N_1^e}{\partial z} & 0 & \frac{\partial N_1^e}{\partial x} & \frac{\partial N_2^e}{\partial z} & 0 & \frac{\partial N_2^e}{\partial x} & \dots & \frac{\partial N_n^e}{\partial z} & 0 & \frac{\partial N_n^e}{\partial x} \\ 0 & \frac{\partial N_1^e}{\partial z} & \frac{\partial N_1^e}{\partial y} & 0 & \frac{\partial N_2^e}{\partial z} & \frac{\partial N_2^e}{\partial y} & \dots & 0 & \frac{\partial N_n^e}{\partial z} & \frac{\partial N_n^e}{\partial y} \end{bmatrix} \quad (3.47)$$

La matriz $\mathbf{B}^{(e)}$ indicada en la ecuación anterior está conformada por 8 submatrices $\mathbf{B}_{(e)}^{(i)}$, asociadas a cada nudo i , así:

$$\mathbf{B}^{(e)} = \left[\mathbf{B}_{(e)}^{(1)} \quad \mathbf{B}_{(e)}^{(2)} \quad \dots \quad \mathbf{B}_{(e)}^{(8)} \right] \quad (3.48a)$$

donde,

$$\mathbf{B}_{(e)}^{(i)} = \begin{bmatrix} B_{11}^{(i)} & 0 & 0 \\ 0 & B_{22}^{(i)} & 0 \\ 0 & 0 & B_{33}^{(i)} \\ B_{41}^{(i)} & B_{42}^{(i)} & 0 \\ B_{51}^{(i)} & 0 & B_{53}^{(i)} \\ 0 & B_{62}^{(i)} & B_{63}^{(i)} \end{bmatrix}_{6 \times 3} \quad (3.48b)$$

En consecuencia,

$$\mathbf{B}^{(e)} = \begin{bmatrix} B_{11}^{(1)} & 0 & 0 & \dots & B_{11}^{(8)} & 0 & 0 \\ 0 & B_{22}^{(1)} & 0 & \dots & 0 & B_{22}^{(8)} & 0 \\ 0 & 0 & B_{33}^{(1)} & \dots & 0 & 0 & B_{33}^{(8)} \\ B_{41}^{(1)} & B_{42}^{(1)} & 0 & \dots & B_{41}^{(8)} & B_{42}^{(8)} & 0 \\ B_{51}^{(1)} & 0 & B_{53}^{(1)} & \dots & B_{51}^{(8)} & 0 & B_{53}^{(8)} \\ 0 & B_{62}^{(1)} & B_{63}^{(1)} & \dots & 0 & B_{62}^{(8)} & B_{63}^{(8)} \end{bmatrix}_{6 \times 24} \quad (3.48c)$$

Para conformar la matriz $\mathbf{B}^{(e)}$ se utilizan las componentes de la matriz $\mathbf{H}^{(e)}$, mostrada en la [Ecuación 3.34](#). A continuación se muestran los coeficientes correspondientes al nudo i de la matriz $\mathbf{B}^{(e)}$:

$$B_{11}^{(i)} = \frac{\partial N_1}{\partial x} = \hat{B}_{11} J_{11}^{-1} + \hat{B}_{21} J_{12}^{-1} + \hat{B}_{31} J_{13}^{-1} = H_{11}^{(i)} \quad (3.49a)$$

$$B_{11}^{(i)} = \frac{\partial N_1}{\partial x} = \frac{\partial N_1}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial N_1}{\partial \eta} \frac{\partial \eta}{\partial x} + \frac{\partial N_1}{\partial \mu} \frac{\partial \mu}{\partial x} = H_{11}^{(i)} \quad (3.49b)$$

$$B_{22}^{(i)} = \frac{\partial N_1}{\partial y} = \frac{\partial N_1}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial N_1}{\partial \eta} \frac{\partial \eta}{\partial y} + \frac{\partial N_1}{\partial \mu} \frac{\partial \mu}{\partial y} = H_{22}^{(i)} \quad (3.49c)$$

$$B_{33}^{(i)} = \frac{\partial N_1}{\partial z} = \frac{\partial N_1}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial N_1}{\partial \eta} \frac{\partial \eta}{\partial z} + \frac{\partial N_1}{\partial \mu} \frac{\partial \mu}{\partial z} = H_{33}^{(i)} \quad (3.49d)$$

$$B_{12}^{(i)} = 0 \quad B_{13}^{(i)} = 0 \quad B_{21}^{(i)} = 0 \quad B_{23}^{(i)} = 0 \quad B_{31}^{(i)} = 0 \quad (3.49e)$$

$$B_{32}^{(i)} = 0 \quad B_{43}^{(i)} = 0 \quad B_{61}^{(i)} = 0 \quad B_{52}^{(i)} = 0 \quad (3.49f)$$

$$B_{41}^{(i)} = H_{22}^{(i)} \quad B_{63}^{(i)} = H_{22}^{(i)} \quad (3.49g)$$

$$B_{42}^{(i)} = H_{11}^{(i)} \quad B_{53}^{(i)} = H_{11}^{(i)} \quad (3.49h)$$

$$B_{51}^{(i)} = H_{33}^{(i)} \quad B_{62}^{(i)} = H_{33}^{(i)} \quad (3.49i)$$

3.6.7. Matriz constitutiva del material elástico lineal isotrópico

La matriz constitutiva \mathbf{D} de un material elástico lineal isotrópico con relación de Poisson ν y módulo de Young E , se define como se muestra a continuación (Linero *et al.*, 2020):

$$\mathbf{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & \nu & 0 & 0 & 0 \\ & (1-\nu) & \nu & 0 & 0 & 0 \\ & & (1-\nu) & 0 & 0 & 0 \\ & & & \frac{1}{2}(1-2\nu) & 0 & 0 \\ sim. & & & & \frac{1}{2}(1-2\nu) & 0 \\ & & & & & \frac{1}{2}(1-2\nu) \end{bmatrix} \quad (3.50)$$

3.6.8. Matriz de rigidez del hexaedro

A partir de la aplicación del Principio de los trabajos virtuales al método de los elementos finitos para problemas mecánicos estáticos tridimensionales, se tiene que (Felippa, 2017b; Linero *et al.*, 2013; Zienkiewicz, 2013):

$$\sum_{e=1}^{ne} \delta \mathbf{a}_{(e)}^T \mathbf{K}^{(e)} \mathbf{a}_{(e)} - \sum_{e=1}^{ne} \delta \mathbf{a}_{(e)}^T \tilde{\mathbf{f}}^{(e)} - \delta \mathbf{a}_{(e)}^T \mathbf{f}^{(n)} = 0 \quad (3.51)$$

En la anterior ecuación, $\tilde{\mathbf{f}}$ es la suma del vector de fuerzas nodales equivalentes a la acción de fuerzas de superficie $\tilde{\mathbf{f}}_s$ con el vector de fuerzas nodales equivalentes a

la acción de fuerzas de cuerpo $\tilde{\mathbf{f}}_b$, los cuales se mostrarán desde la [Subsección 3.6.9](#) hasta la [Subsección 3.6.12](#). En cambio, el vector $\mathbf{f}^{(n)}$ contiene las fuerzas aplicadas directamente en los nudos de la malla y se explicará en la [Subsección 3.6.13](#). $\mathbf{K}^{(e)}$ se define como la matriz de rigidez del elemento finito, que corresponde a la siguiente integral de volumen:

$$\mathbf{K}^{(e)} = \int_{V^{(e)}} \mathbf{B}_{(e)}^T \mathbf{D}_{(e)} \mathbf{B}_{(e)} dV \quad (3.52)$$

donde $\mathbf{B}_{(e)}$ corresponde a la matriz de operadores diferenciales actuando sobre las funciones de forma mostrada en la [Subsección 3.6.6](#). \mathbf{D} es la matriz constitutiva del material, como se muestra en la [Subsección 3.6.7](#).

La integral en el volumen del elemento finito se evalúa en las coordenadas naturales ξ , η y μ indicadas en la [Subsección 3.6.2](#), definiendo al diferencial de volumen dV , así:

$$dV = dx dy dz = \det \mathbf{J}^{(e)} d\xi d\eta d\mu \quad (3.53)$$

siendo $\det \mathbf{J}^{(e)}$ el determinante de la matriz Jacobiano. En consecuencia la matriz de rigidez de un elemento finito tridimensional es de la forma:

$$\mathbf{K}^{(e)} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \mathbf{B}_{(e)}^T \mathbf{D}_{(e)} \mathbf{B}_{(e)} \det \mathbf{J}^{(e)} d\xi d\eta d\mu \quad (3.54)$$

La integral anterior se calcula numéricamente utilizando el método de la cuadratura de Gauss (Zienkiewicz, 2013). Esto se logra a través de una sumatoria ponderada de la función por los coeficientes ω_i , ω_j y ω_k de cada eje coordenado natural, evaluada en coordenadas específicas (ξ_i, η_j, μ_k) , conocidas como puntos de Gauss. La aproximación de la integral depende de la cantidad de puntos de Gauss ni , nj y nk , establecidos para los ejes ξ , η y μ , respectivamente. Por lo tanto, la matriz de rigidez de un elemento finito tridimensional es aproximadamente igual a:

$$\mathbf{K}^{(e)} = \sum_{i=1}^{ni} \sum_{j=1}^{nj} \sum_{k=1}^{nk} \omega_i \omega_j \omega_k \mathbf{B}_{(e)}^T(\xi_i, \eta_j, \mu_k) \mathbf{D} \mathbf{B}_{(e)}(\xi_i, \eta_j, \mu_k) \det \mathbf{J}^{(e)}(\xi_i, \eta_j, \mu_k) \quad (3.55)$$

La ecuación anterior puede simplificarse a una sola sumatoria, de la forma:

$$\mathbf{K}^{(e)} = \sum_{g=1}^{ng} \omega_g \mathbf{B}_{(e)}^T(\xi_g, \eta_g, \mu_g) \mathbf{D} \mathbf{B}_{(e)}(\xi_g, \eta_g, \mu_g) \det \mathbf{J}^{(e)}(\xi_g, \eta_g, \mu_g) \quad (3.56)$$

donde ω_g es el coeficiente de ponderación simplificado y (ξ_g, η_g, μ_g) es la coordenada natural en un punto de Gauss g .

(a) $ng = 1 \times 1 \times 1 = 1$					(b) $ng = 2 \times 2 \times 2 = 8$				
pg	ξ_g	η_g	μ_g	ω_g	pg	ξ_g	η_g	μ_g	ω_g
G1	0	0	0	4	G1	$-\sqrt{1/3}$	$-\sqrt{1/3}$	$-\sqrt{1/3}$	1
					G2	$+\sqrt{1/3}$	$-\sqrt{1/3}$	$-\sqrt{1/3}$	1
					G3	$+\sqrt{1/3}$	$+\sqrt{1/3}$	$-\sqrt{1/3}$	1
					G4	$-\sqrt{1/3}$	$+\sqrt{1/3}$	$-\sqrt{1/3}$	1
					G5	$-\sqrt{1/3}$	$-\sqrt{1/3}$	$+\sqrt{1/3}$	1
					G6	$+\sqrt{1/3}$	$-\sqrt{1/3}$	$+\sqrt{1/3}$	1
					G7	$+\sqrt{1/3}$	$+\sqrt{1/3}$	$+\sqrt{1/3}$	1
					G8	$-\sqrt{1/3}$	$+\sqrt{1/3}$	$+\sqrt{1/3}$	1

Tabla 3.2: Coordenadas naturales y coeficientes de ponderación del método de la cuadratura de Gauss para un elemento finito hexaédrico, considerando: (a) $1 \times 1 \times 1 = 1$ punto de Gauss y (b) $2 \times 2 \times 2 = 8$ puntos de Gauss.

El primer orden completo de aproximación de la integral corresponde a un punto de Gauss $ng = 1 \times 1 \times 1 = 1$, cuya coordenada y coeficiente de ponderación se indica en la [Tabla 3.2\(a\)](#). El segundo orden completo de aproximación se obtiene utilizando 2 puntos de Gauss asociados a cada coordenada natural, es decir un total de $ng = ni \times nj \times nk = 2 \times 2 \times 2 = 8$ puntos de Gauss. Las coordenadas y coeficientes de ponderación de este orden de aproximación se presentan en la [Tabla 3.2\(b\)](#). El tercer orden completo de aproximación evalúa las sumatorias en 3 puntos por cada eje natural, para un total de $ng = ni \times nj \times nk = 3 \times 3 \times 3 = 27$ puntos de Gauss.

En la [Figura 3.4](#) se muestra la ubicación de los puntos de integración de Gauss para $ng = 1$ y $ng = 8$ (Felippa, 2017a). Al igual que en la [Tabla 3.2](#), G1, G2, ..., G8, indican los puntos de Gauss en cada orden de aproximación.

La matriz de rigidez de un elemento hexaédrico de 8 nudos alcanza una aproximación satisfactoria para $ng = 8$, utilizando el método de integración numérica de Gauss (Felippa, 2017a). Razón por la cual, en este trabajo se implementó dicho orden de aproximación sobre el elemento finito.

3.6.9. Vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento

El vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento $\tilde{\mathbf{f}}_b^{(e)}$, así como el vector de fuerzas equivalentes a la acción de presiones $\tilde{\mathbf{f}}_s^{(e)}$, el cual se analizará en las siguientes secciones, proviene de la [Ecuación 3.51](#). Estos vectores

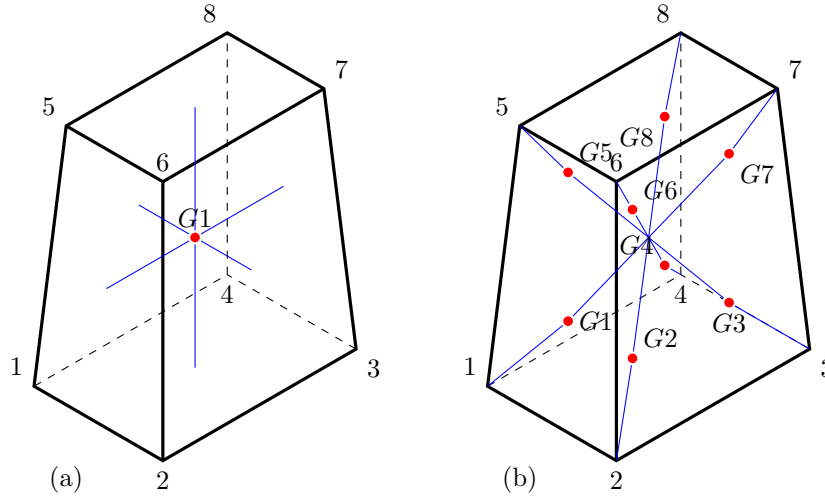


Figura 3.4: Distribución de los puntos de Gauss para el elemento hexaédrico de 8 nudos a)1 punto b)8 puntos

conforman al vector $\tilde{\mathbf{f}}^{(e)}$, así:

$$\tilde{\mathbf{f}}^{(e)} = \tilde{\mathbf{f}}_s^{(e)} + \tilde{\mathbf{f}}_b^{(e)} \quad (3.57)$$

El vector $\tilde{\mathbf{f}}_b^{(e)}$ corresponde a las fuerzas equivalentes a la acción de las cargas por unidad de volumen, producidas por las fuerzas másicas en el interior del elemento finito.

Al igual que en el cálculo de la matriz de rigidez, el diferencial de volumen del elemento hexaédrico es $dV = dx dy dz = \det \mathbf{J}^{(e)} d\xi d\eta d\mu$, por lo que resulta:

$$\tilde{\mathbf{f}}_b^{(e)} = \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N}_{(e)}^T \mathbf{b}_{(e)} \det \mathbf{J}^{(e)} d\xi d\eta d\mu \quad (3.58)$$

siendo $\mathbf{b}_{(e)}$ el vector de fuerzas de cuerpo, el cual contiene las componentes de la fuerza por unidad de volumen del elemento en cada dirección global, así:

$$\mathbf{b}^{(e)} = \left[\mathbf{b}_1^{(e)T} \quad \mathbf{b}_2^{(e)T} \quad \dots \quad \mathbf{b}_8^{(e)T} \right]_{1 \times 24}^T \quad (3.59a)$$

siendo

$$\mathbf{b}_i^{(e)} = [b_{ix} \quad b_{iy} \quad b_{iz}]^T \quad (3.59b)$$

Considerando que las cargas por unidad de volumen corresponden al peso propio del material, orientado en dirección $-y$ y con peso específico γ (Figura 3.3), se tiene:

$$\mathbf{b}_i^{(e)} = [0 \quad -\gamma \quad 0]^T \quad (3.60a)$$

por lo tanto el vector $\mathbf{b}^{(e)}$ para el elemento hexaédrico es de la forma:

$$\mathbf{b}^{(e)} = [0 \quad -\gamma \quad 0 \quad \dots \quad 0 \quad -\gamma \quad 0]_{1 \times 24}^T \quad (3.60b)$$

De manera similar al cálculo de la matriz de rigidez, esta integral triple se puede calcular utilizando el método de integración numérica de Gauss, así:

$$\tilde{\mathbf{f}}_b^{(e)} = \sum_{g=1}^{ng} \omega_g \mathbf{N}_{(e)}^T(\xi_g, \eta_g, \mu_g) \mathbf{b}_{(e)} \det \mathbf{J}^{(e)}(\xi_g, \eta_g, \mu_g) \quad (3.61)$$

donde ω_g es el coeficiente de ponderación simplificado, (ξ_g, η_g, μ_g) es la coordenada natural en un punto de Gauss g y $\det \mathbf{J}^{(e)}(\xi_g, \eta_g, \mu_g)$ corresponde al determinante del Jacobiano mostrado en la [Subsección 3.6.5](#).

3.6.10. Vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global

El vector $\tilde{\mathbf{f}}_s^{(e)}$ es un vector que contiene las fuerzas equivalentes debidas a una presión uniforme $p^{(e)}$, referida al sistema de coordenadas global. Esta presión uniforme puede actuar sobre una o varias de las nc caras del elemento hexaédrico, de la forma:

$$\tilde{\mathbf{f}}_s^{(e)} = \int_{\Gamma^{(e)}} \mathbf{N}_{(e)}^T \mathbf{p}_{(e)} dS = \sum_{c=1}^{nc} \tilde{\mathbf{f}}_s^{(e)(c)} \quad (3.62)$$

siendo, $\tilde{\mathbf{f}}_s^{(e)(c)}$ el vector de fuerzas equivalentes a la acción de una presión uniforme sobre el área de la cara $A_c^{(e)}$, definido como:

$$\tilde{\mathbf{f}}_s^{(e)(c)} = \int_{A_c^{(e)}} \mathbf{N}_{(e)}^T \mathbf{p}_c^{(e)} dS \quad (3.63)$$

Este vector está conformado por subvectores asociados a cada nudo $\tilde{\mathbf{f}}_{si}^{(e)(c)}$, así:

$$\tilde{\mathbf{f}}_s^{(e)(c)} = \begin{bmatrix} \tilde{\mathbf{f}}_{s1}^{(e)(c)} \\ \tilde{\mathbf{f}}_{s2}^{(e)(c)} \\ \vdots \\ \tilde{\mathbf{f}}_{s8}^{(e)(c)} \end{bmatrix} \quad (3.64)$$

donde cada subvector es de la forma:

$$\tilde{\mathbf{f}}_{sci}^{(e)} = \int_{A_c^{(e)}} \mathbf{N}_{(i)}^{(e)T} \mathbf{p}_c^{(e)} dA \quad (3.65)$$

En esta ecuación, $\mathbf{p}_c^{(e)}$ es el vector de componentes de presión sobre la cara c en sistema coordenado global, expresado como:

$$\tilde{\mathbf{p}}_c^{(e)} = \begin{bmatrix} p_{xc}^{(e)} \\ p_{yc}^{(e)} \\ p_{zc}^{(e)} \end{bmatrix} \quad (3.66)$$

Sustituyendo las funciones de forma del elemento, se obtienen las fuerzas equivalentes en los nudos debidas a la presión $\mathbf{p}_c^{(e)}$, así:

$$\tilde{\mathbf{f}}_{sci}^{(e)} = \frac{A_c^{(e)}}{4} \begin{bmatrix} p_{xc}^{(e)} \\ p_{yc}^{(e)} \\ p_{zc}^{(e)} \end{bmatrix} \begin{cases} i = 1, 2, 3, 4, 5, 6, 7, 8 \\ c = (1234)(1265)(3487)(2376)(1584)(5678) \end{cases} \quad (3.67)$$

Se observa que dicha presión es equivalente a una fuerza de igual magnitud en cada uno de los nudos de la cara c donde está aplicada.

Cabe mencionar que para cada $\tilde{\mathbf{f}}_{si}^{(e)}$, en los nudos del elemento que no están contenidos en la cara c , la fuerza debida a la presión $\mathbf{p}_c^{(e)}$ es igual a cero. Por ejemplo, para una presión actuando en la cara (1234) el vector $\tilde{\mathbf{f}}_{si}^{(e)(1234)}$ contiene las fuerzas equivalentes a la presión actuando únicamente en los nudos 1, 2, 3 y 4.

$$\tilde{\mathbf{f}}_s^{(e)(1234)} = \frac{A_{1234}^{(e)}}{4} \begin{bmatrix} \mathbf{p}_{1234}^{(e)} & \mathbf{p}_{1234}^{(e)} & \mathbf{p}_{1234}^{(e)} & \mathbf{p}_{1234}^{(e)} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \end{bmatrix}_{1 \times 24}^T$$

donde

$$\mathbf{p}_{1234}^{(e)} = \begin{bmatrix} p_{x1234}^{(e)} & p_{y1234}^{(e)} & p_{z1234}^{(e)} \end{bmatrix}, \quad \mathbf{O} = [0 \ 0 \ 0]$$

De igual manera, para una presión uniforme $\mathbf{p}_c^{(e)}$ referidas al sistema global de coordenadas actuando sobre la cara (1265), el vector $\tilde{\mathbf{f}}_{si}^{(e)(1265)}$ es:

$$\tilde{\mathbf{f}}_s^{(e)(1265)} = \frac{A_{1265}^{(e)}}{4} \begin{bmatrix} \mathbf{p}_{1265}^{(e)} & \mathbf{p}_{1265}^{(e)} & \mathbf{O} & \mathbf{O} & \mathbf{p}_{1265}^{(e)} & \mathbf{p}_{1265}^{(e)} & \mathbf{O} & \mathbf{O} \end{bmatrix}_{1 \times 24}^T$$

donde

$$\mathbf{p}_{1265}^{(e)} = \begin{bmatrix} p_{x1265}^{(e)} & p_{y1265}^{(e)} & p_{z1265}^{(e)} \end{bmatrix}, \quad \mathbf{O} = [0 \ 0 \ 0]$$

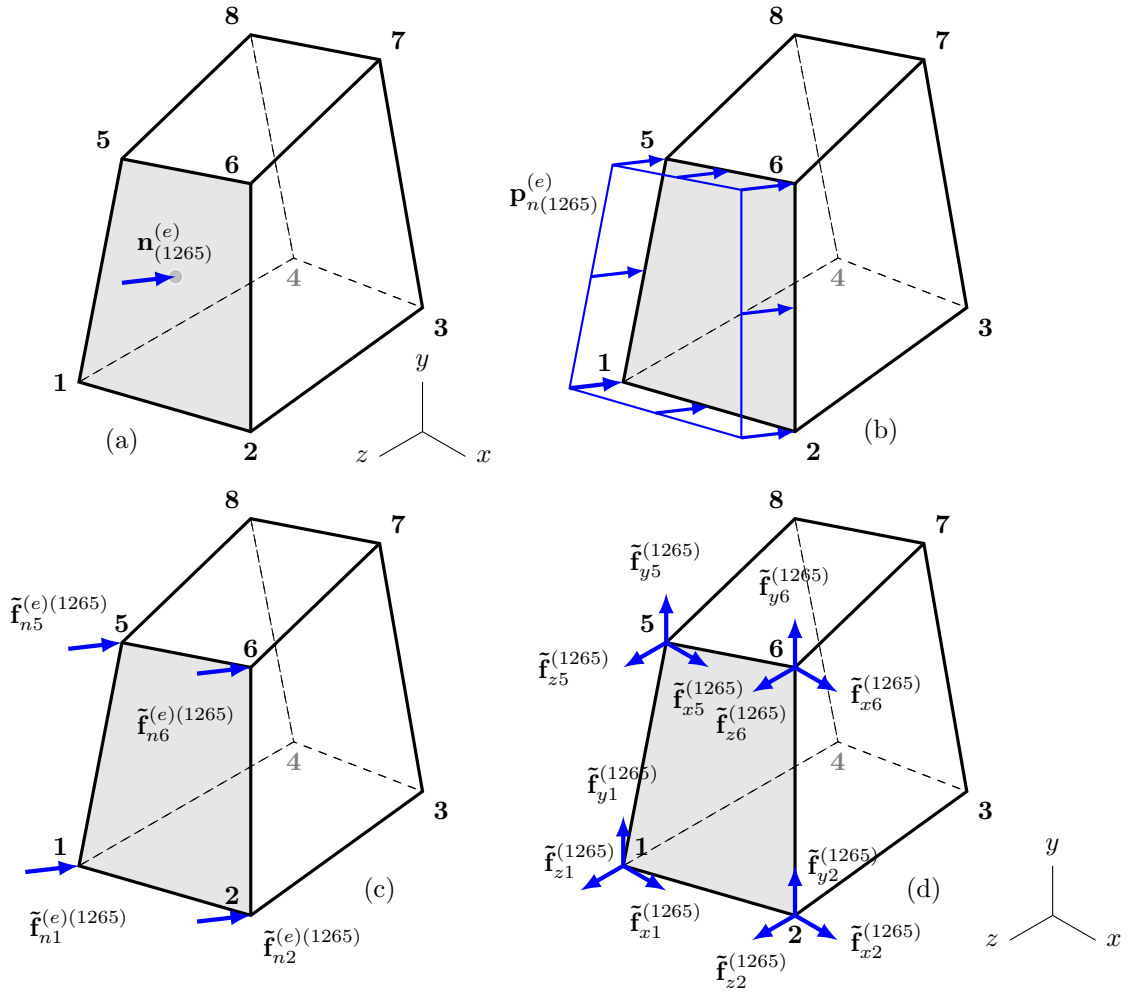


Figura 3.5: Presión uniforme normal que entra a la cara 1-2-6-5 del hexaedro: (a) vector normal entrando a la cara, (b) presión uniforme normal que entra a la cara, (c) fuerzas equivalentes en dirección normal, y (d) componentes de las fuerzas equivalentes en sistema coordenado global.

3.6.11. Vector de fuerzas equivalentes a la acción de una presión uniforme normal a una cara del elemento

La cara c de un elemento finito e tiene un vector normal unitario a su superficie $\mathbf{n}_{(c)}^{(e)}$ entrando al elemento, como se muestra en la [Figura 3.5\(a\)](#), el cual está definido en coordenadas globales así:

$$\mathbf{n}_c^{(e)} = \begin{bmatrix} n_x^{(c)} & n_y^{(c)} & n_z^{(c)} \end{bmatrix}^T \quad (3.68)$$

Con este vector es posible transformar la presión $p_{nc}^{(e)}$ mostrada en [Figura 3.5\(b\)](#) a fuerzas equivalentes aplicadas en cada uno de los i nudos de la cara como se muestra en [Figura 3.5\(c\)](#), en donde cada componente de fuerza en el nudo equivale a:

$$\tilde{f}_{ni}^{(e)(c)} = \frac{1}{4} p_{nc}^{(e)} A_{nc}^{(e)} \quad (3.69)$$

De esta manera, el subvector de fuerzas equivalentes en el nudo i de la cara c con respecto al sistema coordenado global es:

$$\tilde{\mathbf{f}}_{si}^{(e)(c)} = \tilde{f}_{ni}^{(e)(c)} \mathbf{n}^{(c)} \quad (3.70a)$$

$$\tilde{\mathbf{f}}_{si}^{(e)(c)} = \tilde{f}_{ni}^{(e)(c)} \begin{bmatrix} n_x^{(c)} & n_y^{(c)} & n_z^{(c)} \end{bmatrix}^T \quad (3.70b)$$

Para obtener el vector de fuerzas equivalentes completo $\tilde{\mathbf{f}}_s^{(e)(c)}$ correspondiente a la presión normal a la cara c del elemento e , cuyas componentes se muestran en [Figura 3.5\(d\)](#), se considera que en los nudos del elemento que no hacen parte de la cara c , la fuerza equivalente es igual a cero.

Por ejemplo, si se aplica una presión normal a la cara (1265), el vector de fuerzas equivalentes completo $\tilde{\mathbf{f}}_s^{(e)(1265)}$ es:

$$\tilde{\mathbf{f}}_s^{(e)(1265)} = \frac{1}{4} p_{n1265}^{(e)} A_{1265}^{(e)} \begin{bmatrix} \mathbf{n}_{1265}^{(e)} & \mathbf{n}_{1265}^{(e)} & \mathbf{O} & \mathbf{O} & \mathbf{n}_{1265}^{(e)} & \mathbf{n}_{1265}^{(e)} & \mathbf{O} & \mathbf{O} \end{bmatrix}_{1 \times 24}^T \quad (3.71)$$

donde

$$\mathbf{n}_{1265}^{(e)} = \begin{bmatrix} n_{x1265}^{(e)} & n_{y1265}^{(e)} & n_{z1265}^{(e)} \end{bmatrix}, \quad \mathbf{O} = [0 \ 0 \ 0]$$

Para el cálculo del vector unitario normal $\mathbf{n}_c^{(e)}$ que entra a la cara (1265), se puede seguir el siguiente procedimiento (Linero *et al.*, 2020):

- Calcular los vectores unitarios $\mathbf{m}^{(12)}$ y $\mathbf{m}^{(15)}$, los cuales determinan la dirección de los lados 1 – 2 y 1 – 5 y hacen parte de la cara cargada (1265).
- Calcular el vector normal a la cara cargada $\tilde{\mathbf{n}}^{(1265)}$, mediante la operación $\tilde{\mathbf{n}}^{(1265)} = \mathbf{m}^{(12)} \times \mathbf{m}^{(15)}$
- Calcular el vector unitario $\mathbf{m}^{(14)}$, quien define la dirección del lado 1 – 4, el cual no pertenece a la cara cargada (1265). Esto para garantizar que la dirección del vector normal a la cara cargada sea entrando al elemento.

- Definir la dirección del vector normal entrando al elemento así:

$$\tilde{\mathbf{n}}^{(1265)} = \begin{cases} \tilde{\mathbf{n}}^{(1265)} & \text{si } \tilde{\mathbf{n}}^{(1265)} \cdot \mathbf{m}^{(14)} > 0 \\ -\tilde{\mathbf{n}}^{(1265)} & \text{si } \tilde{\mathbf{n}}^{(1265)} \cdot \mathbf{m}^{(14)} < 0 \end{cases} \quad (3.72)$$

- Normalizar el vector $\tilde{\mathbf{n}}^{(1265)}$:

$$\mathbf{n}^{(1265)} = \frac{\tilde{\mathbf{n}}^{(1265)}}{|\tilde{\mathbf{n}}^{(1265)}|} \quad (3.73)$$

- El vector normal unitario $\mathbf{n}^{(1265)}$ referido al sistema de coordenadas global xyz es:

$$\mathbf{n}^{(1265)} = \begin{bmatrix} n_x^{(1265)} & n_y^{(1265)} & n_z^{(1265)} \end{bmatrix}^T \quad (3.74)$$

3.6.12. Vector de fuerzas equivalentes a la acción de una presión hidrostática

Se define la presión causada por el agua sobre la superficie de un sólido en función de su profundidad en dirección y (Linero *et al.*, 2020), como:

$$p_{nw} = \begin{cases} \gamma_w(h_w - y) & \text{si } y < h_w \\ 0 & \text{si } y \geq h_w \end{cases} \quad (3.75)$$

Siendo γ_w el peso específico del agua y h_w la posición máxima del agua con respecto a y .

La presión $p_{nw}^{(e)}$ se considera uniforme en la cara del elemento finito e y se evalúa a la profundidad $y_c^{(e)}$ del centroide de dicha cara. Por lo tanto la presión uniforme normal que actúa en la cara del elemento finito e donde interviene una presión hidrostática es:

$$p_{nw}^{(e)} = \begin{cases} \gamma_w(h_w - y_c^{(e)}) & \text{si } y_c^{(e)} < h_w \\ 0 & \text{si } y_c^{(e)} \geq h_w \end{cases} \quad (3.76)$$

Para el cálculo de las fuerzas en los nudos equivalentes a la acción de una presión uniforme normal a una cara del elemento se utiliza el procedimiento indicado en la [Subsección 3.6.11](#).

3.6.13. Vector de fuerzas aplicadas directamente en los nudos de la malla

Este vector denominado $\mathbf{f}^{(n)}$, contiene todas las fuerzas aplicadas directamente a los nudos de la malla, cada una de las cuales está asociada a un grado de libertad. Este vector se utiliza para formar el vector de fuerzas nodales totales de la malla \mathbf{f} , como se muestra en la [Ecuación 3.6](#). Asimismo, contiene tanto las fuerzas conocidas, las cuales están asociadas a los grados de libertad desconocidos, como las fuerzas desconocidas o reacciones, asociadas a los grados de libertad conocidos, como se muestra en la [Sección 3.2](#).

3.7. Formulación del método de los elementos finitos para el elemento prisma o cuña

Para el enmallado de algunos sólidos, de acuerdo a su geometría es conveniente utilizar alguna de las formas modificadas del hexaedro. En esta sección se muestra una de las formas modificadas del elemento hexaédrico, el pentaedro, también conocido como prisma o elemento tipo cuña.

Esta geometría complementa adecuadamente al elemento hexaedro cuando se requiere el enmallado de algunos tipos de sólidos con formas cilíndricas o similares.

El elemento tipo prisma se obtiene cuando dos parejas de nudos del elemento hexaédrico se fusionan, tal como se muestra en la [Figura 3.1\(b\)](#). Esto genera un elemento de 6 nudos, 9 lados y 5 caras.

Debido a estos cambios en la geometría del elemento hexaédrico original, es necesario modificar las funciones de forma (Zienkiewicz, 2013).

3.7.1. Numeración de los nudos

Se considera que las parejas de nudos 3 – 4 y 7 – 8 de la geometría original se unen, por lo que es conveniente renombrar los nudos, como se muestra en la [Figura 3.6](#). La reglas de numeración de los nudos del elemento prisma son las siguientes:

- Tomar un vértice inicial y designarlo como 1, luego seleccionar una cara triangular a la cual pertenezca ese vértice.
- Designar los otros tres vértices de la cara seleccionada en el ítem anterior con los números 2 y 3, recorriendo los vértices en el sentido anti-horario de un vector normal a la cara seleccionada entrando al elemento.
- Designar el nudo 4 en el vértice de la cara opuesta que se conecta con una línea al nudo 1.

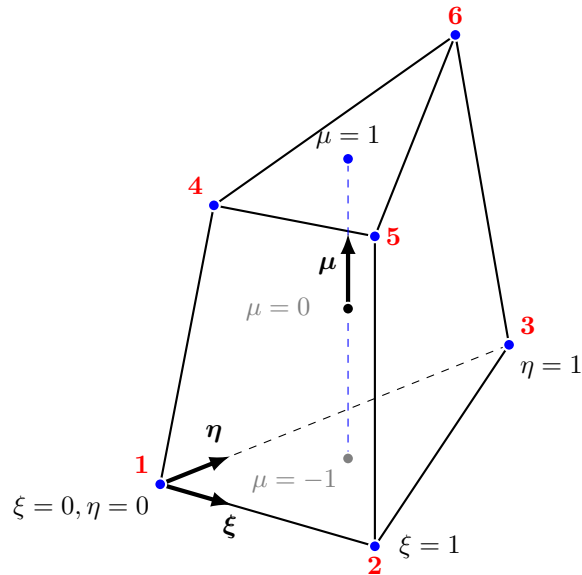


Figura 3.6: Numeración de los nudos y coordenadas naturales en el elemento prisma

- Designar los demás vértices de la cara opuesta con los números 5 y 6, recorriéndolos en sentido horario a un vector unitario a la cara opuesta entrando al elemento

3.7.2. Coordenadas naturales

El elemento prisma presenta tres caras cuadrilaterales de cuatro nudos y dos caras triangulares de tres nudos, como se muestra en la [Figura 3.6](#). Por esta razón, la función de aproximación en un punto está definida por una interpolación lineal en coordenadas naturales en las direcciones ξ y η , y una interpolación isoparamétrica lineal en la dirección μ (Zienkiewicz, 2013).

La [Figura 3.6](#) indica las coordenadas naturales (ξ, η, μ) de los nudos del elemento prisma y el origen de cada eje coordenado. Particularmente, las coordenadas naturales ξ y η toman valores entre 0 y 1, siendo $\xi = 0$ en el nudo 1 y $\xi = 1$ en el nudo 2. Asimismo, η es igual a cero en el nudo 1 y corresponde a 1 en el nudo 3. Por otro lado, la coordenada natural μ es igual a -1 en los puntos sobre la cara (123), mientras que toma un valor de $+1$ en los puntos sobre la cara (456). Las coordenadas naturales para los nudos, lados y caras del elemento finito se muestran en la [Tabla 3.3](#).

Al igual que el elemento hexaédrico de ocho nudos, el prisma está conformado por un nudo en cada uno de sus seis vértices de coordenadas (x_i, y_i, z_i) . Cada nudo

nudo	ξ	η	μ	lado	coordenada	cara	coordenada
1	0	0	-1	1-2	$\eta = 0, \mu = -1$	1-3-2	$\mu = -1$
2	1	0	-1	1-3	$\xi = 0, \mu = -1$	4-5-6	$\mu = +1$
3	0	1	-1	2-3	$\xi = 1, \eta = 1, \mu = -1$	1-2-5-4	$\eta = 0$
4	0	0	+1	4-5	$\eta = 0, \mu = 1$	1-4-6-3	$\xi = 0$
5	1	0	+1	4-6	$\xi = 0, \mu = 1$	2-3-6-5	$\xi = 1, \eta = 1$
6	0	1	+1	5-6	$\xi = 1, \eta = 1, \mu = +1$		
				1-4	$\xi = 0, \eta = 0$		
				2-5	$\xi = 1, \eta = 0$		
				3-6	$\xi = 0, \eta = 1$		

Tabla 3.3: Coordenadas naturales para el prisma: a la izquierda coordenadas de los nudos, al centro coordenadas de los lados y a la derecha coordenadas de las caras.

tiene tres grados de libertad, asociados a las componentes de desplazamiento $u_i^{(e)}$, $v_i^{(e)}$ y $w_i^{(e)}$. Por lo tanto el elemento prisma tiene en total 18 grados de libertad, como se muestra en la [Figura 3.7](#).

3.7.3. Funciones de forma

La matriz de funciones de forma del elemento prisma e está conformada por las submatrices $\mathbf{N}_i^{(e)}$ asociadas a cada nudo i , así:

$$\mathbf{N}^{(e)} = \left[\mathbf{N}_1^{(e)} \quad \mathbf{N}_2^{(e)} \quad \dots \quad \mathbf{N}_6^{(e)} \right] \quad (3.77a)$$

siendo,

$$\mathbf{N}_i^{(e)} = \begin{bmatrix} N_i^{(e)} & 0 & 0 \\ 0 & N_i^{(e)} & 0 \\ 0 & 0 & N_i^{(e)} \end{bmatrix} \quad (3.77b)$$

Por lo tanto, $\mathbf{N}^{(e)}$ es una matriz de 3 filas y 18 columnas con los siguientes términos:

$$\mathbf{N}^{(e)} = \begin{bmatrix} N_1^{(e)} & 0 & 0 & N_2^{(e)} & 0 & 0 & \dots & N_6^{(e)} & 0 & 0 \\ 0 & N_1^{(e)} & 0 & 0 & N_2^{(e)} & 0 & \dots & 0 & N_6^{(e)} & 0 \\ 0 & 0 & N_1^{(e)} & 0 & 0 & N_2^{(e)} & \dots & 0 & 0 & N_6^{(e)} \end{bmatrix}_{3 \times 18} \quad (3.77c)$$

Las seis funciones de forma del elemento prisma en función de las coordenadas

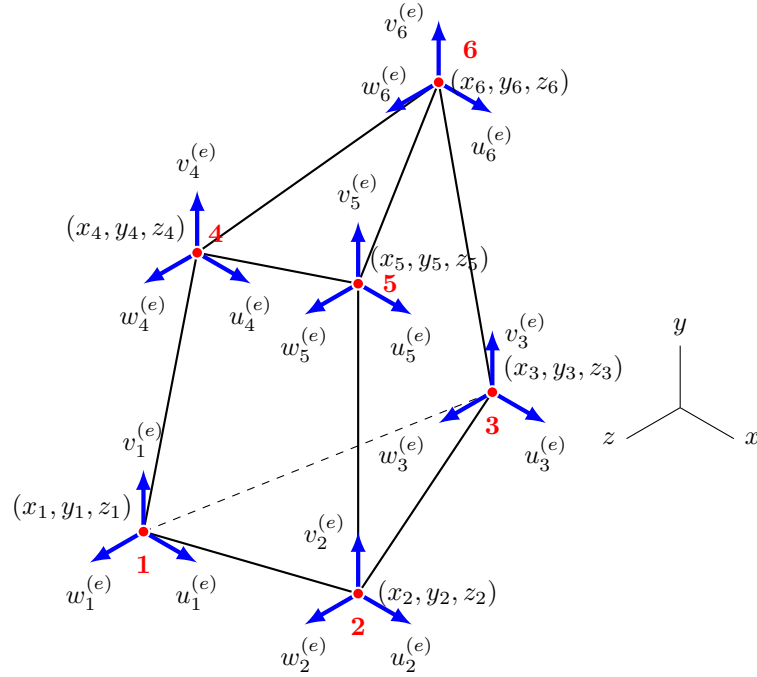


Figura 3.7: Grados de libertad del elemento prisma

naturales son:

$$N_1^{(e)} = \frac{1}{2}(1 - \xi - \eta)(1 - \mu) \quad (3.78a)$$

$$N_2^{(e)} = \frac{1}{2}\xi(1 - \mu) \quad (3.78b)$$

$$N_3^{(e)} = \frac{1}{2}\eta(1 - \mu) \quad (3.78c)$$

$$N_4^{(e)} = \frac{1}{2}(1 - \xi - \eta)(1 + \mu) \quad (3.78d)$$

$$N_5^{(e)} = \frac{1}{2}\xi(1 + \mu) \quad (3.78e)$$

$$N_6^{(e)} = \frac{1}{2}\eta(1 + \mu) \quad (3.78f)$$

Se observa que al reemplazar las coordenadas naturales de un nudo en su función de forma correspondiente, el resultado es 1, lo cual es una de las propiedades de las funciones de forma (Zienkiewicz, 2013).

3.7.4. Matriz de operadores diferenciales con respecto al sistema coordenado natural actuando sobre funciones de forma

Las derivadas parciales de las funciones de forma con respecto a las coordenadas cartesianas se obtienen mediante la regla de la cadena, de la forma mostrada en la [Ecuación 3.31a](#). Al reescribir estas ecuaciones de forma matricial, se definen las matrices $\mathbf{H}^{(e)}$, $\mathbf{J}^{-1(e)}$ y $\hat{\mathbf{B}}^{(e)}$, como se muestra en la [Ecuación 3.32](#). Las matrices $\mathbf{H}^{(e)}$ y $\mathbf{J}^{-1(e)}$ se muestran en la [Subsección 3.7.5](#).

La matriz $\hat{\mathbf{B}}^{(e)}$ es la matriz de operadores diferenciales actuando sobre las funciones de forma en sistema coordenado natural para el elemento prisma, tiene tamaño 3×6 y esta conformada por las submatrices columna $\hat{\mathbf{B}}_{(i)}^{(e)}$ de cada nudo, de la forma:

$$\hat{\mathbf{B}}^{(e)} = \begin{bmatrix} \hat{\mathbf{B}}_{(1)}^{(e)} & \hat{\mathbf{B}}_{(2)}^{(e)} & \cdots & \hat{\mathbf{B}}_{(6)}^{(e)} \end{bmatrix} \quad (3.79)$$

Al reescribir las anteriores ecuaciones en forma matricial, se tiene que:

$$\hat{\mathbf{B}}^{(e)}(\xi, \eta, \mu) = \begin{bmatrix} \frac{\partial N_1(\xi, \eta, \mu)}{\partial \xi} & \cdots & \frac{\partial N_6(\xi, \eta, \mu)}{\partial \xi} \\ \frac{\partial N_1(\xi, \eta, \mu)}{\partial \eta} & \cdots & \frac{\partial N_6(\xi, \eta, \mu)}{\partial \eta} \\ \frac{\partial N_1(\xi, \eta, \mu)}{\partial \mu} & \cdots & \frac{\partial N_6(\xi, \eta, \mu)}{\partial \mu} \end{bmatrix}_{3 \times 6} \quad (3.80)$$

Estas derivadas parciales de las funciones de forma corresponden a:

$$\frac{\partial N_1^e}{\partial \xi} = \frac{\mu}{2} - \frac{1}{2}, \quad \frac{\partial N_2^e}{\partial \xi} = \frac{1}{2} - \frac{\mu}{2}, \quad \frac{\partial N_3^e}{\partial \xi} = 0 \quad (3.81a)$$

$$\frac{\partial N_4^e}{\partial \xi} = -\frac{\mu}{2} - \frac{1}{2}, \quad \frac{\partial N_5^e}{\partial \xi} = \frac{\mu}{2} + \frac{1}{2}, \quad \frac{\partial N_6^e}{\partial \xi} = 0 \quad (3.81b)$$

$$\frac{\partial N_1^e}{\partial \eta} = \frac{\mu}{2} - \frac{1}{2}, \quad \frac{\partial N_2^e}{\partial \eta} = 0, \quad \frac{\partial N_3^e}{\partial \eta} = \frac{1}{2} - \frac{\mu}{2} \quad (3.81c)$$

$$\frac{\partial N_4^e}{\partial \eta} = -\frac{\mu}{2} - \frac{1}{2}, \quad \frac{\partial N_5^e}{\partial \eta} = 0, \quad \frac{\partial N_6^e}{\partial \eta} = \frac{\mu}{2} + \frac{1}{2} \quad (3.81d)$$

$$\frac{\partial N_1^e}{\partial \mu} = \frac{\xi}{2} + \frac{\eta}{2} - \frac{1}{2}, \quad \frac{\partial N_2^e}{\partial \mu} = -\frac{\xi}{2}, \quad \frac{\partial N_3^e}{\partial \mu} = -\frac{\eta}{2} \quad (3.81e)$$

$$\frac{\partial N_4^e}{\partial \mu} = \frac{1}{2} - \frac{\eta}{2} - \frac{\xi}{2}, \quad \frac{\partial N_5^e}{\partial \mu} = \frac{\xi}{2}, \quad \frac{\partial N_6^e}{\partial \mu} = \frac{\eta}{2} \quad (3.81f)$$

Por lo tanto la matriz $\hat{\mathbf{B}}^{(e)}$ para el elemento prisma es:

$$\hat{\mathbf{B}}^{(e)} = \begin{bmatrix} \frac{\mu}{2} - \frac{1}{2} & \frac{1}{2} - \frac{\mu}{2} & 0 & -\frac{\mu}{2} - \frac{1}{2} & \frac{\mu}{2} + \frac{1}{2} & 0 \\ \frac{\mu}{2} - \frac{1}{2} & 0 & \frac{1}{2} - \frac{\mu}{2} & -\frac{\mu}{2} - \frac{1}{2} & 0 & \frac{\mu}{2} + \frac{1}{2} \\ \frac{\xi}{2} + \frac{\eta}{2} - \frac{1}{2} & -\frac{\xi}{2} & -\frac{\eta}{2} & \frac{1}{2} - \frac{\eta}{2} - \frac{\xi}{2} & \frac{\xi}{2} & \frac{\eta}{2} \end{bmatrix} \quad (3.82)$$

3.7.5. Matriz Jacobiana y su inversa

La matriz Jacobiana se calcula como el producto entre la matriz $\hat{\mathbf{B}}^{(e)}$ y la matriz $\mathbf{c}^{(e)}$ como lo indica la [Ecuación 3.36](#). Esta última corresponde a la matriz de coordenadas en los nudos de un elemento prisma y está definida como:

$$\mathbf{c}^{(e)} = \begin{bmatrix} x_1^{(e)} & y_1^{(e)} & z_1^{(e)} \\ x_2^{(e)} & y_2^{(e)} & z_2^{(e)} \\ \vdots & \vdots & \vdots \\ x_6^{(e)} & y_6^{(e)} & z_6^{(e)} \end{bmatrix}_{6 \times 3} \quad (3.83)$$

donde x_i , y_i y z_i definen la posición de cada nudo i del elemento en coordenadas globales.

A partir de la [Ecuación 3.40](#), se observa que la matriz Jacobiana tiene la misma dimensión (3×3) tanto en el elemento hexaédrico como en el prisma, ya que su dimensión no depende del número de nudos del elemento. Por lo tanto, se aplica el mismo procedimiento para calcular su inversa presentado en la [Subsección 3.6.5](#).

La matriz $\mathbf{H}^{(e)}$ se obtiene utilizando la matriz $\hat{\mathbf{B}}^{(e)}$ y el inverso del Jacobiano, como lo indica la [Ecuación 3.32](#). El resultado corresponde a una matriz conformada por submatrices $\mathbf{H}_{(i)}^{(e)}$ asociadas a cada nudo así:

$$\mathbf{H}^{(e)} = \begin{bmatrix} \mathbf{H}_{(1)}^{(e)} & \mathbf{H}_{(2)}^{(e)} & \cdots & \mathbf{H}_{(6)}^{(e)} \end{bmatrix} \quad (3.84)$$

3.7.6. Matriz de operadores diferenciales actuando sobre las funciones de forma

La matriz de operadores diferenciales con respecto a las coordenadas globales actuando sobre las funciones de forma de un elemento prisma $\mathbf{B}^{(e)}$ es:

$$\mathbf{B}^{(e)} = \begin{bmatrix} \mathbf{B}_{(1)}^{(e)} & \mathbf{B}_{(2)}^{(e)} & \cdots & \mathbf{B}_{(6)}^{(e)} \end{bmatrix} \quad (3.85a)$$

la cual está conformada por submatrices $\mathbf{B}_{(i)}^{(e)}$ asociadas a cada nudo i y definidas como:

$$\mathbf{B}_{(i)}^{(e)} = \begin{bmatrix} B_{11}^{(i)} & 0 & 0 \\ 0 & B_{22}^{(i)} & 0 \\ 0 & 0 & B_{33}^{(i)} \\ B_{41}^{(i)} & B_{42}^{(i)} & 0 \\ B_{51}^{(i)} & 0 & B_{53}^{(i)} \\ 0 & B_{62}^{(i)} & B_{63}^{(i)} \end{bmatrix}_{6 \times 3} \quad (3.85b)$$

Los términos de la matriz $\mathbf{B}_i^{(e)}$ se obtiene a partir de las componentes de la matriz $\mathbf{H}_i^{(e)}$, como se muestra en la [Ecuación 3.49](#).

De acuerdo a lo anterior, la matriz $\mathbf{B}^{(e)}$ tiene 6 filas y 18 columnas con los siguientes coeficientes:

$$\mathbf{B}^{(e)} = \begin{bmatrix} B_{11}^{(1)} & 0 & 0 & \dots & B_{11}^{(6)} & 0 & 0 \\ 0 & B_{22}^{(1)} & 0 & \dots & 0 & B_{22}^{(6)} & 0 \\ 0 & 0 & B_{33}^{(1)} & \dots & 0 & 0 & B_{33}^{(6)} \\ B_{41}^{(1)} & B_{42}^{(1)} & 0 & \dots & B_{41}^{(6)} & B_{42}^{(6)} & 0 \\ B_{51}^{(1)} & 0 & B_{53}^{(1)} & \dots & B_{51}^{(6)} & 0 & B_{53}^{(6)} \\ 0 & B_{62}^{(1)} & B_{63}^{(1)} & \dots & 0 & B_{62}^{(6)} & B_{63}^{(6)} \end{bmatrix}_{6 \times 18} \quad (3.85c)$$

3.7.7. Matriz de rigidez del elemento finito

El procedimiento para obtener la matriz de rigidez del elemento finito prisma o cuña es similar al del elemento hexaédrico mostrado en la [Subsección 3.6.8](#). La diferencia está en: (i) la matriz de operadores diferenciales actuando sobre funciones de forma dada en la sección anterior y (ii) la ubicación y el coeficiente de ponderación de los puntos de Gauss del proceso de integración numérica.

Particularmente en este trabajo se aproxima la integral de la matriz de rigidez utilizando $ng = 6$ puntos de Gauss G1, G2, ..., G6. La posición en coordenadas naturales (ξ_g, η_g, μ_g) y el coeficiente de ponderación ω_g de dichos puntos se indican en la [Figura 3.8](#) y la [Tabla 3.4](#) (Kohnke, 2009; Zienkiewicz, 2013).

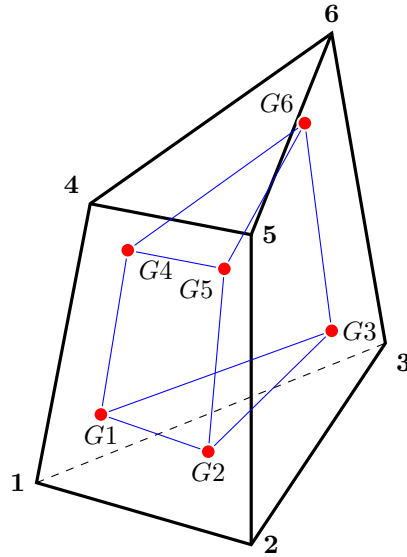


Figura 3.8: Distribución de los 6 puntos de Gauss en el elemento tipo prisma

Tabla 3.4: Puntos de Gauss para elemento prima o cuña y sus coeficientes de ponderación

pg	ξ_g	η_g	μ_g	ω_g
G1	$\frac{1}{6}$	$\frac{1}{6}$	$-\sqrt{\frac{1}{3}}$	$\frac{1}{6}$
G2	$\frac{2}{3}$	$\frac{1}{6}$	$-\sqrt{\frac{1}{3}}$	$\frac{1}{6}$
G3	$\frac{1}{6}$	$\frac{2}{3}$	$-\sqrt{\frac{1}{3}}$	$\frac{1}{6}$
G4	$\frac{1}{6}$	$\frac{1}{6}$	$\sqrt{\frac{1}{3}}$	$\frac{1}{6}$
G5	$\frac{2}{3}$	$\frac{1}{6}$	$\sqrt{\frac{1}{3}}$	$\frac{1}{6}$
G6	$\frac{1}{6}$	$\frac{2}{3}$	$\sqrt{\frac{1}{3}}$	$\frac{1}{6}$

3.7.8. Vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento finito

El vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en el elemento prisma se obtiene de la [Ecuación 3.61](#). Particularmente, el vector de fuerzas de cuerpo $\mathbf{b}^{(e)}$ contiene las componentes de fuerza por unidad de volumen en las direcciones globales, organizadas por subvectores asociados a cada nudo $\mathbf{b}_i^{(e)}$, así:

$$\mathbf{b}^{(e)} = \left[\mathbf{b}_1^{(e)T} \quad \mathbf{b}_2^{(e)T} \quad \dots \quad \mathbf{b}_6^{(e)T} \right]_{1 \times 18}^T \quad (3.86a)$$

siendo,

$$\mathbf{b}_i^{(e)} = [b_{ix} \quad b_{iy} \quad b_{iz}]^T \quad (3.86b)$$

Las fuerzas por unidad de volumen corresponden al peso propio del material en dirección $-y$, con peso específico γ . Por lo tanto, los subvectores por nudo se definen como:

$$\mathbf{b}_i^{(e)} = [0 \quad -\gamma \quad 0]^T \quad (3.87a)$$

de esta manera, el vector $\mathbf{b}^{(e)}$ para el prisma es:

$$\mathbf{b}^{(e)} = [0 \quad -\gamma \quad 0 \quad \dots \quad 0 \quad -\gamma \quad 0]_{1 \times 18}^T \quad (3.87b)$$

3.7.9. Vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global

El vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global sobre todas las caras de un elemento $\tilde{\mathbf{f}}_s^{(e)}$ está definido por la [Ecuación 3.62](#). Asimismo, el vector correspondiente a la presión sobre una cara específica c indicado como $\tilde{\mathbf{f}}_s^{(e)(c)}$ se obtiene de la [Ecuación 3.63](#). Este último está conformado por subvectores $\tilde{\mathbf{f}}_{si}^{(e)(c)}$ en cada nudo i así:

$$\tilde{\mathbf{f}}_s^{(e)(c)} = \begin{bmatrix} \tilde{\mathbf{f}}_{s1}^{(e)(c)} \\ \tilde{\mathbf{f}}_{s2}^{(e)(c)} \\ \vdots \\ \tilde{\mathbf{f}}_{s6}^{(e)(c)} \end{bmatrix} \quad (3.88)$$

A diferencia del elemento hexaédrico, el elemento prisma puede tener caras conformadas por 3 y 4 nudos. Esto significa que para el cálculo de las fuerzas equivalentes

a la acción de una presión se debe tener en cuenta el número de nudos de la cara cargada.

Si la cara cargada del prisma es un cuadrilátero, el subvector de fuerzas equivalentes a la acción de la presión en los nudos que conforman la cara es:

$$\tilde{\mathbf{f}}_{sci}^{(e)} = \frac{A_c^{(e)}}{4} \begin{bmatrix} p_{xc}^{(e)} \\ p_{yc}^{(e)} \\ p_{zc}^{(e)} \end{bmatrix} \begin{cases} i = 1, 2, 3, 4, 5, 6 \\ c = (1254), (1463), (2365) \end{cases} \quad (3.89)$$

Por ejemplo, si se considera una presión uniforme en coordenadas globales aplicada a la cara (1254), los coeficientes que corresponden a los nudos que no pertenecen a la cara cargada son iguales a cero. El vector $\tilde{\mathbf{f}}_{si}^{(e)(1254)}$ es:

$$\tilde{\mathbf{f}}_s^{(e)(1254)} = \frac{A_{1254}^{(e)}}{4} \left[\mathbf{p}_{1254}^{(e)} \quad \mathbf{p}_{1254}^{(e)} \quad \mathbf{O} \quad \mathbf{p}_{1254}^{(e)} \quad \mathbf{p}_{1254}^{(e)} \quad \mathbf{O} \right]_{1 \times 18}^T$$

donde

$$\mathbf{p}_{1254}^{(e)} = \begin{bmatrix} p_{x1254}^{(e)} & p_{y1254}^{(e)} & p_{z1254}^{(e)} \end{bmatrix}, \quad \mathbf{O} = [0 \quad 0 \quad 0]$$

Cuando la cara cargada del prisma es triangular, el subvector de fuerzas equivalentes a la acción de la presión en coordenadas globales $\tilde{\mathbf{f}}_{sci}^{(e)}$, sobre los nudos que conforman la cara es:

$$\tilde{\mathbf{f}}_{sci}^{(e)} = \frac{A_c^{(e)}}{3} \begin{bmatrix} p_{xc}^{(e)} \\ p_{yc}^{(e)} \\ p_{zc}^{(e)} \end{bmatrix} \begin{cases} i = 1, 2, 3, 4, 5, 6 \\ c = (132), (456) \end{cases} \quad (3.90)$$

Por ejemplo si la cara cargada es la (132), el vector $\tilde{\mathbf{f}}_{si}^{(e)(132)}$ corresponde a:

$$\tilde{\mathbf{f}}_s^{(e)(132)} = \frac{A_{132}^{(e)}}{3} \left[\mathbf{p}_{132}^{(e)} \quad \mathbf{p}_{132}^{(e)} \quad \mathbf{p}_{132}^{(e)} \quad \mathbf{O} \quad \mathbf{O} \quad \mathbf{O} \right]_{1 \times 18}^T$$

donde

$$\mathbf{p}_{132}^{(e)} = \begin{bmatrix} p_{x132}^{(e)} & p_{y132}^{(e)} & p_{z132}^{(e)} \end{bmatrix}, \quad \mathbf{O} = [0 \quad 0 \quad 0]$$

3.7.10. Vector de fuerzas equivalentes a la acción de una presión uniforme normal a una cara del elemento

El vector normal unitario a una cara cuadrilateral del elemento prisma se define siguiendo el procedimiento indicado en la [Subsección 3.6.11](#), que fue utilizado en las caras de elementos hexaédricos. Asimismo, el vector normal unitario a una cara triangular se obtiene de la misma manera que en elementos tetraédricos, como se indica en algunas referencias (Linero *et al.*, 2020). Este procedimiento se resume en calcular el vector normal a la superficie mediante el producto cruz entre dos de sus lados, utilizar un nudo externo para definir la dirección entrando al elemento y normalizar el vector. Por ejemplo, la [Figura 3.9\(a\)](#) muestra el vector normal unitario a la cara (456).

Si la cara cargada tiene 4 nudos, cada componente de fuerza en dirección normal en el nudo es igual a:

$$\tilde{f}_{ni}^{(e)(c)} = \frac{1}{4} p_{nc}^{(e)} A_{nc}^{(e)} \quad (3.91)$$

En cambio, si la cara cargada tiene 3 nudos, cada componente de fuerza en dirección normal en el nudo es:

$$\tilde{f}_{ni}^{(e)(c)} = \frac{1}{3} p_{nc}^{(e)} A_{nc}^{(e)} \quad (3.92)$$

Por ejemplo, si se aplica una presión normal $p_{n(456)}^{(e)}$ a la cara triangular (456), las fuerzas equivalentes en los nudos serán $\tilde{\mathbf{f}}_{n4}^{(e)(456)}$, $\tilde{\mathbf{f}}_{n5}^{(e)(456)}$ y $\tilde{\mathbf{f}}_{n6}^{(e)(456)}$, como se ilustra en las figuras [3.9\(b\)](#) y [\(c\)](#), respectivamente.

El vector que contiene las componentes de las de fuerzas equivalentes completo $\tilde{\mathbf{f}}_s^{(e)(456)}$ es:

$$\tilde{\mathbf{f}}_s^{(e)(456)} = \frac{1}{3} p_{n456}^{(e)} A_{456}^{(e)} \left[\mathbf{O} \quad \mathbf{O} \quad \mathbf{O} \quad \mathbf{n}_{456}^{(e)} \quad \mathbf{n}_{456}^{(e)} \quad \mathbf{n}_{456}^{(e)} \right]_{1 \times 18}^T$$

donde

$$\mathbf{n}_{456}^{(e)} = \left[n_{x456}^{(e)} \quad n_{y456}^{(e)} \quad n_{z456}^{(e)} \right], \quad \mathbf{O} = \left[0 \quad 0 \quad 0 \right]$$

Las componentes del vector de fuerzas equivalentes a la acción de una presión normal sobre una cara, obtenida en los nudos que la conforman, se pueden expresar en el sistema coordenado global mediante la [Ecuación 3.70](#). En el ejemplo anterior la [3.9\(d\)](#) muestra las componentes de la fuerza equivalente en la cara (456).

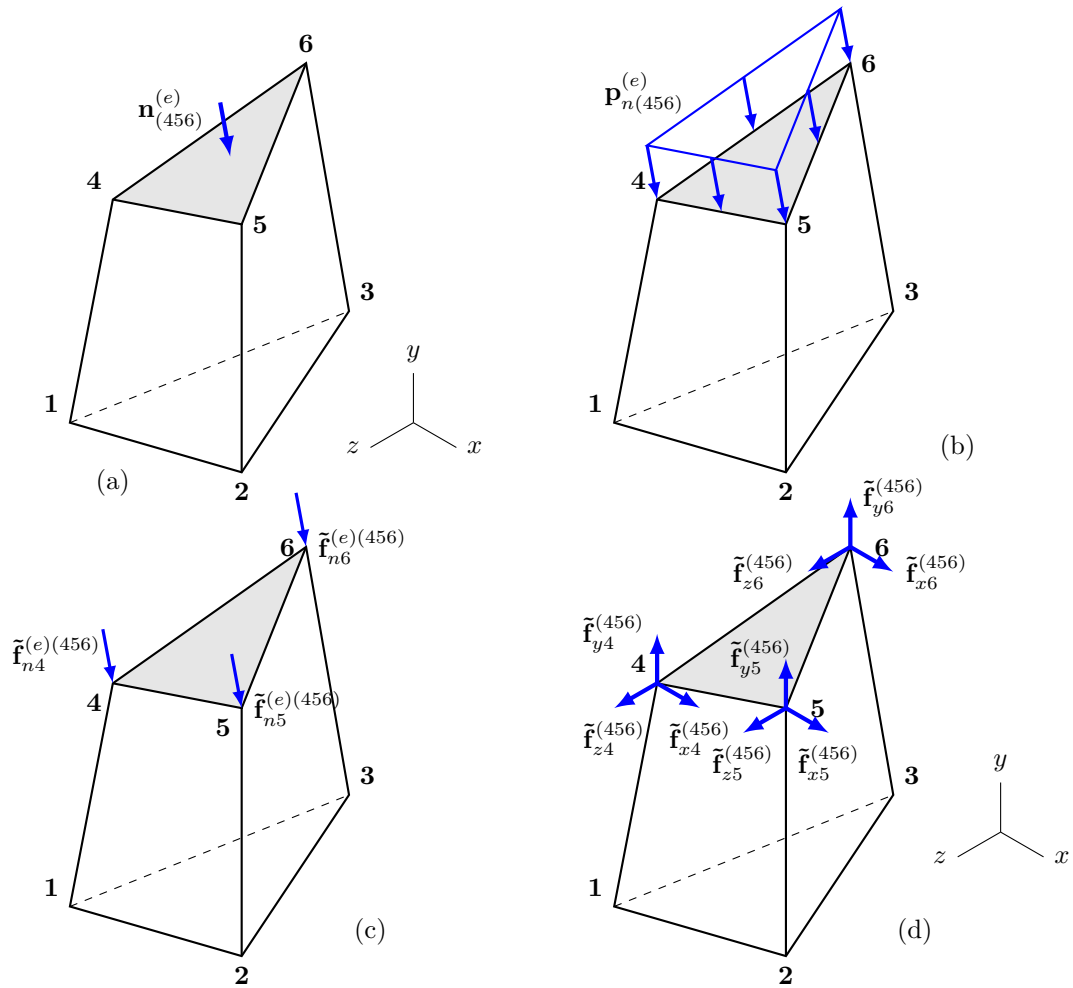


Figura 3.9: Presión uniforme normal que entra a la cara 4-5-6 del prisma: (a) vector normal entrando a la cara, (b) presión uniforme normal que entra a la cara, (c) fuerzas equivalentes en dirección normal, y (d) componentes de las fuerzas equivalentes en sistema coordenado global.

3.7.11. Vector de fuerzas equivalentes a la acción de una presión hidroestática

El procedimiento para calcular la presión hidroestática sobre las caras de los elementos prisma es el mismo que fue aplicado a los elementos hexaédricos en la [Subsección 3.6.12](#). Esta presión es uniforme en la cara del elemento finito y es función de la profundidad dada por la coordenada $y_c^{(e)}$ del centroide de dicha cara.

Las fuerzas equivalentes a la acción de una presión normal producida por el agua, se obtiene de la misma forma como se describió en la [Subsección 3.7.10](#).

Capítulo 4

Implementación del programa de computador

En este capítulo se muestra la implementación del método de los elementos finitos para la solución de un problema mecánico elástico lineal tridimensional, utilizando el elemento hexaedro de 8 nudos y el prisma de 6 nudos.

La implementación del método de los elementos finitos para los elementos tratados en este trabajo final de maestría, tiene como punto de partida el programa PEFiCA 2.0 (Linero *et al.*, 2020), realizado en lenguaje de alto nivel GNU-OCTAVE (Eaton *et al.*, 2017) y acompañado del pre y post proceso gráfico con el programa GMSH (Geuzaine & Remacle, 2019).

A continuación se muestra la estructura general del programa, así como los procedimientos y las rutinas nuevas que se implementan en el programa PEFiCA 2.0, junto con las modificaciones a las rutinas existentes, que amplían la capacidad y el alcance del programa original.

4.1. Descripción de la estructura general del programa desarrollado a partir de PEFiCA 2.0

Tal como se mencionó en la [Subsección 2.2.3](#), este trabajo utiliza la estructura principal y las rutinas del programa PEFiCA 2.0 para dar la solución al problema elástico tridimensional mediante el método de elementos finitos.

Todas las rutinas de PEFiCA 2.0 está hecha en programación estructurada, por lo que no hay saltos arbitrarios en la secuencia de ejecución (Molina, 2006). De la misma manera, las rutinas programadas para la solución del problema de la mecánica estructural mediante el método de los elementos finitos con elementos hexaedro y prisma mantienen esos mismos lineamientos.

Para dar solución a un problema elástico tridimensional el proceso general del programa desarrollado a partir de PEFiCA 2.0 (Linero *et al.*, 2020) es el siguiente:

- (a) Utilizando el entorno gráfico del programa GMSH se define la geometría del sólido, las propiedades de los materiales, los desplazamientos iniciales y las fuerzas o presiones actuantes. También se especifican los parámetros del enmallado y se genera la malla tridimensional de elementos. Esta información se exporta a un archivo con extensión `.msh`.
- (b) El archivo con extensión `.msh` es leído por PEFiCA 2.0 mediante la función `LEGMSH`. Allí se extraen las tablas y parámetros que se utilizan para la solución del problema. Algunas de estas tablas son las siguientes:
 - Coordenadas de los nudos
 - Conectividades y categorías de los elementos
 - Características de las categorías de los elementos
 - Desplazamientos conocidos
 - Fuerzas aplicadas en los nudos de la malla
 - Fuerzas uniformes distribuidas o presiones aplicadas en las caras de los elementos

También se extraen los siguientes valores escalares:

- Número de nudos
 - Número de elementos
 - Número máximo de nudos por elemento
 - Número máximo de puntos de Gauss en un elemento
 - Número de categorías de elementos
- (c) A partir de la tabla de desplazamientos conocidos de algunos nudos, se crea la tabla de grados de libertad por nudo, el subvector de desplazamientos conocidos, y obtener el número de grados de libertad por nudo y el número de grados de libertad conocidos.
 - (d) Se define el tamaño de la matriz de rigidez de la malla. A continuación se genera un ciclo por el número de elementos, donde se calcula la matriz de rigidez del elemento finito y se ensambla en la matriz de rigidez de la malla. La librería de

elementos finitos cuenta con el tetraedro de 4 nudos, el hexaedro de 8 nudos y el prisma de 6 nudos. La implementación de las matrices de rigidez para los dos últimos elementos se describen en la [Sección 4.5](#).

- (e) A partir de la matriz de rigidez de la malla ensamblada \mathbf{K} , se generan las submatrices $\mathbf{K}_{\alpha\alpha}$, $\mathbf{K}_{\alpha\beta}$, $\mathbf{K}_{\beta\alpha}$ y $\mathbf{K}_{\beta\beta}$, de acuerdo al número de grados de libertad asociados a los desplazamientos desconocidos del problema.
- (f) Se genera el vector de fuerzas equivalentes a las fuerzas de cuerpo en la malla mediante un ciclo por elemento, en el cual se ensambla el vector de fuerzas equivalentes a las fuerzas de cuerpo cada elemento de acuerdo a sus grados de libertad. Estos últimos se describen en la [Subsección 3.6.9](#) para el elemento hexaédrico y en la [Subsección 3.7.8](#) para el prisma.
- (g) Se define el vector de fuerzas equivalentes a fuerzas de superficie en la malla utilizando principalmente la tabla de fuerzas uniformes distribuidas aplicadas en las caras de los elementos, la tabla de incidencias y las coordenadas de los nudos. El cálculo de las fuerzas equivalentes a las presiones se muestra en las subsecciones [3.6.10](#), [3.6.11](#) y [3.6.12](#) para el caso del hexaedro, y en las subsecciones [3.7.9](#), [3.7.9](#) y [3.7.11](#) para el caso del prisma. Posteriormente ensamblar en el vector de fuerzas equivalentes a las fuerzas de superficie de la malla.
- (h) Se establece el vector de fuerzas aplicadas directamente en los nudos de la malla, utilizando la tabla fuerzas aplicadas en los nudos.
- (i) Se obtiene el vector de fuerzas totales equivalentes en los nudos de la malla a partir de la suma de los vectores anteriores, como se muestra en la [Ecuación 3.6](#).
- (j) Se calcula el vector de los desplazamientos desconocidos en los nudos de la malla mediante la [Ecuación 3.12](#) y a partir de la numeración de los grados de libertad, se construye una tabla de desplazamientos ordenados por nudo.
- (k) Se evalúa el vector de las reacciones en los nudos de la malla mediante la [Ecuación 3.12](#) y a partir de la numeración de los grados de libertad, se construye una tabla de reacciones ordenadas por nudo.
- (l) Se obtienen las deformaciones y los esfuerzos en los vértices de cada elemento finito. Para ello se realiza un ciclo por elemento anidado con otro ciclo por coordenada del vértice, donde se calcula lo siguiente:
 - la matriz de operadores diferenciales actuando sobre funciones de forma del elemento tal como se indica en la [Subsección 3.6.6](#) y en la [Subsección 3.7.6](#),

- el vector de desplazamientos nodales del elemento a partir del proceso de extracción como lo indica la [Ecuación 3.16](#),
 - el vector de las componentes de deformación como se indica en [Ecuación 3.17](#),
 - la matriz constitutiva del material de acuerdo con la [Subsección 3.6.7](#),
 - el vector de las componentes de esfuerzo como lo muestra la [Ecuación 3.19](#),
 - las deformaciones principales, los esfuerzos principales y el esfuerzo equivalente de Von Mises, tal como se muestra en la [Sección 3.3](#) y en la [Sección 3.4](#).
- (m) Mediante la función `IMGMSH`, generar el archivo de extensión `.pos` en formato de postproceso de GMSH, el cual contiene la información de desplazamientos nodales y las reacciones en los nudos. Asimismo, incluye los esfuerzos y las deformaciones para cada categoría y elemento finito.
- (n) Para el análisis de los resultados, se carga el archivo de postproceso en el entorno gráfico de GMSH.

En la [Figura 4.1](#) se muestra un resumen de las etapas principales del procedimiento general del programa implementado a partir de la estructura de PEFiCA 2.0.

4.2. Alcance del programa implementado

En los ejemplos de aplicación se discretizó el sólido con mallas estructuradas, conformadas por elementos finitos hexaédrico de 8 nudos, prisma de 6 nudos o tetraédrico de 4 nudos. Tales mallas se obtuvieron utilizando las funciones propias del entorno de preproceso de GMSH, descritas en el manual de usuario presentado en la [Sección 7.1](#).

Esta malla de elementos puede representar varios materiales en diferentes partes del sólido. Asimismo, para determinadas geometrías, como por ejemplo las cilíndricas, el programa GMSH asigna automáticamente combinaciones de elementos hexaédricos y prismas. El alcance de este trabajo incluye estas combinaciones, como también las mallas en las que sólo hay un tipo de elemento, bien sea tetraedro, hexaedro o prisma, para uno o varios materiales.

En la etapa de análisis, las rutinas originales de PEFiCA 2.0 y las modificaciones generadas por este trabajo, tienen su capacidad limitada por: el tamaño máximo admisible de una matriz cuadrada en el lenguaje de GNU–Octave, el sistema operativo y las características del computador. Por ejemplo, utilizando un computador

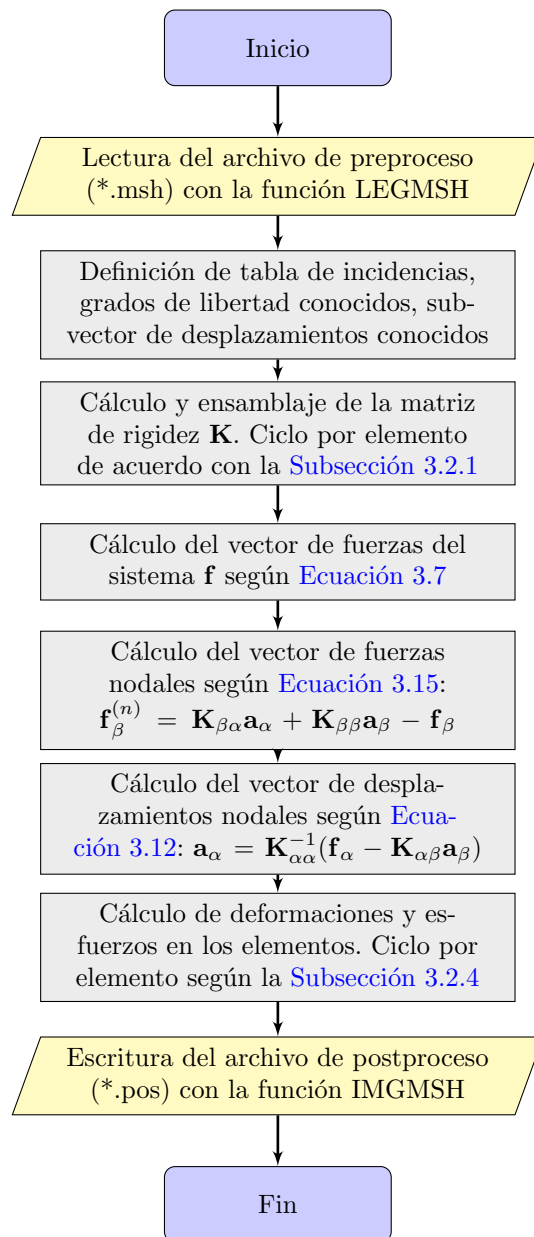


Figura 4.1: Diagrama de flujo resumido de las etapas principales del procedimiento general del programa desarrollado a partir de PEFiCA 2.0, para dar solución a problemas elásticos tridimensionales.

con un procesador Intel Core I7-6700HQ y 12Gb de memoria RAM, y ejecutando el programa con GNU-Octave 4.4.1 en un sistema operativo Windows 7, se pueden analizar problemas con mallas hasta de 9 500 nudos aproximadamente.

Cabe mencionar nuevamente que las rutinas desarrolladas en este trabajo son compatibles con el programa MATLAB. En la simulación de los ejemplos de aplicación, ejecutando con MATLAB el programa desarrollado, se observó que el tiempo de cálculo es del orden de 5 veces menos que en GNU-Octave y que la capacidad es aproximadamente la misma.

4.3. Construcción del archivo de datos de entrada mediante el entorno gráfico de GMSH

El entorno gráfico de preproceso del programa GMSH permite construir un archivo de datos de entrada del problema, el cual incluye la geometría, las condiciones de borde, las acciones impuestas y las propiedades mecánicas de la malla de elementos finitos.

Inicialmente se definen las entidades geométricas del problema que determinan su geometría. A continuación se establecen las entidades físicas que representan las características de los materiales, las cargas y los desplazamientos conocidos. Asimismo, se designan los parámetros de la malla. Finalmente, el programa GMSH genera una malla conformada por nudos y elementos finitos según los parámetros seleccionados.

Esta malla de elementos la exporta GMSH a un archivo de texto con extensión `.msh`, el cual contiene la información de los materiales, las coordenadas de los nudos, los tipos de elemento y las demás condiciones del problema.

El procedimiento para construir el archivo de datos de entrada de un problema, utilizando el entorno gráfico de GMSH se describe en el manual de usuario presentado en la [Sección 7.1](#).

4.4. Lectura del archivo de datos de entrada generado en el preproceso

La rutina LEGMSH escrita en GNU-Octave en el programa PEFiCA 2.0 permite leer e interpretar el archivo de datos de entrada, el cual fue generado durante el preproceso con el programa GMSH (con extensión `.msh`). Asimismo, esta rutina genera las tablas, las matrices y los vectores requeridos en el análisis. El programa aquí desarrollado amplía el alcance de ésta rutina, permitiendo la solución de problemas

mecánicos tridimensionales con elementos finitos hexaédricos y prismas. A continuación se describen los procedimientos adicionados a dicha rutina.

La instrucción que llama a esta función tiene la siguiente estructura:

$$[\langle nnud \rangle, \langle nele \rangle, \langle nnue \rangle, \langle ngau \rangle, \langle ndim \rangle, \langle ncat \rangle, \langle tipr \rangle, \langle ennu \rangle, \langle impr \rangle, \dots \\ \dots \langle xyz \rangle, \langle ele \rangle, \langle cat \rangle, \langle uco \rangle, \langle fun \rangle, \langle fdi \rangle, \langle sup \rangle, \langle nct \rangle] = \text{LEGMSH}(\langle adad \rangle);$$

El parámetro de entrada de la función es $\langle adad \rangle$, el cual para este trabajo es el archivo que contiene la información de la malla de elementos finitos.

Los parámetros de salida de la función son:

- $\langle nnud \rangle$: número de nudos,
- $\langle nele \rangle$: número de elementos,
- $\langle nnue \rangle$: número máximo de nudos por elemento,
- $\langle ngau \rangle$: número máximo de puntos de Gauss en un elemento,
- $\langle ndim \rangle$: número de dimensiones del problema,
- $\langle ncat \rangle$: número de categorías de los elementos,
- $\langle tipr \rangle$: código del tipo de problema,
- $\langle ennu \rangle$: tipo de evaluación de esfuerzos o deformaciones,
- $\langle impr \rangle$: tipo de impresión de los resultados,
- $\langle xyz \rangle$: tabla de coordenadas de los nudos,
- $\langle ele \rangle$: tabla de categorías y conectividades de los elementos,
- $\langle cat \rangle$: tabla de categorías de los elementos,
- $\langle uco \rangle$: tabla de desplazamientos conocidos,
- $\langle fun \rangle$: tabla de fuerzas aplicadas en los nudos,
- $\langle fdi \rangle$: tabla de fuerzas uniformes distribuidas que se aplican en las caras de los elementos,
- $\langle sup \rangle$: tabla auxiliar de identificador de la superficie asociada al elemento finito,

- $\langle nct \rangle$: tabla auxiliar para impresión de archivo de posproceso.

En este trabajo se modificaron algunas características de la función LEGMSH para incluir la compatibilidad con los elementos hexaédrico y prisma, las cuales se describen a continuación:

- Tipos de elementos finitos: el número de tipos de elementos reconocidos por este procedimiento y sus códigos asociados se ampliaron, permitiendo identificar y utilizar el elemento hexaédrico de ocho nudos y el prisma de seis nudos.
- Creación de categorías: los elementos finitos definidos en el archivo generado por GMSH del problema se agrupaban en categorías de acuerdo solamente con el material al cual estaban asociados. En este trabajo se amplió este criterio para que se generen nuevas categorías, de acuerdo también al tipo de elemento ya sea tetraedro, hexaedro o prisma, considerando además sus posibles combinaciones.
- Lectura de condiciones de borde en superficies cuadrilaterales: considerando que tanto el elemento hexaédrico como el elemento prisma pueden tener desplazamientos conocidos y presiones sobre sus caras cuadrilaterales, se incluyó la opción de interpretar este tipo de acciones.
- Tabla auxiliar $\langle nct \rangle$: se creó para facilitar la ubicación de los elementos por categoría dentro de la matriz que almacena los resultados. La sintaxis de cada fila de dicha tabla es la siguiente:

$$[\langle nelc \rangle, \langle nnue \rangle, \langle mine \rangle, \langle maxe \rangle]$$

El parámetro $\langle nelc \rangle$ es el número de elementos de la categoría a la que le corresponde la fila, $\langle nnue \rangle$ es el número de nudos del elemento de la categoría a la que le corresponde la fila, $\langle mine \rangle$ es la mínima posición del primer elemento de la categoría en la matriz ELE y $\langle maxe \rangle$ es la máxima posición del primer elemento de la categoría en la matriz ELE. El número filas de esta tabla corresponde a la cantidad de categorías del problema.

4.5. Rutinas modificadas de PEFiCA 2.0 en la etapa de proceso

El programa PEFiCA 2.0 permite el análisis de problemas mecánicos tridimensionales con elementos tetraédricos. Sus funciones han sido presentadas en el Capítulo 9 de la referencia (Linero *et al.*, 2020).

En este trabajo final de maestría se modificaron algunas funciones del programa PEFiCA 2.0 y se crearon nuevas funciones, con el fin de implementar los elementos finitos hexaédrico y prisma en la solución del problema elástico tridimensional.

A continuación se describen brevemente las funciones de PEFiCA 2.0 que fueron ampliadas y se detallan los cambios realizados en cada una de ellas:

4.5.1. Función PEFiCA

Este es el procedimiento principal del programa, en el cual se encuentra la estructura general de la solución al problema elástico lineal tridimensional. A continuación se muestran los cambios que se realizaron en él con el objetivo de ampliar el alcance del mismo para utilizar los elementos finitos hexaédrico y prisma:

- Uso de la función `linsolve`: luego de realizar la partición según el número de grados de libertad conocidos y desconocidos de la matriz de rigidez del sólido \mathbf{K} , la cual genera las submatrices $\mathbf{K}_{\alpha\alpha}$, $\mathbf{K}_{\alpha\beta}$, $\mathbf{K}_{\beta\alpha}$ y $\mathbf{K}_{\beta\beta}$, y del vector de fuerzas nodales \mathbf{f} , el cual genera los vectores \mathbf{f}_α y \mathbf{f}_β , se requiere encontrar la solución al sistema de ecuaciones simultáneas que calcula el vector de desplazamientos desconocidos $\mathbf{a}_\alpha = \mathbf{K}_{\alpha\alpha}^{-1}(\mathbf{f}_\alpha - \mathbf{K}_{\alpha\beta}\mathbf{a}_\beta)$ indicado en la [Ecuación 3.12](#).

Para dar la solución al sistema, se utilizó la función propia `linsolve` del lenguaje de GNU-OCTAVE (compatible con MATLAB), con el fin de reducir el tiempo de ejecución de la solución del sistema de ecuaciones simultáneas. Dado que la submatriz de rigidez $\mathbf{K}_{\alpha\alpha}$ es simétrica, se seleccionó la opción correspondiente de acuerdo con la siguiente sintaxis:

```
opts.SYM = true;
UAA=linsolve(KAA,AUX,opts);
```

siendo `AUX` el vector resultante de $(\mathbf{f}_\alpha - \mathbf{K}_{\alpha\beta}\mathbf{a}_\beta)$. En la [Tabla 5.4](#) se demuestra que el tiempo de ejecución se redujo notoriamente utilizando esta función.

- Implementación del cálculo de la matriz $\mathbf{B}^{(e)}$: en el cálculo de las deformaciones y esfuerzos en cada elemento, se agregó una instrucción de selección del tipo de elemento finito, de tal forma que se incluya a los elementos hexaédrico y prisma, con sus procedimientos asociados al cálculo de la matriz $\mathbf{B}^{(e)}$. Tal matriz se obtiene en las funciones `NELEME`, `JELEME` y `BELEME`, las cuales se mostrarán en las secciones [4.6.2](#), [4.6.1](#) y [4.5.3](#), respectivamente.
- Instrucciones de llamada a las funciones `LEGMSH` y `IMGMSH`: se modificaron las instrucciones de llamada a éstas funciones, de acuerdo con los cambios en sus argumentos mostrados en las secciones [4.4](#) y [4.7](#).

4.5.2. Función KELEME

Esta función calcula la matriz de rigidez de un elemento finito. La instrucción de llamada de la función es:

$$[\langle mtx \rangle] = \text{KELEME}(\langle tipe \rangle, \langle xye \rangle, \langle cae \rangle);$$

en donde $\langle mtx \rangle$ corresponde a la matriz de rigidez del elemento, $\langle tipe \rangle$ es un escalar que indica el tipo de problema, $\langle xye \rangle$ es una matriz que contiene las coordenadas de los nudos del elemento, y $\langle cae \rangle$ corresponde a una matriz fila que contiene las propiedades de la categoría del elemento (Linero *et al.*, 2020).

La ampliación a esta función consiste en crear las subfunciones correspondientes al cálculo de las matrices de rigidez para el elemento hexaedro KHEXAE, y para el elemento prisma KPRISM, como se puede observar en la [Figura 4.2](#).

Las instrucciones de llamada de las subfunciones KHEXAE y KPRISM, las cuales tienen los mismos argumentos de entrada que la función KELEME, son:

$$[\langle mtx \rangle] = \text{KHEXAE}(\langle tipe \rangle, \langle xye \rangle, \langle cae \rangle);$$

$$[\langle mtx \rangle] = \text{KPRISM}(\langle tipe \rangle, \langle xye \rangle, \langle cae \rangle);$$

Estas subfunciones realizan el cálculo de la matriz de rigidez del elemento hexaédrico y prisma aplicando el procedimiento descrito en la sección [Subsección 3.6.8](#) y [Subsección 3.7.7](#) respectivamente. La implementación de éstos procedimientos se describe a continuación:

- (a) Se obtiene la matriz constitutiva del material del elemento finito $\mathbf{D}^{(e)}$ utilizando la función DELEME (Linero *et al.*, 2020).
- (b) Se definen las coordenadas naturales correspondientes a los puntos de Gauss del elemento finito, así como sus valores de ponderación mediante la función PBPGAU.
- (c) Se inicializa la matriz de rigidez del elemento $\mathbf{K}^{(e)}$ con ceros, de tamaño 24×24 en la subfunción KHEXAE y 18×18 en la subfunción KPRISM.
- (d) Para obtener la matriz de rigidez del elemento $\mathbf{K}^{(e)}$, se debe realizar el procedimiento descrito en la sección [Subsección 3.6.8](#) para cada punto de Gauss y luego sumar el aporte de cada matriz de rigidez asociada a cada punto de Gauss en la matriz $\mathbf{K}^{(e)}$. Esto se hace dentro de un ciclo que va desde $g = 1$ hasta el número de puntos de Gauss del elemento ng , siendo $ng = 8$ para el hexaedro y $ng = 6$ para el prisma, donde se realizan los siguientes pasos:

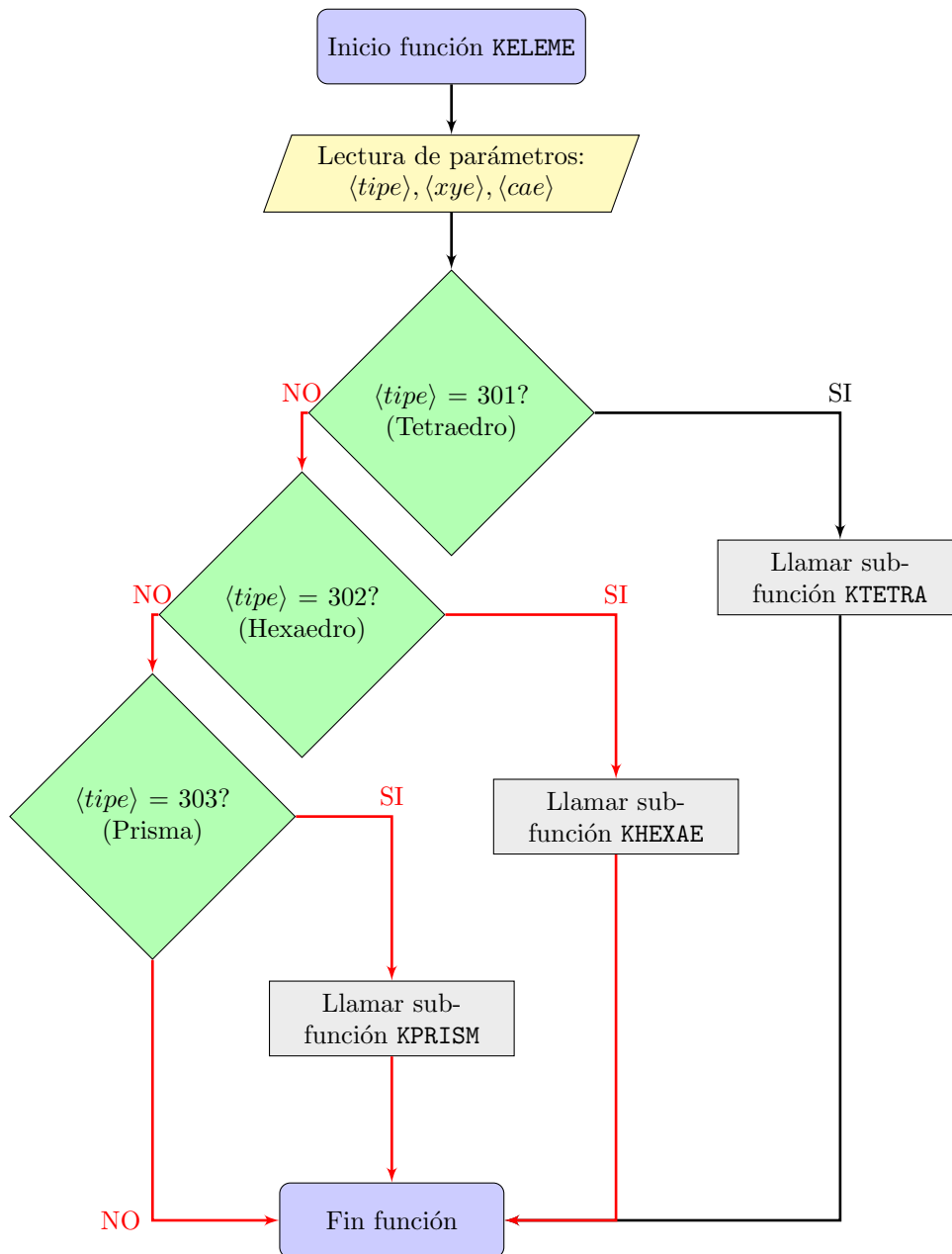


Figura 4.2: Diagrama de flujo de la función KELEME, la cual calcula las matrices de rigidez elementales. Los conectores de color rojo muestran las nuevas etapas implementadas, relacionadas con el cálculo de la matriz de rigidez del elemento hexaédrico ($tipe=302$) y prisma ($tipe=303$).

- Se extrae la coordenada natural (ξ_g, η_g, μ_g) y el parámetro de ponderación ω_g del punto de Gauss g .
- Se calcula la matriz de operadores diferenciales con respecto al sistema natural actuando sobre funciones de forma $\hat{\mathbf{B}}^{(e)}$, utilizando la función **NELEME**, evaluada en el punto de Gauss g .
- Con la matriz $\hat{\mathbf{B}}^{(e)}$ y la matriz de coordenadas de los nudos del elemento, utilizando la función **JELEME** se calcula la matriz inversa del Jacobiano del elemento $\mathbf{J}_{(e)}^{-1}$ y su determinante $\det \mathbf{J}_{(e)}$, evaluados en el punto de Gauss g .
- Con la matriz inversa del Jacobiano $\mathbf{J}_{(e)}^{-1}$ y la matriz $\hat{\mathbf{B}}^{(e)}$, utilizando la función **BELEME** se define la matriz de operadores diferenciales actuando sobre las funciones de forma $\mathbf{B}^{(e)}$.
- Se evalúa el diferencial de volumen dV mediante el producto entre el inverso del Jacobiano $\mathbf{J}_{(e)}^{-1}$ y el valor de ponderación ω_g del punto de Gauss g .
- Se suma el aporte a la matriz de rigidez del elemento $\mathbf{K}^{(e)}$ del punto g , mediante $\mathbf{K}^{(e)} = \mathbf{K}^{(e)} + \mathbf{B}^{T(e)}\mathbf{D}^{(e)}\mathbf{B}^{(e)}dV$

(e) La función retorna la matriz resultante $\mathbf{K}^{(e)}$ del elemento.

4.5.3. Función BELEME

Esta función se encarga del cálculo de la matriz \mathbf{B} . La instrucción que llama a la función es de la forma:

$$[\langle bel \rangle] = \text{BELEME}(\langle xye \rangle, \langle xip \rangle, \langle tipe \rangle);$$

Los parámetros de entrada de esta función, para elementos tetraédricos son: la matriz que contiene las coordenadas de los nudos de un elemento $\langle xye \rangle$, la coordenada del punto $\langle xip \rangle$ donde se evalúa la función de forma dentro del elemento en sistema coordenado natural, y el identificador del tipo de elemento finito $\langle tipe \rangle$.

La ampliación de **BELEME** consiste en identificar el tipo de elemento y calcular la matriz $\mathbf{B}^{(e)}$ de un elemento hexaédrico o prisma. Si el parámetro $\langle tipe \rangle = 302$ calcula la matriz del elemento hexaédrico y si $\langle tipe \rangle = 303$ obtiene la matriz de elemento prisma. Lo anterior se puede ver en la [Figura 4.3](#).

Por otro lado, los parámetros de entrada para los elementos hexaédrico y prisma se modificaron, conservando la misma forma de la instrucción de llamada. Para el elemento hexaédrico y prisma, los parámetros de entrada son:

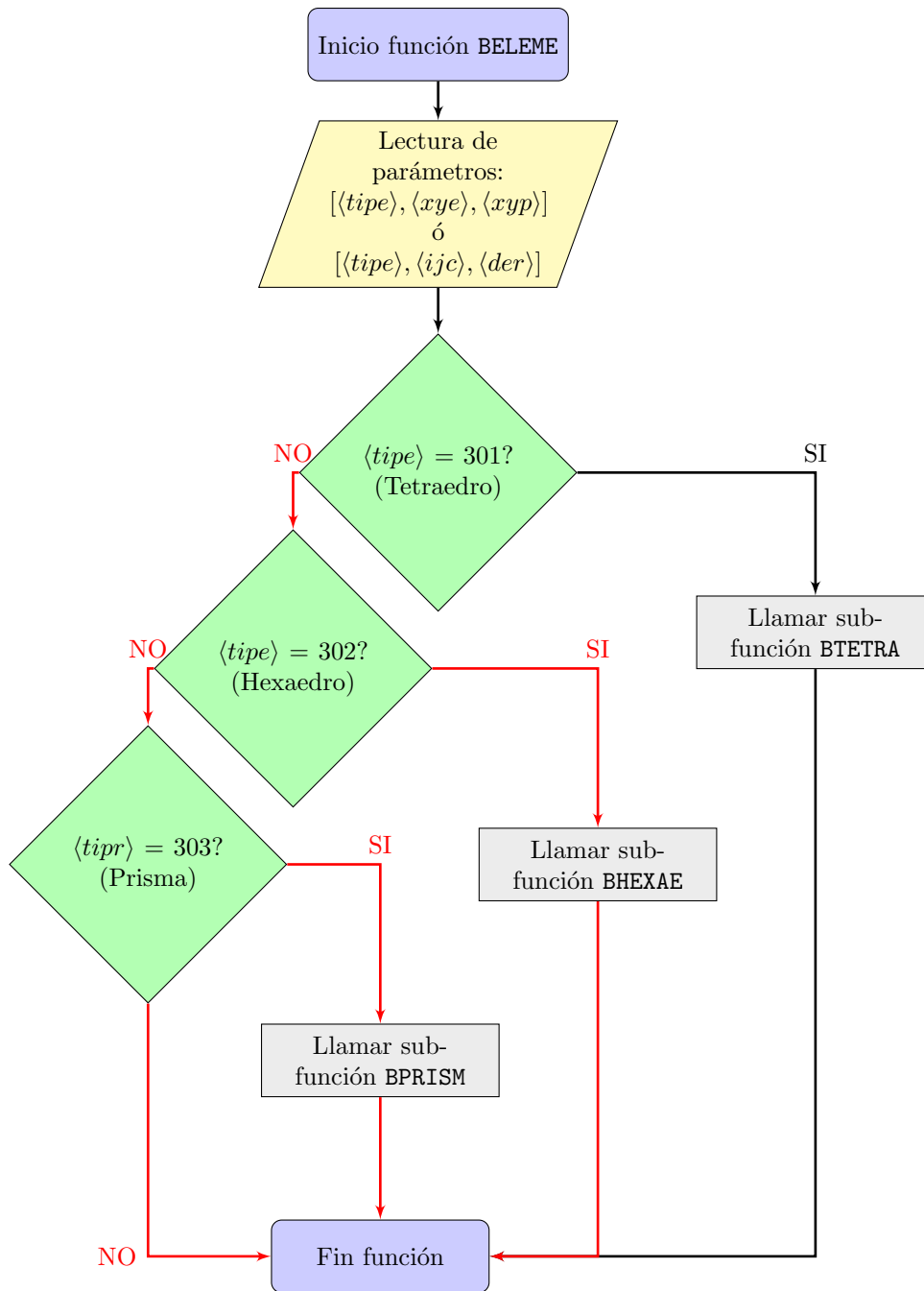


Figura 4.3: Diagrama de flujo de la función BELEME, que calcula la matriz $\mathbf{B}^{(e)}$. Los conectores de color rojo muestran las nuevas etapas implementadas, relacionadas con el cálculo de la matriz $\mathbf{B}^{(e)}$ del elemento hexaédrico ($tipr=302$) y prisma ($tipr=303$).

$$[\langle bel \rangle] = \text{BELEME}(\langle ijc \rangle, \langle der \rangle, \langle tipe \rangle);$$

donde $\langle ijc \rangle$ es la matriz inversa del Jacobiano $\mathbf{J}_g^{(e)}$, evaluada en el punto de Gauss g dentro del elemento en coordenadas naturales, y $\langle der \rangle$ es la matriz de derivadas de las funciones de forma $\tilde{\mathbf{B}}_g^{(e)}$ evaluada en el mismo punto.

Dentro de la función **BELEME**, se encuentra la función **BHEXAE**, que genera la matriz $\mathbf{B}^{(e)}$ del hexaedro según la [Ecuación 3.48c](#), y **BPRISM**, la cual crea la matriz $\mathbf{B}^{(e)}$ del prisma según la [Ecuación 3.85c](#).

4.5.3.1. Función FELEMC

Esta función calcula el vector de fuerzas equivalentes a la acción de fuerzas de cuerpo en un elemento finito. La instrucción que llama a la función es:

$$[\langle fel \rangle] = \text{FELEMC}(\langle xye \rangle, \langle cae \rangle);$$

donde $\langle fel \rangle$ corresponde al vector de fuerzas equivalentes a la acción del peso propio del material sobre el elemento $\tilde{\mathbf{f}}_b^{(e)}$, $\langle xye \rangle$ es la matriz que contiene las coordenadas de los nudos del elemento, y $\langle cae \rangle$ es una matriz fila que contiene las propiedades de la categoría de un elemento.

En este trabajo se implementó el cálculo del vector $\tilde{\mathbf{f}}_b^{(e)}$ para el elemento hexaédrico y el prisma, utilizando las sub funciones **FHEXAC** y **FPRISM** respectivamente, como se ve en la [Figura 4.4](#). La instrucción que llama a la subfunción **FHEXAC** es similar a la que se utiliza para **FPRISM**, y es de la forma:

$$[\langle fel \rangle] = \text{FHEXAC}(\langle nnue \rangle, \langle xye \rangle, \langle cae \rangle);$$

donde los parámetros $\langle fel \rangle$, $\langle xye \rangle$ y $\langle cae \rangle$ son los mismos utilizados para la función **FELEMC**, y se agrega el parámetro $\langle nnue \rangle$, el cual indica el número de nudos del elemento.

Dentro de cada de las subfunciones anteriores, se hace un llamado a la función **PBPGAU** para traer las coordenadas naturales y los parámetros de ponderación de los puntos de integración, de acuerdo con el número de nudos del elemento $\langle nnue \rangle$. Asimismo, se suma el aporte de cada uno de los puntos de Gauss al vector $\tilde{\mathbf{f}}_b^{(e)}$, tal como se indica en la [Ecuación 3.61](#).

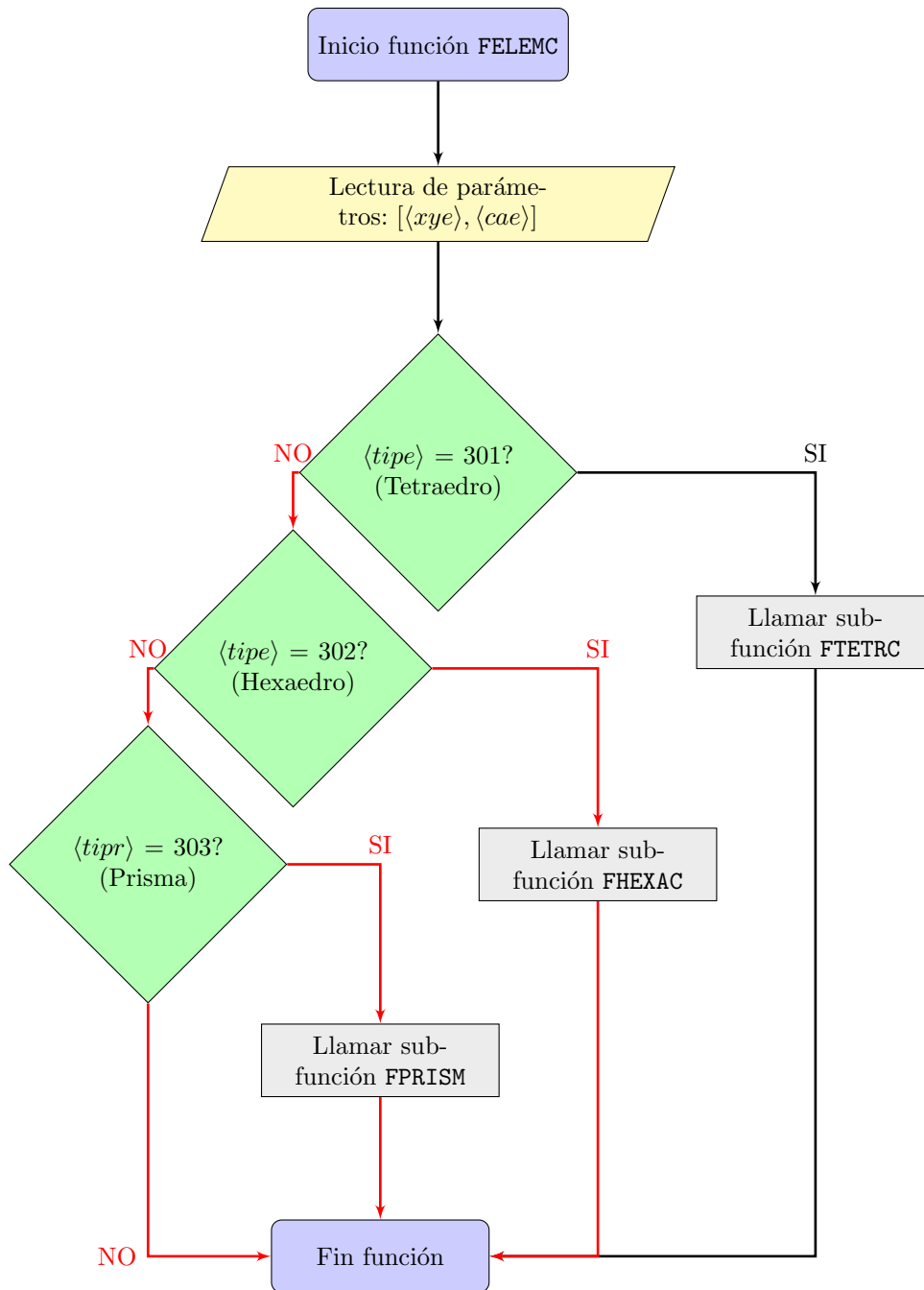


Figura 4.4: Diagrama de flujo de la función FELEMC, que calcula el vector $\mathbf{f}_b^{(e)}$. Los conectores de color rojo muestran las nuevas subfunciones implementadas, relacionadas con el cálculo del vector $\mathbf{f}_b^{(e)}$ del elemento hexaédrico ($\text{tipe}=302$) y prisma ($\text{tipr}=303$).

4.5.4. Función FELEMS

Esta función calcula el vector de fuerzas equivalentes a la acción de las fuerzas de superficie del elemento finito tridimensional. La instrucción que llama a la función es:

$$[\langle f_{el} \rangle] = \text{FELEMS}(\langle tipe \rangle, \langle xye \rangle, \langle xyc \rangle, \langle ela \rangle, \langle fde \rangle);$$

donde:

- $\langle f_{el} \rangle$ corresponde al vector de fuerzas de superficie del elemento finito $\tilde{\mathbf{f}}_s^{(e)}$,
- $\langle tipe \rangle$ es el identificador del tipo de elemento,
- $\langle xye \rangle$ es la matriz que contiene las coordenadas de los nudos del elemento,
- $\langle xyc \rangle$ son las coordenadas de los nudos de la cara cargada, y
- $\langle ela \rangle$ es el identificador de los nudos del elemento.
- $\langle fde \rangle$ corresponde a las características de la carga distribuida extraídas de la tabla *FDI* (Linero *et al.*, 2020).

La tabla *FDI* se amplió para que permita llevar las características de las caras cargadas de cuatro nudos, por lo que $\langle fde \rangle$ es de la forma:

$$[\langle fde \rangle] = (\langle iele \rangle, \langle nudi \rangle, \langle nudj \rangle, \langle nudk \rangle, \langle nudl \rangle, \langle pre1 \rangle, \langle pre2 \rangle, \langle pre3 \rangle, \langle gloc \rangle);$$

donde:

- $\langle iele \rangle$ es el indicador del elemento, y
- $\langle nudi \rangle$, $\langle nudj \rangle$, $\langle nudk \rangle$ y $\langle nudl \rangle$ son los identificadores de los nudos de la cara cargada.

Las variables $\langle pre1 \rangle$, $\langle pre2 \rangle$ y $\langle pre3 \rangle$ dependen del tipo de presión que se aplique en la cara del elemento, las cuales se indican mediante la variable $\langle gloc \rangle$ de la siguiente manera:

- Si $\langle gloc = 0 \rangle$, la presión que está aplicada en la cara del elemento corresponde a una presión uniforme en coordenadas globales, por lo que las variables $\langle pre1 \rangle$, $\langle pre2 \rangle$ y $\langle pre3 \rangle$ corresponden a las componentes de ésta presión: $p_{cx}^{(e)}$, $p_{cy}^{(e)}$ y $p_{cz}^{(e)}$, respectivamente.

- Si $\langle gloc = 1 \rangle$, la presión en la cara es una presión uniforme normal a la cara, cuyo valor $p_{cn}^{(e)}$ se almacena en la variable $\langle pre2 \rangle$. $\langle pre1 \rangle$ y $\langle pre3 \rangle$ en este caso no se utilizan y toman un valor igual a cero.
- Si $\langle gloc = 2 \rangle$, la presión en la cara del elemento es una presión hidrostática que está en función del peso específico del agua γ_w y de su altura h_w , las cuales se almacenan en los parámetros $\langle pre1 \rangle$ y $\langle pre2 \rangle$. El parámetro $\langle pre3 \rangle$ no varía y se fija como cero.

En este trabajo se implementó el cálculo del vector de fuerzas equivalentes a las fuerzas de superficie para el elemento hexaédrico a través de la subfunción FHEXAS, y para el prisma mediante la función FPRISS, como se muestra en la [Figura 4.5](#). La instrucción de llamada para la subfunción FHEXAS es:

$$[\langle fel \rangle] = \text{HEXAS}(\langle xye \rangle, \langle xyc \rangle, \langle ela \rangle, \langle fde \rangle);$$

donde los parámetros de entrada $\langle xye \rangle$, $\langle xyc \rangle$, $\langle ela \rangle$ y $\langle fde \rangle$, usados también en la subfunción FPRISS son los mismos parámetros utilizados para la función FELEMS.

Los procedimientos utilizados para el cálculo del vector de las fuerzas equivalentes $\tilde{\mathbf{f}}_s^{(e)}$ en las subfunciones FHEXAS Y FPRISS dependen del valor de la variable $\langle gloc \rangle$, así:

- Si $\langle gloc = 0 \rangle$, el procedimiento para el cálculo del vector de fuerzas equivalentes a la acción de una presión uniforme en sistema coordenado global para una cara del hexaedro, utilizado por la subfunción FHEXAS, corresponde a la [Subsección 3.6.10](#). Si en cambio la presión se aplica en una cara de un elemento prisma, mediante la subfunción FPRISS, el procedimiento utilizado corresponde al mostrado en la [Subsección 3.7.9](#).
- Si $\langle gloc = 1 \rangle$, el procedimiento utilizado por la subfunción FHEXAS para el cálculo del vector de fuerzas equivalentes a la acción de una presión uniforme normal, para una cara del hexaedro corresponde a la [Subsección 3.6.11](#). Si en cambio la presión se aplica en una cara de un elemento prisma, mediante la subfunción FPRISS, el procedimiento utilizado corresponde al mostrado en la [Subsección 3.7.10](#).
- Si $\langle gloc = 2 \rangle$, el procedimiento utilizado por la subfunción FHEXAS para calcular el vector de fuerzas equivalentes a la acción de una presión hidrostática, para una cara del hexaedro corresponde a la [Subsección 3.6.12](#). Si en cambio la presión se aplica en una cara de un elemento prisma, mediante la subfunción FPRISS, el procedimiento utilizado corresponde al mostrado en la [Subsección 3.7.11](#).

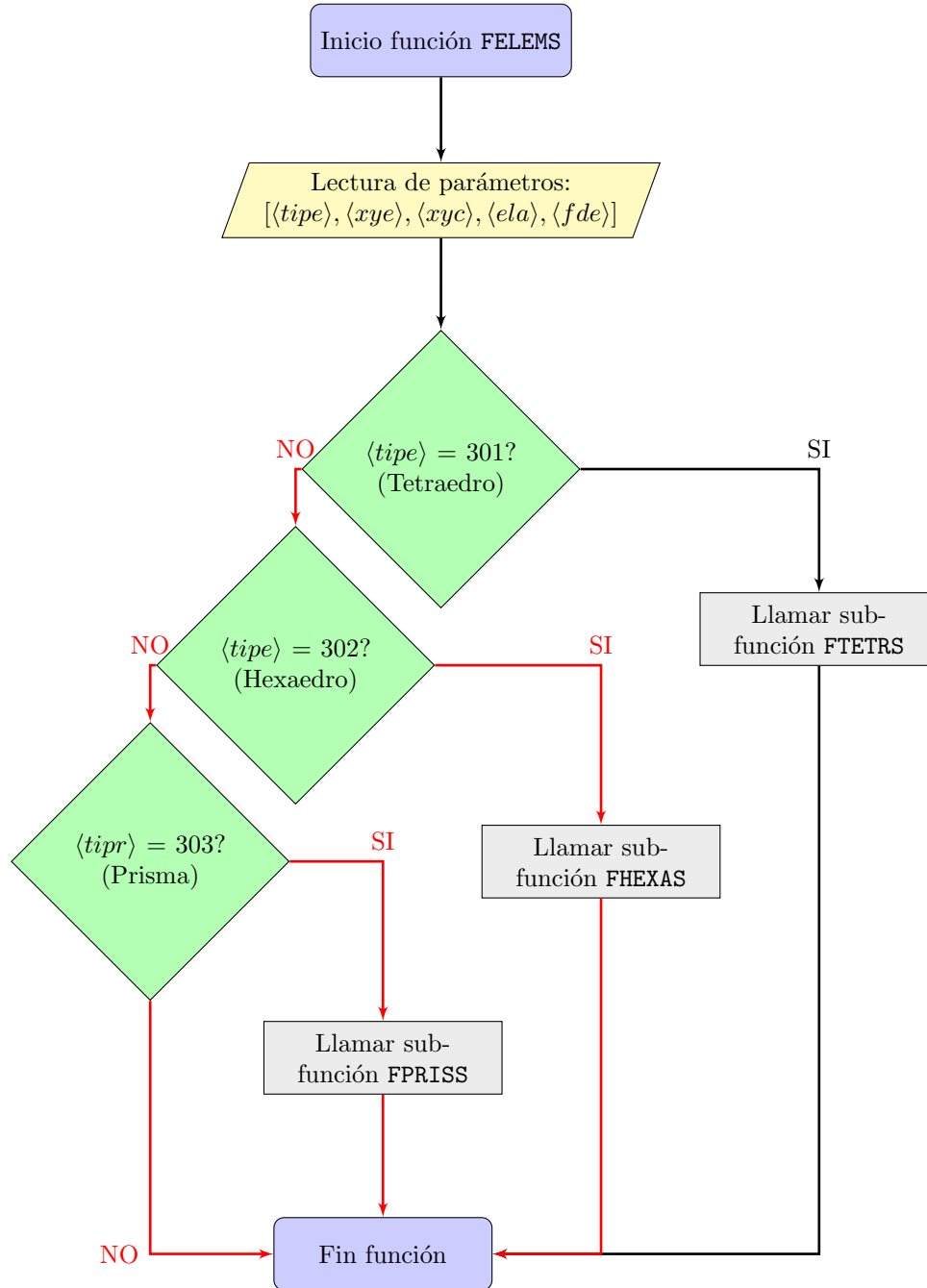


Figura 4.5: Diagrama de flujo de la función FELEMS, que calcula el vector $\mathbf{f}_s^{(e)}$. Los conectores de color rojo muestran las nuevas subfunciones implementadas en este trabajo, relacionadas con el cálculo del vector $\mathbf{f}_s^{(e)}$ del elemento hexaédrico ($\text{tipe}=302$) y prisma ($\text{tipr}=303$).

4.5.5. Función PBNUEL

Esta función identifica el número de nudos de un elemento finito y de sus caras, y el número de puntos de Gauss recomendado a partir de su tipología. La instrucción de llamada de esta función es:

$$[\langle nuel \rangle, \langle nuca \rangle, \langle pgau \rangle] = \text{PBNUEL}(\langle tipe \rangle);$$

donde:

- $\langle nuel \rangle$ es el número de nudos del elemento,
- $\langle nuca \rangle$ corresponde al número de caras,
- $\langle pgau \rangle$ es el número de puntos de Gauss recomendado y
- $\langle tipe \rangle$ es un escalar asociado al tipo de elemento.

Se modificó esta función para que el parámetro de entrada $\langle tipe \rangle$ pueda tomar los valores de $\langle tipe \rangle = 302$ para el caso del elemento hexaedro de 8 nudos, y $\langle tipe \rangle = 303$ para el caso del prisma de 6 nudos, como se muestra en la [Figura 4.6](#).

De esta manera, la función devuelve los siguientes parámetros, según el valor de $\langle tipe \rangle$:

- Si $\langle tipe \rangle = 301$, $\langle nuel \rangle = 4$, $\langle nuca \rangle = 3$ y $\langle pgau \rangle = 1$.
- Si $\langle tipe \rangle = 302$, $\langle nuel \rangle = 8$, $\langle nuca \rangle = 4$ y $\langle pgau \rangle = 8$.
- Si $\langle tipe \rangle = 303$, $\langle nuel \rangle = 6$, $\langle nuca \rangle = 4$ y $\langle pgau \rangle = 6$.

4.5.6. Función PBPGAU

Esta función devuelve los parámetros de ponderación y ubicación de los puntos en coordenadas naturales determinados por el método de integración numérica de la cuadratura de Gauss. La instrucción de llamada de la función es:

$$[\langle mtx \rangle] = \text{PBPGAU}(\langle ngau \rangle, \langle ndim \rangle, \langle ennu \rangle);$$

en donde el parámetro de salida $\langle mtx \rangle$ es una tabla con la ubicación de los puntos de ponderación y sus respectivas constantes de ponderación. El parámetro de entrada $\langle ngau \rangle$ corresponde al número de puntos de Gauss, el cual se asocia además con el tipo de elemento finito. El parámetro $\langle ndim \rangle$ corresponde al número de dimensiones del problema.

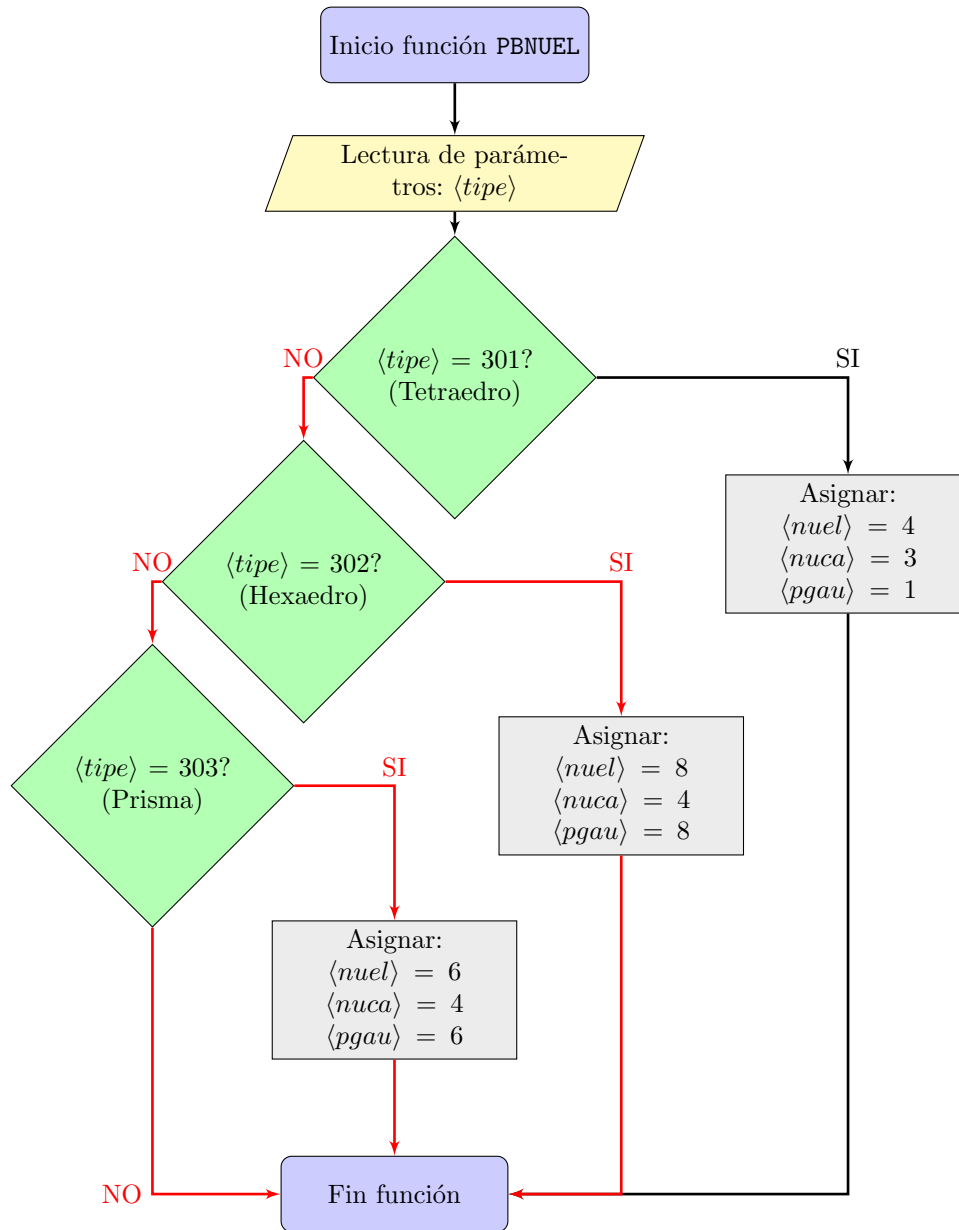


Figura 4.6: Diagrama de flujo de la función PBUUEL, que devuelve el número de nodos de un elemento finito $\langle nuel \rangle$ y de sus caras $\langle nuca \rangle$, y el número de puntos de Gauss $\langle pgau \rangle$. Los conectores de color rojo muestran las nuevas subfunciones implementadas en este trabajo, relacionadas con el elemento hexaédrico y prisma.

Para el caso específico del elemento hexaédrico, las coordenadas de los puntos de Gauss y las ponderaciones que devuelve la función son los mostrados en la [Tabla 3.2\(b\)](#). Asimismo, las coordenadas naturales de los nudos son los mostrados en la [Tabla 3.1](#).

Para el caso del elemento prisma, los puntos de Gauss son los mostrados en la [Tabla 3.4](#), y las coordenadas naturales de los nudos son las mostradas en la [Tabla 3.3](#).

En general, la función PBPGAU permite extraer la ubicación de diferentes puntos en el interior del elemento, de acuerdo con el parámetro $\langle ennu \rangle$, así:

- Si $\langle ennu \rangle = 0$, la función devuelve en la matriz $\langle mtx \rangle$ las coordenadas naturales de los puntos de Gauss y sus ponderaciones.
- Si $\langle ennu \rangle = 1$, la función devuelve en la matriz $\langle mtx \rangle$ los nudos del elemento en coordenadas naturales.
- Si $\langle ennu \rangle = 2$, la función devuelve en la matriz $\langle mtx \rangle$ las coordenadas del centro del elemento, en coordenadas naturales.

4.6. Rutinas nuevas en la etapa de proceso

En esta sección se describen las rutinas nuevas que se implementaron en el desarrollo de este trabajo. Particularmente, dichas funciones se utilizan en el cálculo de la matriz Jacobiana y de la matriz de funciones de forma de los elementos.

4.6.1. Función JELEME

Esta función calcula la matriz Jacobiana, su inversa y su determinante para los elementos tridimensionales. La instrucción que llama a esta función tiene la forma:

$$[\langle jac \rangle, \langle ij c \rangle, \langle det j \rangle] = \text{JELEME}(\langle xye \rangle, \langle der \rangle, \langle tipe \rangle);$$

A su vez este procedimiento contiene las funciones JHEXAE y JPRISM, las cuales calculan la matriz Jacobiana, su inversa y su determinante para el elemento hexaédrico y prisma, respectivamente. Para calcular la matriz Jacobiana del elemento hexaédrico y prisma, se utiliza la [Ecuación 3.36](#).

Con la matriz Jacobiana de tamaño 3×3 , su inversa y su determinante se calculan utilizando las funciones que trae incorporadas el lenguaje de GNU-OCTAVE (compatibles con MATLAB). Estas instrucciones son `inv(JAC)` para el cálculo de la inversa y `det(JAC)` para el cálculo del determinante.

4.6.2. Función NELEME

Esta función puede calcular la matriz de funciones de forma y las derivadas estas últimas con respecto a las coordenadas naturales en los elementos tridimensionales. La instrucción que llama a esta función tiene la forma:

$$[\langle nel \rangle] = \text{NELEME}(\langle tipe \rangle, \langle tare \rangle, \langle xi \rangle, \langle eta \rangle, \langle mu \rangle);$$

donde el parámetro de salida $\langle nel \rangle$ puede ser: la matriz de las funciones de forma o la matriz de las derivadas de las funciones de forma con respecto a las coordenadas naturales, dependiendo del parámetro de entrada $\langle tare \rangle$ y del parámetro de identificación del elemento $\langle tipe \rangle$, de la siguiente manera:

- Si $\langle tare \rangle = 1$ y $\langle tipe \rangle = 302$, la función devuelve la matriz de las funciones de forma para el elemento hexaédrico, evaluada en el punto de coordenadas naturales (ξ, η, μ) , de acuerdo con la [Ecuación 3.28c](#).
- Si $\langle tare \rangle = 2$ y $\langle tipe \rangle = 302$, la función retorna la matriz de las derivadas de las funciones de forma con respecto a las coordenadas naturales para el elemento hexaédrico, evaluada en el punto (ξ, η, μ) , como se muestra en la [Ecuación 3.38](#).
- Si $\langle tare \rangle = 1$ y $\langle tipe \rangle = 303$, la función devuelve la matriz de las funciones de forma para el elemento prisma, evaluada en el punto de coordenadas naturales (ξ, η, μ) , de acuerdo a la [Ecuación 3.77](#).
- Si $\langle tare \rangle = 2$ y $\langle tipe \rangle = 303$, la función retorna la matriz de las derivadas de las funciones de forma con respecto a las coordenadas naturales, evaluadas en el punto (ξ, η, μ) para el elemento prisma, como se muestra en la [Ecuación 3.82](#).

Las coordenadas naturales ξ , η y μ corresponden a los parámetros de entrada $\langle xi \rangle$, $\langle eta \rangle$ y $\langle mu \rangle$, respectivamente.

4.7. Creación del archivo de postproceso

La función `IMGMSH` en el programa PEFiCA 2.0 (Linero *et al.*, 2020), genera el archivo de postproceso, el cual contiene los resultados del problema elástico tridimensional analizado. Esta función crea los archivos con extensión `.pos` y `.pos.opt`, en el formato adecuado para que puedan ser leídos y mostrados a través del entorno gráfico de post proceso del programa GMSH.

En este trabajo se amplió la función `IMGMSH` para generar las tablas correspondientes a los desplazamientos, fuerzas, esfuerzos y deformaciones en los nudos de

elementos hexaédricos y prismas, en un archivo de salida para su posterior visualización en GMSH. Asimismo, se modificó esta rutina considerando adicionalmente las diferentes categorías de tipo de elemento finito. La instrucción para ejecutar esta función es de la forma:

$$\text{IMGSH}(\langle adad \rangle, \langle nnud \rangle, \langle nele \rangle, \langle nnue \rangle, \langle ngau \rangle, \langle ncat \rangle, \langle xyz \rangle, \dots \\ \dots \langle ele \rangle, \langle sup \rangle, \langle uxy \rangle, \langle fxy \rangle, \langle sre \rangle, \langle ere \rangle, \langle pro \rangle, \langle cat \rangle, \langle nct \rangle);$$

Los parámetros de entrada de la función IMGSH son:

- $\langle adad \rangle$ es el nombre del archivo de datos sin extensión,
- $\langle nnud \rangle$ es el número de nudos,
- $\langle nele \rangle$ es el número de elementos,
- $\langle nnue \rangle$ es el número de nudos máximo de un elemento,
- $\langle ngau \rangle$ es el número máximo de puntos de Gauss de un elemento,
- $\langle ncat \rangle$ es el número de categorías,
- $\langle xyz \rangle$ es la tabla de coordenadas de los nudos,
- $\langle ele \rangle$ es la tabla de conectividades de los elementos,
- $\langle sup \rangle$ es la tabla de identificación de superficies asociada al elemento,
- $\langle uxy \rangle$ es la tabla de desplazamientos en los nudos,
- $\langle fxy \rangle$ es la tabla de fuerzas en los nudos,
- $\langle sre \rangle$ es la tabla de esfuerzos en los elementos,
- $\langle ere \rangle$ es la tabla de deformaciones en los elementos,
- $\langle pro \rangle$ es el número indicador para la escritura de los esfuerzos y deformaciones promedio en los nudos o en el interior de los elementos
- $\langle cat \rangle$ es la tabla de categorías de los elementos,
- $\langle nct \rangle$ es la tabla auxiliar para impresión de archivo de posproceso.

La función `IMGMSH` utiliza otras sub funciones vinculadas al posprocesamiento de la solución del problema. La sub función que fue modificada en este trabajo es `ORSOCA`. Esta función crea la tabla del promedio de un resultado sobre los nudos a partir de la misma cantidad obtenida en los extremos de los elementos que pertenecen a la misma categoría. La instrucción de llamada de la función tiene la forma:

$$[\langle nud \rangle, \langle nuca \rangle] = \text{ORSOCA}(\langle ele \rangle, \langle rel \rangle, \langle nnud \rangle, \langle ncom \rangle, \langle icat \rangle, \langle nct \rangle, \langle ncat \rangle);$$

donde los parámetros de salida de la función son:

- $\langle nud \rangle$ es la tabla de resultados promedio de la cantidad en los nudos, y
- $\langle nuca \rangle$ es el número de nudos de la categoría.

En cambio, los parámetros de entrada de la función son los siguientes:

- $\langle ele \rangle$ es la tabla de categorías y conectividades de los elementos,
- $\langle rel \rangle$ es la tabla con los esfuerzos o deformaciones en los nudos de cada elemento,
- $\langle nnud \rangle$ es el número de nudos del elemento,
- $\langle ncom \rangle$ es el número de componentes de los resultados,
- $\langle icat \rangle$ es el indicador de la categoría de los elementos,
- $\langle ncat \rangle$ es el número de categorías y
- $\langle nct \rangle$ es una tabla auxiliar para impresión del archivo de postproceso, cuyo número de filas corresponde a la cantidad de categorías del problema. La sintaxis de cada fila de esta tabla es de la forma:

$$[\langle nelc \rangle, \langle nnue \rangle, \langle mine \rangle, \langle maxe \rangle]$$

El parámetro $\langle nelc \rangle$ es el número de elementos de la categoría, $\langle nnue \rangle$ es el número de nudos del elemento de la categoría, $\langle mine \rangle$ es la mínima posición del primer elemento de la categoría en la matriz ELE y $\langle maxe \rangle$ es la máxima posición del primer elemento de a categoría en la matriz ELE.

Esta función se modificó utilizando los nuevos parámetros $\langle ncat \rangle$ y $\langle nct \rangle$, para lograr extraer el contenido de la tabla de resultados $\langle rel \rangle$ de acuerdo a la categoría $\langle icat \rangle$.

Capítulo 5

Ejemplos de aplicación y análisis de resultados

En este capítulo se muestran algunos problemas tridimensionales de sólidos que fueron resueltos utilizando el programa presentado. Para cada uno de éstos problemas se muestra además:

- Un archivo que contiene la geometría, los desplazamientos conocidos, las acciones aplicadas y los parámetros de construcción de malla en formato de preproceso de GMSH, con extensión `.geo`.
- Un archivo que contiene la malla de elementos finitos utilizada, con extensión `.msh`.
- Dos archivos de resultados en el formato de postproceso de GMSH, con extensiones `.pos` y `.pos.opt`.

El objetivo en este capítulo es realizar una comparación entre las soluciones obtenidas mediante el método de los elementos finitos con las siguientes características:

- Análisis realizado con el programa desarrollado en este trabajo final, construyendo mallas de elementos hexaédricos de 8 nudos, utilizando el programa GMSH.
- Análisis realizado con el programa PEFiCA 2.0, construyendo mallas de elementos tetraédricos de 4 nudos, utilizando el programa GMSH.
- Análisis realizado con el programa comercial ANSYS, construyendo mallas de elementos hexaédricos de 8 nudos con integración simplificada (SOLID185). Esta simulación es utilizada como verificación.

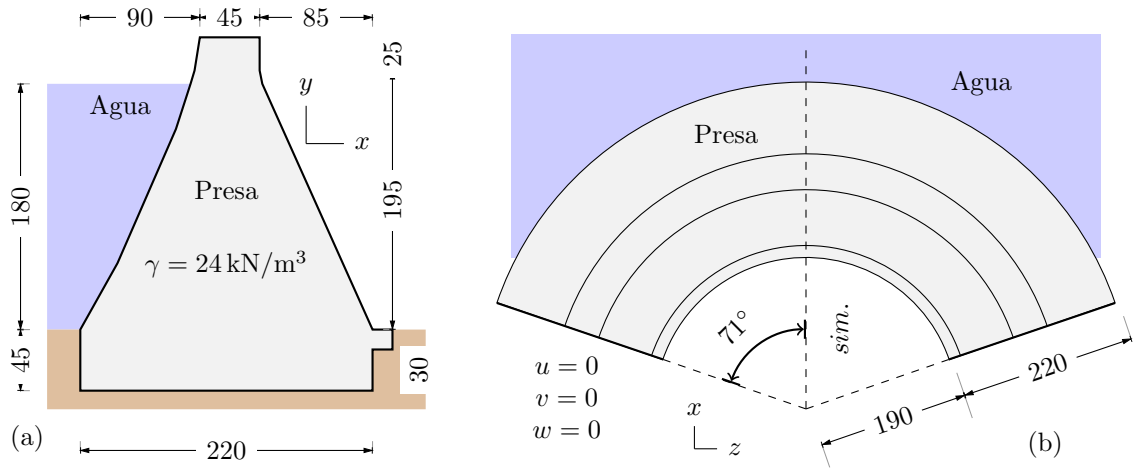


Figura 5.1: Geometría de la presa sometida a la presión hidráulica y su peso propio: (a) sección transversal y (b) vista de planta. Unidades en metros.

5.1. Ejemplo 1: Presa de bóveda sometida a la presión hidráulica y su peso propio

Este ejemplo de aplicación está basado en el presentado en el capítulo 9 de la referencia (Linero *et al.*, 2020). Esta estructura tiene una geometría similar a la presa Hoover en Estados Unidos y está sometida a la presión hidráulica y a su propio peso. Se encuentra empotrada en la roca en sus lados y en su base, en donde se asume que las componentes de desplazamiento son nulas, es decir, $u = 0$, $v = 0$ y $w = 0$.

El agua tiene una altura $h_w = 180.0$ m y un peso específico $\gamma_w = 9.8$ kN/m³, como se ilustra en la Figura 5.1(a). La presa es de concreto con peso específico $\gamma = 24$ kN/m³, módulo de Young $E = 45.6$ kN/m² y relación de Poisson $\nu = 0.21$.

El problema es simétrico con respecto al eje x mostrado en la Figura 5.1(b), razón por la cual se simula solamente la mitad de la presa. En su plano de simetría, la componente de desplazamiento en dirección z es igual a cero, es decir $w = 0$. Mientras que en el plano del costado opuesto, correspondiente al apoyo lateral en la roca, las componentes x , y y z del desplazamiento son iguales a cero, es decir $u = 0$, $v = 0$ y $w = 0$.

5.1.1. Creación de las mallas de elementos finitos

Para la creación de la malla utilizando elementos hexaédricos mediante GMSH, se define la sección transversal como un plano compuesto por superficies cuadrilaterales, a partir de las cuales se genera su volumen por rotación. Lo anterior permite generar

una malla estructurada con elementos hexaédricos.

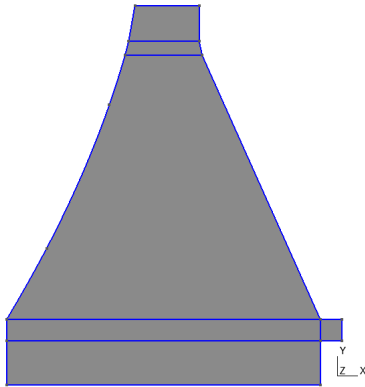


Figura 5.2: Sección transversal dividida en cuadriláteros realizada en el programa GMSH .

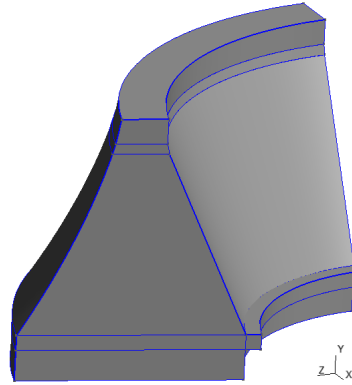


Figura 5.3: Volumen completo de la presa compuesto por volúmenes asociados a superficies cuadrilaterales, realizada en el programa GMSH.

En la [Figura 5.2](#) se muestra la división de la sección transversal en cuadriláteros y en la [Figura 5.3](#) se presenta el volumen compuesto generado en GMSH. La información detallada de la creación de la malla se presenta en la [Sección 7.1](#).

Considerando las condiciones iniciales del problema, se asignan las propiedades de los materiales, se introducen los parámetros para generar la malla estructurada en cada línea, se indican los planos con desplazamientos restringidos y también se aplica la presión de agua siguiendo el procedimiento descrito en la [Sección 7.1](#).

En la primera simulación se generó una malla de elementos finitos hexaédricos presentada en la [Figura 5.4](#) y en la [Figura 5.5](#). Se realizó el enmallado con 8862 nudos, y 7580 hexaedros. La segunda simulación tiene una malla de 45480 elementos finitos tetraédricos y 8864 nudos. Esta última se presenta en la [Figura 5.6](#) y en la [Figura 5.7](#).

Cabe resaltar que aunque el número de nudos en las dos mallas es casi el mismo, la cantidad de tetraedros del segundo modelo es 6 veces el número de hexaedros de la primera. Esto se debe a que para conformar un volumen hexaédrico, el generador de malla de GMSH construye 6 elementos tetraédricos.

Con las mallas de hexaedros y tetraedros, se generan los archivos `.msh` correspondientes, mediante el procedimiento descrito en la [Sección 7.1](#).

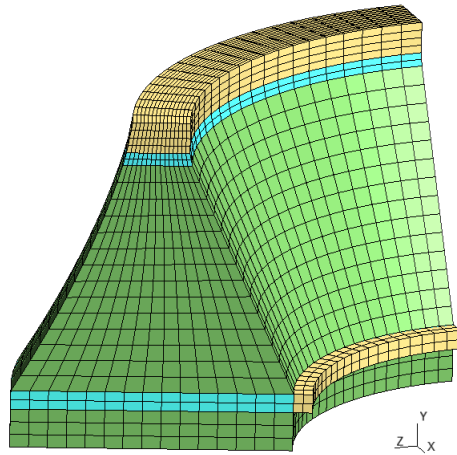


Figura 5.4: Malla de elementos finitos hexaédricos en el programa GMSH. Los colores ilustran los sub volúmenes asociados a la extrusión de las superficies cuadrilaterales.

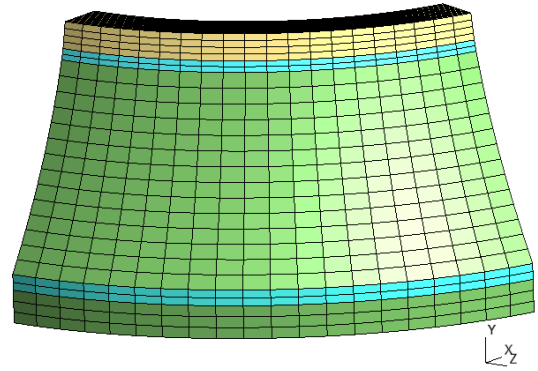


Figura 5.5: Malla de elementos finitos hexaédricos en el programa GMSH. Los colores ilustran los sub volúmenes asociados a la extrusión de las superficies cuadrilaterales.

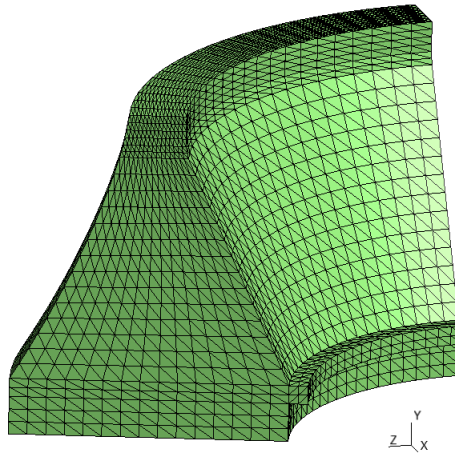


Figura 5.6: Malla de elementos finitos tetraédricos en el programa GMSH. El elemento tetraédrico no requiere sub volúmenes para su implementación

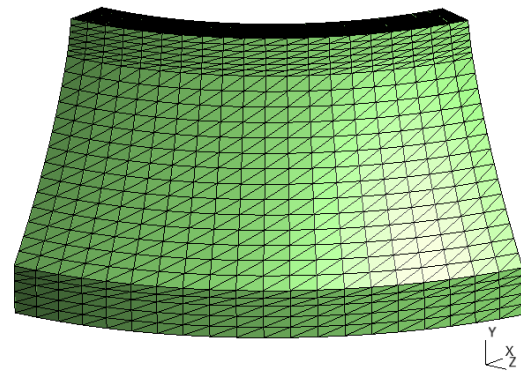


Figura 5.7: Malla de elementos finitos tetraédricos en el programa GMSH. El elemento tetraédrico no requiere sub volúmenes para su implementación.

5.1.2. Solución del problema utilizando el programa desarrollado

Luego de obtener los archivos de preproceso con extensión `.msh` que contiene cada una de las mallas de elementos creadas con las condiciones del problema, se ejecuta el programa desarrollado mediante la instrucción `PEFICA HooverH 10`, desde el directorio activo `\PEFTRI` en el entorno de GNU- OCTAVE. El programa analiza el problema y produce los archivos con extensión `.pos` y `.pos.opt` con los resultados, para que puedan ser leídos por el entorno gráfico de post proceso del programa GMSH. Este procedimiento se explica en detalle en la [Sección 7.2](#). Particularmente, el procedimiento de cálculo realizado por el programa desarrollado, para problemas mecánicos tridimensionales con elementos tetraédricos, es el mismo implementado en el programa PEFiCA 2.0.

5.1.3. Resultados

A partir de los archivos de los resultados creados por el programa desarrollado, desde el módulo de post procesamiento del programa GMSH se leen los archivos con extensión `.pos`.

La configuración deformada de la mitad de la presa bajo la acciones indicadas anteriormente y la norma del vector desplazamiento en los nudos se ilustran en la [Figura 5.8](#) para la simulación con elementos hexaédricos y en la [Figura 5.9](#) para la simulación con elementos tetraédricos.

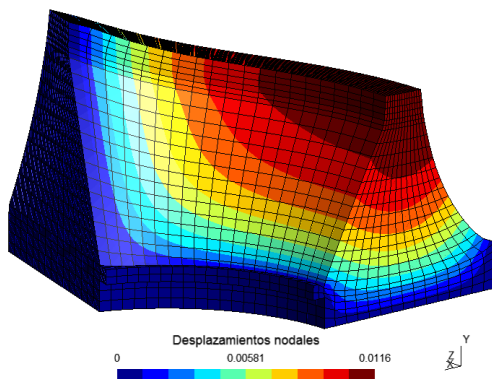


Figura 5.8: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada con hexaedros.

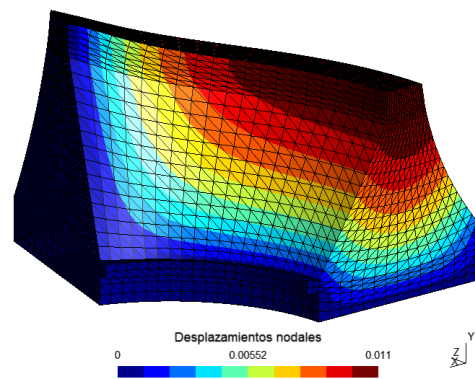


Figura 5.9: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada con tetraedros.

La distribución del esfuerzo principal menor σ_3 para la simulación con elementos hexaédricos se muestra en la [Figura 5.10](#) y para la simulación con elementos

tetraédricos se muestra en la [Figura 5.11](#).

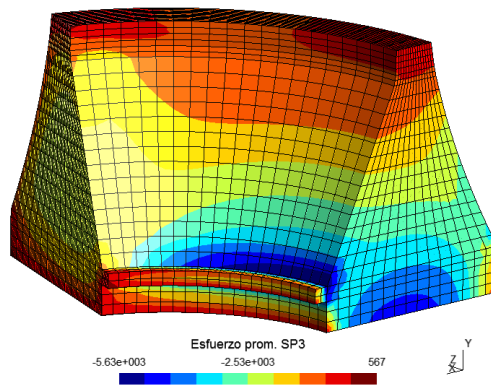


Figura 5.10: Distribución de esfuerzo principal menor σ_3 medido en kN/m^2 de la simulación realizada con hexaedros.

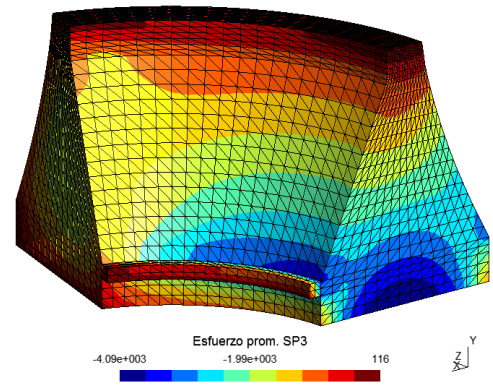


Figura 5.11: Distribución de esfuerzo principal menor σ_3 medido en kN/m^2 de la simulación realizada con tetraedros.

También se realizó la simulación del problema en el programa ANSYS, en donde se utilizó una malla con 8862 nudos y 7580 elementos hexaédricos como se muestra en la [Figura 5.12](#) y en la figura [Figura 5.13](#).

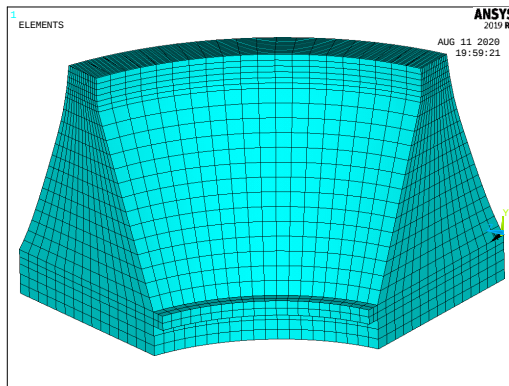


Figura 5.12: Malla de elementos finitos hexaédricos de tipo SOLID185 en el programa ANSYS

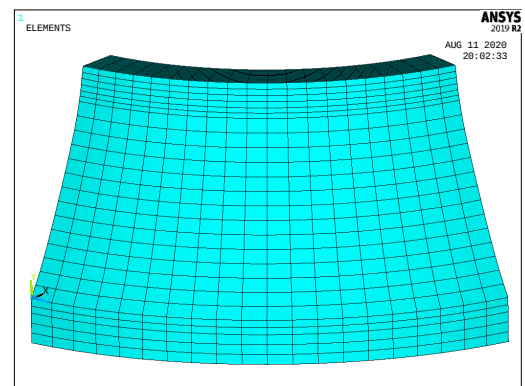


Figura 5.13: Malla de elementos finitos hexaédricos de tipo SOLID185 en el programa ANSYS

En la simulación realizada en el programa ANSYS se utilizó el elemento "SOLID185" (Kohnke, 2009), con la opción "Simple enhanced strain". Esta opción de integración introduce selectivamente 9 grados de libertad internos adicionales en algunos elementos para evitar el error generado por el bloqueo por cortante. Este error hace que el elemento parezca más rígido y produce desplazamientos menores de lo

que deberían ser. Esta técnica se puede consultar en los trabajos de Simo y Rifai (Simo & Rifai, 1990). Los resultados de esta simulación se toman como referencia.

Los desplazamientos en los nudos de la simulación realizado en el programa ANSYS se muestran en la Figura 5.14. Asimismo, la distribución del esfuerzo principal menor σ_3 para esta simulación se muestra en la Figura 5.15.

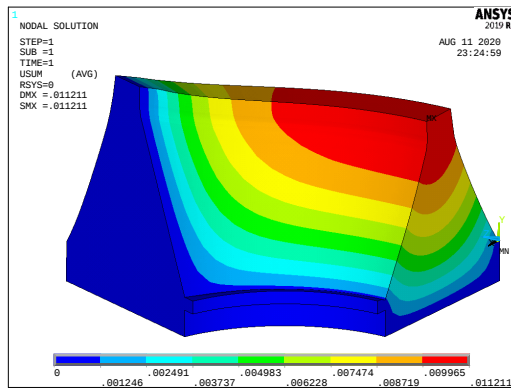


Figura 5.14: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada en ANSYS

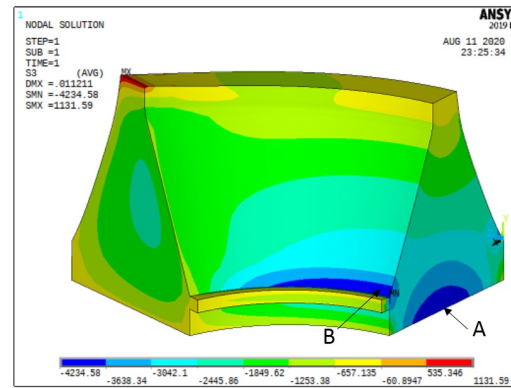


Figura 5.15: Distribución de esfuerzo principal menor σ_3 medido en kN/m^2 de la simulación realizada en ANSYS.

5.1.4. Análisis de resultados

A partir de la solución del problema mecánico realizada con elementos hexaédricos y tetraédricos utilizando el programa desarrollado en este trabajo final de maestría, y a la simulación realizada en el programa ANSYS, en la Tabla 5.1 se comparan los resultados en un mismo punto de la norma del vector de desplazamiento y del esfuerzo principal menor σ_3 .

El valor máximo de la norma del vector de desplazamiento en los nudos que se utiliza como comparación, ocurre en todos los casos en la esquina superior de la cara no restringida de la presa, como se puede observar en la Figura 5.8, la Figura 5.9 y en la Figura 5.14.

El valor del esfuerzo principal menor que se compara en los tres casos ocurre en la cara no restringida de la base de la presa, indicada como A en la Figura 5.15. Se tomaron los valores de esfuerzo principal menor en el nudo ubicado en A para las tres simulaciones. Estas distribuciones de esfuerzos se observan en la Figura 5.10, la Figura 5.11 y en la Figura 5.15.

En la zona alrededor del punto B indicado en la Figura 5.15, el esfuerzo principal

menor presenta valores cercanos a los indicados en el punto A. Por tal razón este resultado no se comparó.

Tabla 5.1: Comparación de resultados

Programa	Elemento	$\ \mathbf{u}\ ^{max}$ (m)	σ_3^{max} (kN/m ²)	σ_1^{max} (kN/m ²)
De este trabajo	hexaedro de 8 nudos	0.0116	-4 679	6134
PEFiCA 2.0	tetraedro de 4 nudos	0.0110	-4 087	5010
ANSYS	hexaedro de 8 nudos con enriquecimiento de la deformación (SO-LID185)	0.0112	-4 071	5897

En la [Tabla 5.1](#) se observa que la norma del vector de desplazamiento obtenida en la simulación con elementos hexaédricos en el programa desarrollado difiere un 5.2 % con respecto a la simulación con la malla de elementos tetraédricos y un 3.4 % en relación con el programa ANSYS.

La pequeña diferencia entre los valores de la norma del desplazamiento entre la malla de elementos hexaédricos y de elementos tetraédricos se debe a la mayor rigidez que ofrece un volumen conformado por este último tipo de elemento, cuya función es de un orden menor al del elemento hexaédrico. Asimismo, el resultado obtenido con el programa desarrollado es similar al calculado en ANSYS.

También se observa que el esfuerzo principal menor σ_3 en el punto A de la simulación con elementos hexaédricos en el programa desarrollado, difiere un 12.7 % con respecto a la simulación con la malla de elementos tetraédricos y un 13.0 % en relación con el programa ANSYS.

Adicionalmente se observa que el esfuerzo principal mayor σ_1 con elementos hexaédricos en el programa desarrollado, difiere un 18.3 % con respecto a la simulación con la malla de elementos tetraédricos y un 3.9 % en relación con el programa ANSYS.

La diferencia moderada entre los valores de esfuerzo principal menor y esfuerzo principal mayor comparados, puede estar asociada a la aproximación de la deformación en PÉFICA 2.0 y ANSYS.

5.1.5. Análisis de sensibilidad

Se realizó un análisis de sensibilidad de la densidad de la malla con elementos hexaédricos, simulando el mismo problema con 7 mallas de diferente cantidad de elementos. En la [Figura 5.16](#) se relaciona el valor máximo absoluto de la norma del desplazamiento y del esfuerzo principal menor sobre los nudos de la malla, en función de la cantidad de elementos finitos ne . Allí se observa que las magnitudes representativas son asintóticas a un valor en la medida en que la malla tiene mayor número de elementos.

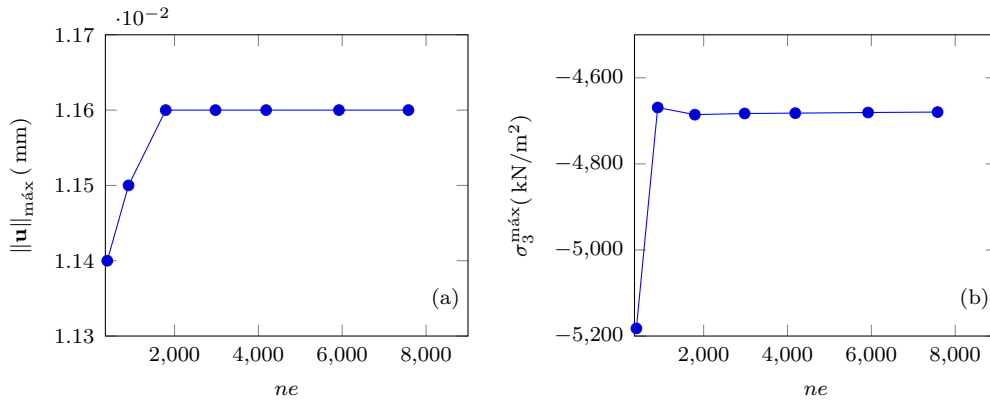


Figura 5.16: Valores máximos absolutos en la malla en función del número de elementos finitos de 7 mallas de elementos hexaédricos, correspondientes a: (a) la norma del desplazamiento de la malla (mm) y (b) el esfuerzo principal menor(kN/m²).

5.2. Ejemplo 2: Ensayo de resistencia al arrancamiento del concreto

A continuación se simula un ensayo para determinar la resistencia a la compresión del concreto, mediante la evaluación de la fuerza necesaria para desprender la cabeza de una barra de acero embebida en su interior. Este ensayo se encuentra estandarizado en la norma ASTM C-900 (ASTM C900, 2007) y ha sido simulado previamente con el programa PEFiCA 2.0 utilizando elementos tetraédricos (Linero *et al.*, 2020).

Este ensayo consiste en el arrancamiento de una barra acero de 10mm de diámetro y un cabezal de acero de 30mm de diámetro, embebidos en una región de concreto cilíndrica de 60mm de altura y 120mm de diámetro. Se aplica una carga puntual P en el extremo de la barra de acero. El concreto se apoya en un anillo circular. El concreto y el acero solamente están en contacto en la cara superior del cabezal, como se muestra en la Figura 5.17.

Debido a sus simetrías, el dominio del problema se puede simplificar a una cuarta parte, donde la componente del desplazamiento en dirección z es igual a cero en el plano de simetría xy y la componente de desplazamiento x es cero en el plano de simetría yz .

El acero tiene un módulo de elasticidad $E_a = 200$ GPa, una relación de Poisson de $\nu_c = 0.20$ y un esfuerzo de fluencia $\sigma_{ay} = 250$ MPa.

El concreto tiene un módulo de Young $E_c = 20$ GPa, una relación de Poisson $\nu_c = 0.25$ y resistencia a la compresión $\sigma_{cu} = 26.3$ MPa.

En las simulaciones se aplica una presión uniforme en el cara superior de la barra $w_y = P/A$, asumiendo un valor de $P = 1N$. Como el área transversal de la barra de

10mm de radio es $A = 78.54 \text{ mm}^2$, la carga $w_y = 12.72 \times 10^{-3} \text{ N/mm}^2$.

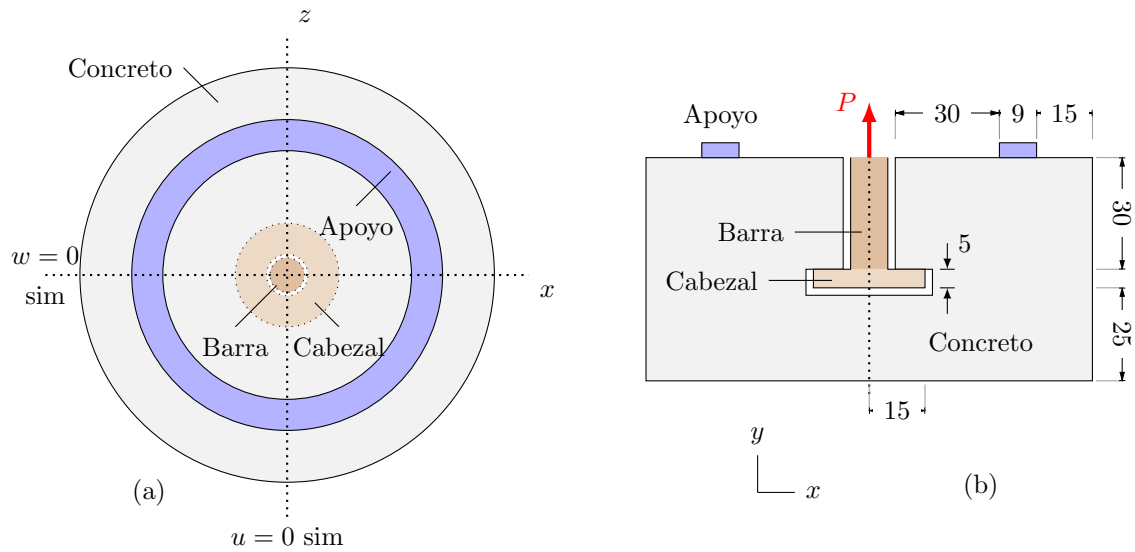


Figura 5.17: Geometría del ensayo resistencia al arrancamiento del concreto. (a)Plano xz y (b)Plano xy. Unidades en milímetros.

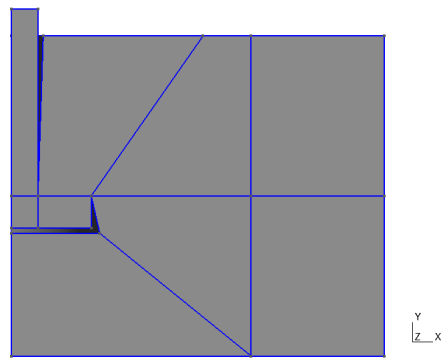


Figura 5.18: Sección transversal dividida en cuadriláteros, creada en el entorno gráfico del programa GMSH.

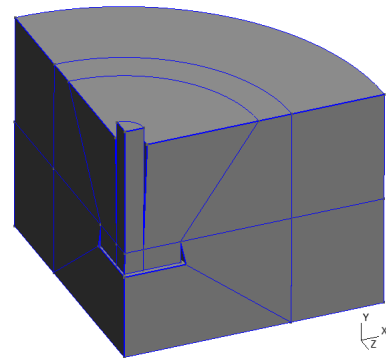


Figura 5.19: Volumen del dominio modelado conformado por volúmenes asociados a superficies cuadrilaterales, generado desde el entorno gráfico del programa GMSH.

5.2.1. Creación de las mallas de elementos finitos

Para la solución a este problema mecánico se realizaron tres simulaciones distintas: utilizando el programa desarrollado con elementos hexaédricos de 8 nudos, utilizando el programa desarrollado con elementos tetraédricos de 4 nudos, y utilizando el programa ANSYS con elementos hexaédricos con deformación enriquecida.

Para la primera simulación, se construye la sección transversal a partir de superficies cuadrilaterales, y luego se genera su volumen a partir de la extrusión de dicha sección. Se utiliza una malla estructurada con elementos hexaédricos.

En la [Figura 5.18](#) se muestra la división de la sección transversal en cuadriláteros y en la [Figura 5.19](#) se presenta el volumen compuesto generado en GMSH. La información detallada de la creación de la malla se presenta en la [Sección 7.1](#).

Después de construidas las entidades geométricas, se asignan a las superficies y volúmenes los desplazamientos restringidos en el concreto, la presión aplicada a la barra y las propiedades mecánicas de los materiales. A continuación se introducen los parámetros para generar la malla estructurada. Este procedimiento está detallado en la [Sección 7.1](#).

La malla de elementos finitos hexaédricos se presenta en la [Figura 5.20](#) y en la [Figura 5.21](#). Se realizó el enmallado con 7524 nudos, 416 prismas y 5980 hexaedros. Esta malla utiliza prismas en la zona cercana al eje del cilindro, para completar el volumen.

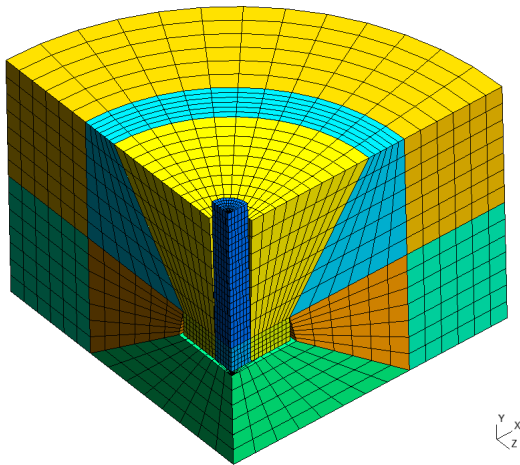


Figura 5.20: Malla de elementos finitos hexaédricos en el programa GMSH. Los colores ilustran los sub volúmenes asociados a la extrusión de las superficies cuadrilaterales.

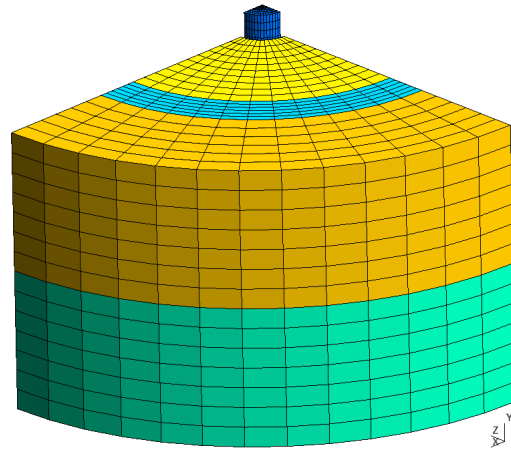


Figura 5.21: Malla de elementos finitos hexaédricos en el programa GMSH. Los colores ilustran los sub volúmenes asociados a la extrusión de las superficies cuadrilaterales.

En la segunda simulación se construyó una malla de 37128 elementos finitos

tetraédricos y 7524 nudos, la cual se presenta en la [Figura 5.22](#) y en la [Figura 5.23](#). Al igual que en el ejemplo anterior, se mantuvo el número de nudos en la malla de hexaedros, igual al de la malla de tetraedros.

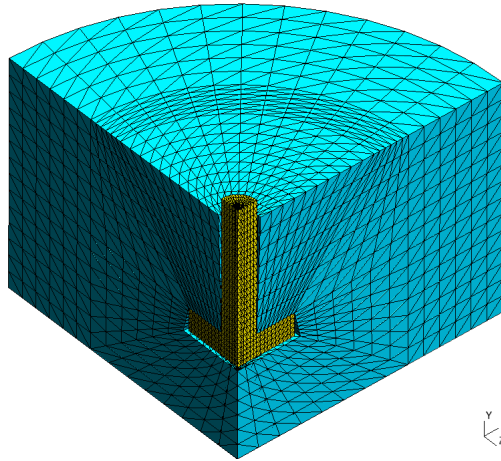


Figura 5.22: Malla de elementos finitos tetraédricos en el programa GMSH.

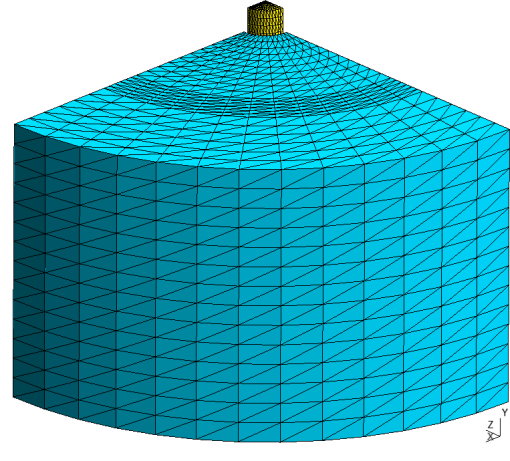


Figura 5.23: Malla de elementos finitos tetraédricos en el programa GMSH.

Con las mallas de hexaedros y tetraedros, se generan los archivos `.msh` correspondientes, mediante el procedimiento descrito en la [Sección 7.1](#).

5.2.2. Solución del problema

Con los archivos de preproceso con extensión `.msh` que contienen la malla de elementos y las condiciones del problema, se ejecuta la instrucción `PEFICA PullOutH 12`, desde el directorio activo `\PEFTRI` en el entorno de GNU- OCTAVE. El programa analiza y produce los archivos con extensión `.pos` y `.pos.opt` con los resultados, para ser leídos por el programa GMSH.

5.2.3. Resultados

A partir de los archivos de los resultados creados por el programa desarrollado, desde el módulo de post procesamiento del programa GMSH se leen los archivos con extensión `.pos`.

La configuración deformada de la simulación bajo las acciones indicadas anteriormente y la norma del vector desplazamiento en los nudos se ilustran en la [Figura 5.24](#) para la simulación con elementos hexaédricos y en la [Figura 5.25](#) para la simulación con elementos tetraédricos.

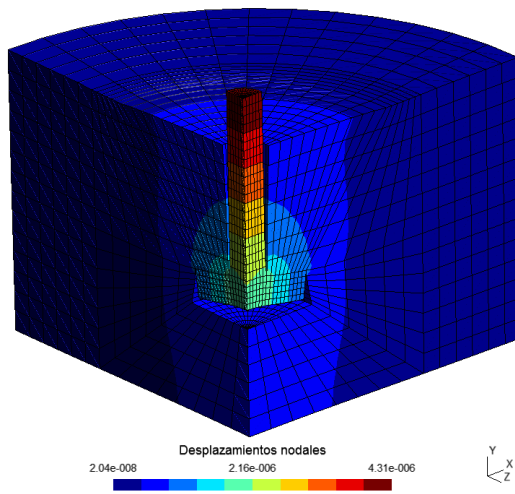


Figura 5.24: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada con hexaedros.

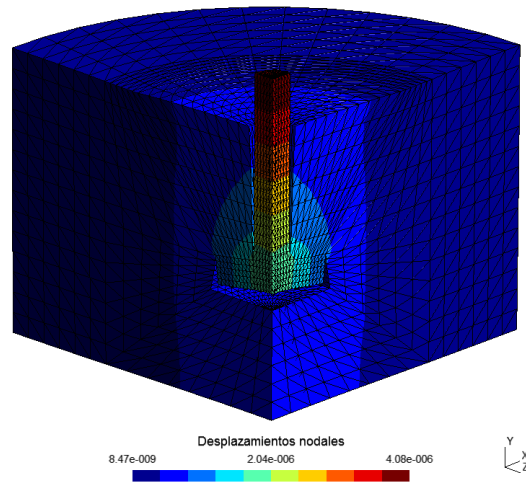


Figura 5.25: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada con tetraedros.

La distribución del esfuerzo principal menor σ_3 para la simulación con elementos hexaédricos se muestra en la [Figura 5.26](#) y para la simulación con elementos tetraédricos se muestra en la [Figura 5.27](#).

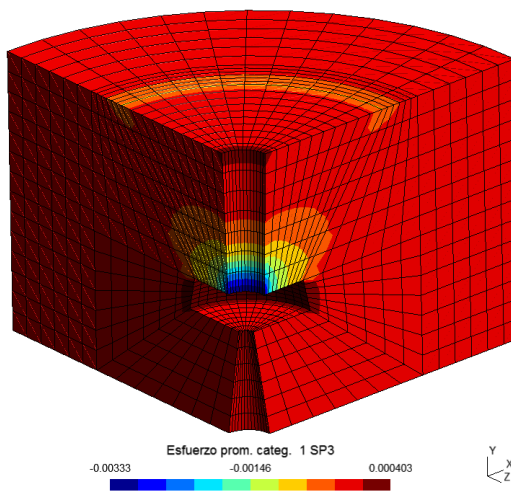


Figura 5.26: Distribución de esfuerzo principal menor σ_3 en el concreto, medido en kN/m^2 de la simulación realizada con hexaedros.

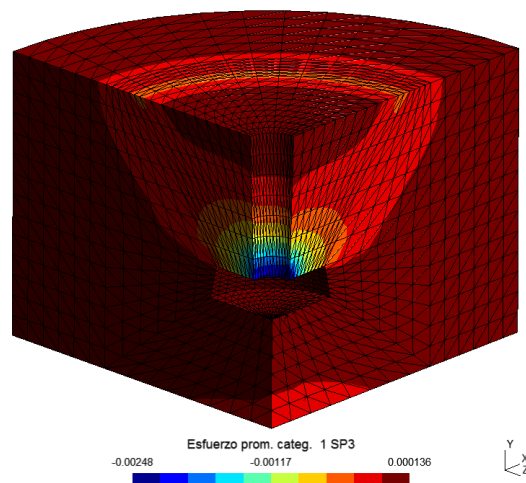


Figura 5.27: Distribución de esfuerzo principal menor σ_3 en el concreto, medido en kN/m^2 de la simulación realizada con tetraedros.

La distribución del esfuerzo de Von Mises σ_{vm} en el acero para la simulación con elementos hexaédricos se muestra en la [Figura 5.28](#) y para la simulación con elementos tetraédricos se muestra en la [Figura 5.29](#).

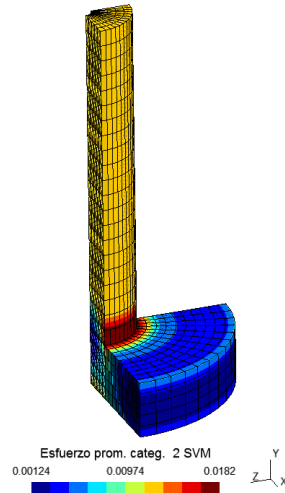


Figura 5.28: Distribución de esfuerzo de Von Mises σ_{vm} en el acero medido en kN/m^2 de la simulación realizada con hexaedros y prismas.

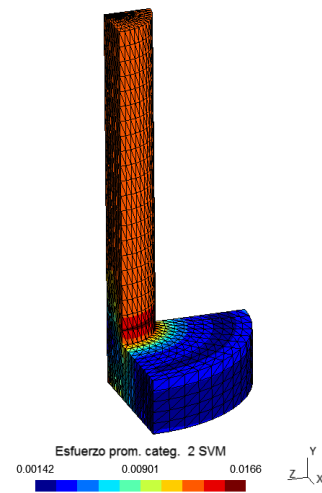


Figura 5.29: Distribución de esfuerzo de Von Mises σ_{vm} en el acero medido en kN/m^2 de la simulación realizada con tetraedros.

También se realizó la simulación del problema en el programa ANSYS, en donde se utilizó una malla con 7616 nudos y 6125 elementos de ocho nudos, utilizando el elemento “SOLID185” de la librería de ANSYS, como se muestra en la [Figura 5.30](#) y en la [Figura 5.31](#). A diferencia del enmallado con elementos hexaédricos realizado por GMSH, este enmallado no utiliza prismas, sino que acomoda los hexaedros para que lleguen al centro del cilindro.

Se utilizó la opción de integración del elemento “Simple enhanced strain”, la cual presenta diferencias en la formulación con el elemento hexaédrico implementado en este trabajo, ya que agrega en algunos elementos 9 grados de libertad internos para evitar el error de bloqueo por cortante, tal como se menciona en la [Subsección 5.1.3](#).

Los desplazamientos en los nudos de la simulación realizada en el programa ANSYS se muestran en la [Figura 5.32](#). La distribución del esfuerzo principal menor σ_3 se muestra en la [Figura 5.33](#) y la distribución del esfuerzo de Von Mises σ_{vm} se muestra en la [Figura 5.34](#).

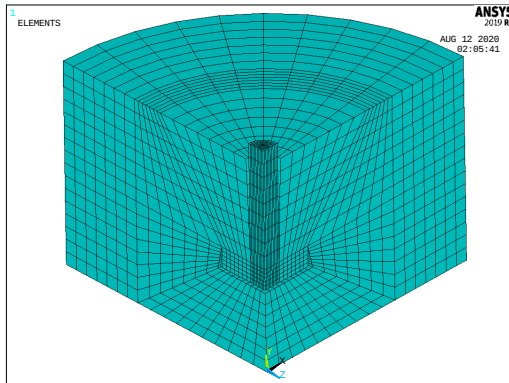


Figura 5.30: Malla de elementos finitos hexaédricos en el programa ANSYS.

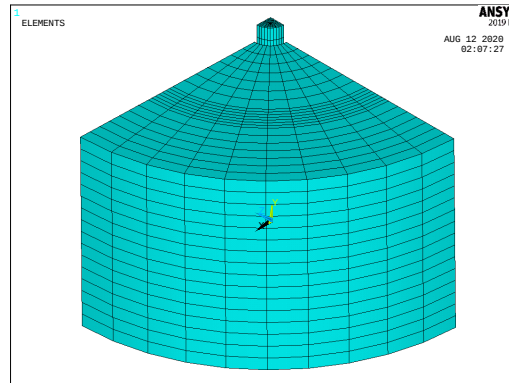


Figura 5.31: Malla de elementos finitos hexaédricos en el programa ANSYS.

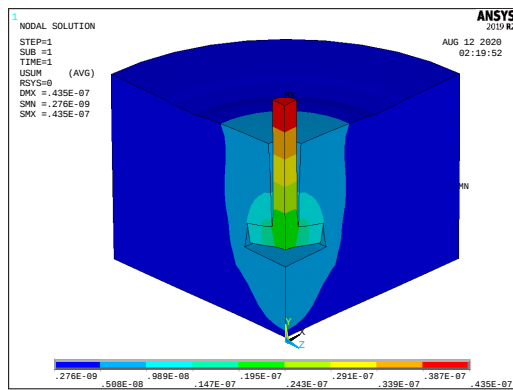


Figura 5.32: Configuración deformada y norma del vector de desplazamiento en metros de la simulación realizada en ANSYS

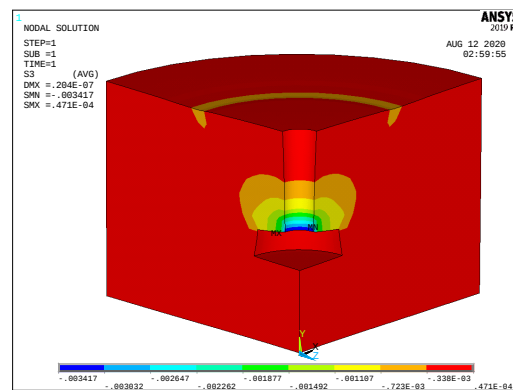


Figura 5.33: Distribución de esfuerzo principal menor σ_3 en el concreto, medido en kN/m^2 de la simulación realizada en ANSYS.

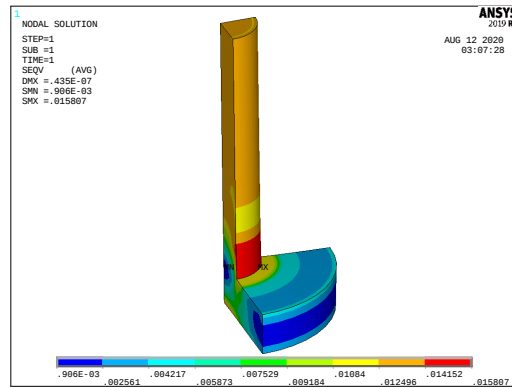


Figura 5.34: Distribución de esfuerzo de Von Mises σ_{vm} en el acero medido en kN/m^2 de la simulación realizada en ANSYS.

5.2.4. Análisis de resultados

A partir de las soluciones calculadas por las tres simulaciones, se comparan los valores de la norma del vector de deformación, del esfuerzo principal menor σ_3 en el concreto y del esfuerzo de Von Mises σ_{vm} en el acero. Estos valores se muestran en la [Tabla 5.2](#).

El valor máximo de la norma del vector de desplazamiento en los nudos que se utiliza como comparación ocurre en todos los casos en la zona superior de la barra de acero, como se puede observar en la [Figura 5.24](#), la [Figura 5.25](#) y en la [Figura 5.32](#).

Por otro lado, el valor del esfuerzo principal menor en el concreto que se compara en los tres casos, ocurre en la zona de contacto con el cabezal, como se puede observar en la [Figura 5.26](#), la [Figura 5.27](#) y en la [Figura 5.33](#).

El valor del esfuerzo de Von Mises en el acero σ_{vm} , que se compara en los tres casos, ocurre en la barra, cerca a la zona de unión con el cabezal, hacia el exterior del cilindro, como se puede observar en la [Figura 5.26](#), la [Figura 5.27](#) y en la [Figura 5.33](#).

Tabla 5.2: Comparación de resultados

Programa	Elemento	Máxima norma del desplazamiento (m)	Esfuerzo principal menor (kN/m^2)	Esfuerzo de Von Mises (kN/m^2)
aquí desarrollado	hexaedro de 8 nudos	4.31×10^{-6}	-3.33×10^{-3}	1.82×10^{-2}
PEFiCA 2.0	tetraedro de 4 nudos	4.08×10^{-6}	-2.48×10^{-3}	1.66×10^{-2}
ANSYS	hexaedro de 8 nudos con enriquecimiento de la deformación (SO-LID185)	4.35×10^{-6}	-3.41×10^{-3}	1.58×10^{-2}

De los datos recopilados en la [Tabla 5.2](#), se observa en los valores de la norma del vector de desplazamiento que la simulación con elementos hexaédricos en el programa desarrollado difiere un 5.3% con respecto a la simulación con la malla de elementos tetraédricos y un 0.9% en relación con el programa ANSYS.

También se observa que en los valores de esfuerzo principal menor σ_3 en la base de la presa, la simulación con elementos hexaédricos en el programa desarrollado difiere un 25.5% con respecto a la simulación con la malla de elementos tetraédricos y un 2.4% en relación con el programa ANSYS. Lo anterior indica una aproximación satisfactoria del programa desarrollado para elementos hexaédricos en comparación con ANSYS y un resultado aceptable de la simulación con elementos tetraédricos, recordando que el orden de aproximación del elemento hexaedro es mayor al tetraedro.

Con respecto a los valores de esfuerzo de Von Mises σ_{vm} en la unión entre la barra y el cabezal, la simulación con elementos hexaédricos en el programa desarrollado difiere un 8.8% con respecto a la simulación con la malla de elementos tetraédricos y un 13.1% en relación con el programa ANSYS.

5.2.5. Análisis de sensibilidad

Se realizó un análisis de sensibilidad de la densidad de la malla con elementos hexaédricos, simulando el mismo problema con 8 mallas de diferente cantidad de elementos.

En la [Figura 5.35](#) se indica el valor máximo absoluto para la malla en función de la cantidad de elementos finitos, correspondiente a la norma del desplazamiento (mm), el esfuerzo equivalente de von Mises en el acero (kN/m^2) y el esfuerzo principal menor en el concreto (kN/m^2). Allí se observa que los resultados son asintóticos con el incremento del número de elementos de la malla.

5.3. Análisis de costo computacional

A continuación se presenta la comparación del costo computacional asociada al tipo de elemento finito y a la función intrínseca de GNU-OCTAVE (compatible con MATLAB) utilizada para resolver el sistema de ecuaciones simultáneas.

5.3.1. Comparación del costo computacional por tipo de elemento

En la [Tabla 5.3](#) se indica el tiempo de ejecución de las simulaciones para los dos ejemplos de aplicación, con mallas de igual cantidad de nudos y con elementos finitos hexaédricos y tetraédricos, utilizando GNU-OCTAVE y MATLAB. Allí se observó que

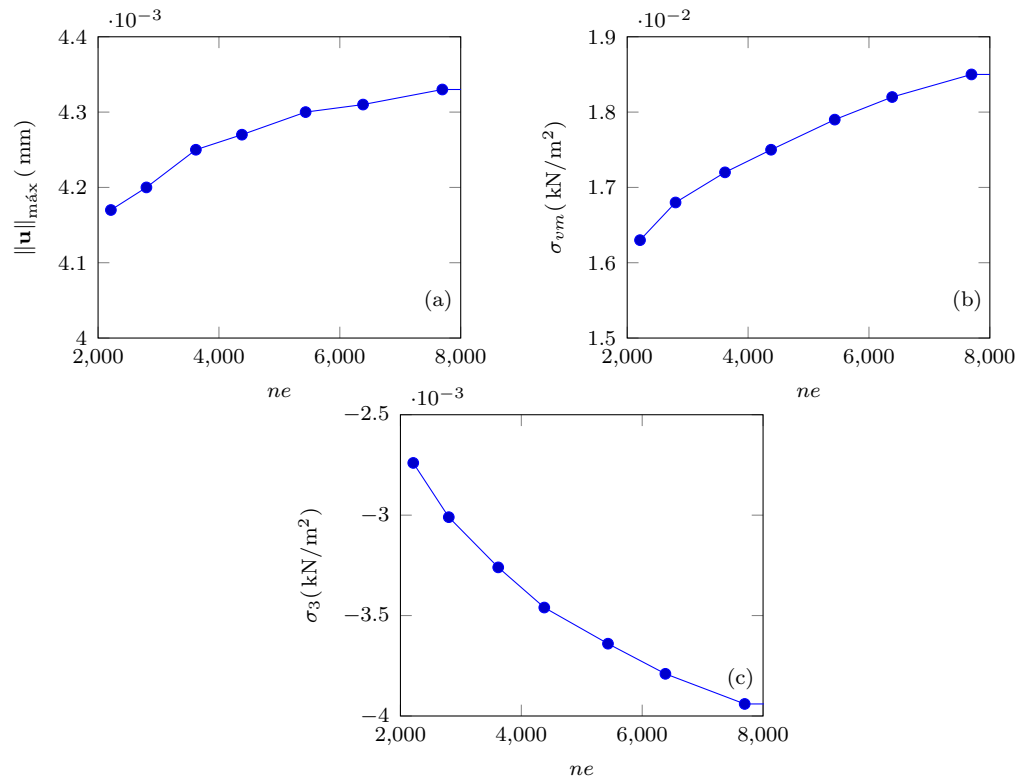


Figura 5.35: Valores máximos absolutos en la malla en función del número de elementos finitos de 8 mallas de elementos hexaédricos, correspondientes a: (a) la norma del desplazamiento de la malla (mm), (b) el esfuerzo equivalente de von Mises en el acero (kN/m²) y (c) el esfuerzo principal menor en el concreto (kN/m²)

Tabla 5.3: Tiempos de ejecución del programa desarrollado, en GNU OCTAVE y MATLAB, con elementos hexaédricos y tetraédricos

Simulación	Programa	Tipo de elemento finito	Tiempo de ejecución (s)
presa Hoover	GNU-OCTAVE	hexaedro de 8 nudos	659
presa Hoover	GNU-OCTAVE	tetraedro de 4 nudos	1455
presa Hoover	MATLAB	hexaedro de 8 nudos	166
presa Hoover	MATLAB	tetraedro de 4 nudos	168
pull out	GNU-OCTAVE	hexaedro de 8 nudos	565
pull out	GNU-OCTAVE	tetraedro de 4 nudos	1068
pull out	MATLAB	hexaedro de 8 nudos	190
pull out	MATLAB	tetraedro de 4 nudos	196

el tiempo de ejecución con la malla de elementos hexaédricos es aproximadamente un 51 % menor que con la malla de elementos tetraédricos utilizando GNU-OCTAVE y un 2 % utilizando MATLAB. Lo anterior demuestra la eficiencia del elemento hexaédrico de 8 nudos con respecto al elemento tetraédrico lineal de 4 nudos.

Tabla 5.4: Tiempos de ejecución del programa en MATLAB utilizando la sintaxis indicada y el comando `linsolve` con la opción `SYM`, en una simulación con 8862 nudos, en las diferentes etapas del proceso de análisis. Tiempos en segundos (s)

Solucionador	T1	T2	T3	T4	T5	T6	T7	Tiempo total
$UAA = (KAA) (FAA - KAB * UBB)$	1.0	0.3	42.9	0.9	94.4	3.0	2.0	144.4
<code>linsolve</code> con opción <code>SYM</code>	0.8	0.2	47.4	0.9	79.4	2.8	2.0	133.5

Tabla 5.5: Tiempos de ejecución del programa en GNU-OCTAVE utilizando la sintaxis indicada y el comando `linsolve` con la opción `SYM`, en una simulación con 8862 nudos, en las diferentes etapas del proceso de análisis. Tiempos en segundos (s)

Solucionador	T1	T2	T3	T4	T5	T6	T7	Tiempo total
$UAA = (KAA) (FAA - KAB * UBB)$	90.8	3.7	198.2	22.3	128.6	65.8	47.2	556.6
<code>linsolve</code> con opción <code>SYM</code>	89.7	3.7	197.2	22.3	122.1	65.2	47.1	547.4

5.3.2. Comparación del costo computacional de acuerdo al método de solución de ecuaciones simultáneas

La formulación del problema mecánico elástico tridimensional conduce a calcular los desplazamientos en los nudos de la malla, mediante la resolución de un sistema de ecuaciones simultáneas. Este último está definido por una matriz de rigidez simétrica del sólido. En los métodos de solución de sistemas de ecuaciones simultáneas incluidos en las funciones intrínsecas de GNU-OCTAVE, existe uno aplicable a matrices simétricas, el cual requiere menos de operaciones que los métodos para matrices no simétricas.

La implementación de la función `linsolve`, propia del lenguaje de GNU-Octave (compatible con MATLAB) mencionada en la [Subsección 4.5.1](#), redujo el tiempo de ejecución de la solución del sistema de ecuaciones simultáneas. Los tiempos de ejecución de la tarea relacionada con la solución del sistema de ecuaciones se redujeron alrededor de 15.8 % en MATLAB y 6.6 % en GNU-OCTAVE. Los tiempos totales en la ejecución del programa se redujeron alrededor de 7.5 % en MATLAB y 2.0 % en GNU-OCTAVE, como se muestra en la [Tabla 5.4](#) y en la [Tabla 5.5](#).

En las tablas mencionadas, T1 representa la etapa de la lectura de datos, T2 es la designación de los grados de libertad, T3 es el cálculo de la matriz de rigidez, T4 es la obtención del vector de fuerzas, T5 es el cálculo de desplazamientos y reacciones, T6 el cálculo de las deformaciones y esfuerzos y T7 la presentación de resultados. La tarea en donde se utiliza el solucionador del sistema de ecuaciones es T5.

Capítulo 6

Conclusiones y recomendaciones

A continuación se presentan las conclusiones de este trabajo final de maestría y las recomendaciones para futuros trabajos de investigación derivados del programa desarrollado.

6.1. Conclusiones

Como conclusión general, la formulación del método de los elementos finitos para el análisis tridimensional estático de sólidos con material elástico lineal, se puede implementar utilizando elementos hexaédricos de 8 nudos y prismas de 6 nudos, en lenguaje interpretado GNU-OCTAVE y MATLAB. Asimismo la programación estructurada de dicha formulación, la elaboración del manual de usuario y la representación gráfica del pre y del post proceso, contribuyen potencialmente al aprendizaje del método de los elementos finitos en la mecánica de sólidos. A continuación se presentan las conclusiones específicas:

- El programa desarrollado permite la lectura de los datos del problema generados previamente en el entorno gráfico de preproceso y la escritura de los resultados en el formato de representación gráfica en el entorno de postproceso. Lo anterior se aplica particularmente para mallas de elementos finitos hexaédricos y prismas.
- El campo del desplazamiento, el esfuerzo y la deformación de un sólido bajo la acción de cargas estáticas, se puede obtener utilizando el programa desarrollado, mediante el método de los elementos finitos y utilizando elementos hexaédricos y prismas. Los resultados del análisis de: una presa de bóveda bajo el peso propio y la presión hidroestática y un ensayo de arrancamiento de una

barra de acero en un cilindro de mortero, son sustancialmente similares a los obtenidos con un programa comercial reconocido.

- En las simulaciones realizadas se observó que la diferencia del desplazamiento y de los esfuerzos principales entre la malla de elementos hexaédricos y tetraédricos, se debe al orden de aproximación y a la mayor rigidez que ofrece un volumen conformado por este último tipo de elemento.
- Los resultados representativos de las simulaciones realizadas para diferentes densidades de malla, demostraron la convergencia hacia un valor común.
- La forma de escribir el código del programa desarrollado en un lenguaje como GNU-OCTAVE o MATLAB, mantiene la robustez del proceso de cálculo, y potencialmente facilita la comprensión de los procedimientos y la posibilidad de ampliación mediante nuevas subrutinas.
- La simetría de la matriz de rigidez de la malla en problemas elásticos lineales tridimensionales, permitió utilizar un método de solución del sistema de ecuaciones simultáneas particular para matrices simétricas. Durante la ejecución del programa se observó que este último, calculó los desplazamientos nodales de la malla en menos tiempo, en comparación con el método para matrices no simétricas.
- El tiempo de ejecución de la simulación de los ejemplos de aplicación con la malla de elementos hexaédricos, fue menor que aquella con elementos tetraédricos conservando el mismo número de nudos. Lo anterior demuestra la eficiencia del elemento hexaédrico de 8 nudos con respecto al elemento tetraédrico lineal de 4 nudos.

6.2. Recomendaciones para futuros trabajos

A continuación se presentan algunas sugerencias para futuros trabajos que se pueden realizar a partir del presente trabajo final de maestría:

- Para evitar el error denominado bloqueo por cortante, se podría implementar el elemento finito hexaédrico con enriquecimiento de la deformación, el cual está incluido en la librería de elementos finitos de ANSYS. Esta técnica introduce selectivamente nuevos grados de libertad y está basada en los trabajos desarrollados de Simo y Rifai (Simo & Rifai, 1990).

- Con el propósito de mejorar la precisión de la integración numérica del elemento hexaédrico de ocho nudos, se propone implementar dicha integración con un número mayor a ocho puntos de Gauss sobre el elemento.
- Para mejorar la precisión de la solución del problema se pueden implementar elementos finitos tetraédricos y hexaédricos de segundo orden.
- Con el fin de utilizar programas de generación de mallas que producen elementos finitos piramidales, se recomienda la implementación de este tipo de elemento.
- Se recomienda desarrollar los algoritmos de análisis dinámico y no lineal para solucionar problemas mecánicos tridimensionales. Lo anterior con el fin de acercarse al comportamiento real de los sólidos.

Capítulo 7

Manual del usuario del programa

En este capítulo se describen los principales aspectos para utilizar el programa desarrollado en este trabajo final de maestría, para resolver problemas mecánicos tridimensionales. Tal como se mencionó en el procedimiento general presentado en la [Sección 4.1](#), el proceso consta de tres partes principales:

- El preproceso o la construcción del archivo que contiene las entidades geométricas y las características físicas, utilizando el programa gratuito GMSH.
- El proceso o la solución al problema mecánico tridimensional mediante las rutinas modificadas y generadas en este trabajo final de maestría, en el programa GNU-OCTAVE.
- El post proceso o la representación de los resultados en el dominio del problema, utilizando el módulo de post proceso del programa GMSH.

Las siguientes secciones contienen un breve manual de usuario para cada una de las partes mencionadas. Una descripción más detallada de cada una de éstas se puede consultar en la referencia (Lineró *et al.*, 2020).

7.1. Creación de la geometría y el enmallado en el programa GMSH

En esta sección se describe el procedimiento para crear la geometría, la designación de características físicas y la malla de elementos finitos hexaédricos y prismas desde el módulo gráfico de preproceso del programa GMSH.

El enmallado con elementos hexaédricos descrito en este trabajo, se logra a través de un proceso de recombinación de una malla estructurada de elementos tetraédricos.

Este enmallado con elementos finitos hexaédricos comparte algunas de las etapas utilizadas para construir mallas de elementos tetraédricos mostrado en la referencia (Linero *et al.*, 2020), pero adicionalmente se deben cumplir ciertas condiciones.

Estas condiciones son debidas a los algoritmos de recombinación de elementos, propios del programa GMSH. Las condiciones son:

- Las superficies que conforman los sólidos deben estar delimitadas por cuadriláteros. En caso de requerirse formas complejas, se pueden realizar divisiones a las superficies de manera que se formen cuadriláteros, como se mostró para los ejemplos en la [Figura 5.2](#) y [Figura 5.18](#).
- Todas las líneas que conforman cada uno de los volúmenes del sólido debe tener asignado un parámetro llamado **Transfinite Curve**, el cual establece la cantidad de las divisiones que la línea tendrá cuando se realice el enmallado. Cuanto mayor sea su valor, más elementos finitos tendrá la futura malla.
- Los valores de la propiedad **Transfinite Curve** asignados a las líneas del problema deben ser consistentes en la geometría considerando la futura malla de elementos. Esto quiere decir que si se desea, por ejemplo, asignarle a una cara de un cuadrilátero m divisiones en la malla, la cara opuesta también deberá tener m divisiones. Esto se ilustra en la [Figura 7.1](#) y en la [Figura 7.2](#).

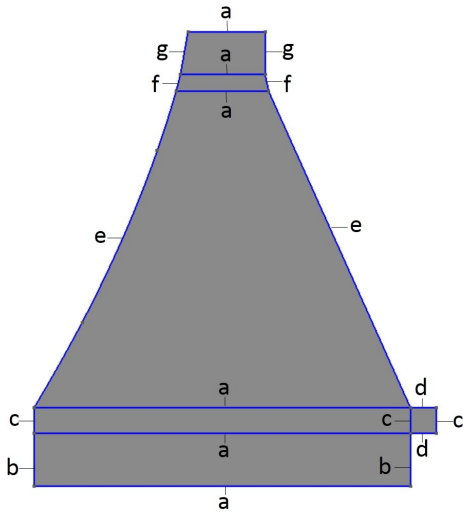


Figura 7.1: Asignación de la propiedad **Transfinite Curve** a las líneas de la sección. Las líneas con letras iguales representan las que deben tener el mismo valor de esta propiedad.

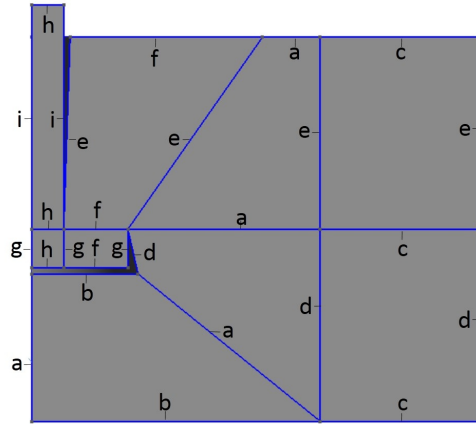


Figura 7.2: Asignación de la propiedad **Transfinite Curve** a las líneas de la sección. Las líneas con letras iguales representan las que deben tener el mismo valor de esta propiedad.

- Todas las superficies que conforman los volúmenes del sólido del problema, deben también tener asignada la propiedad **Transfinite Surface**. Esta propiedad aplicada a las superficies, define el orden de sus vértices, los cuales se deben seleccionar de forma consecutiva.
- Todos los volúmenes que conforman el sólido del problema, también deben tener asignada la propiedad **Transfinite Volume**. En esta propiedad aplicada en los volúmenes, se deben definir los vértices que lo conforman.
- Una vez cumplidos los anteriores requisitos, se debe dar la instrucción de convertir la malla de elementos tetraédricos en elementos hexaédricos, a través de la instrucción **Recombine Surface**.

Por otro lado, en los enmallados generados por GMSH los elementos prismas se utilizan solamente como elementos adicionales a los hexaedros para cubrir geometrías con características muy específicas, como por ejemplo la zona cercana al eje de la cuarta parte del cilindro mostrado en la [Sección 5.2](#).

Para una descripción mas detallada del procedimiento para la generación de mallas con elementos tetraédricos, así como para una ampliación a la información presentada en este capítulo, se puede consultar el capítulo 10 de la referencia (Linero

et al., 2020) o el manual de usuario del programa GMSH (Geuzaine & Remacle, 2019).

7.1.1. Consideraciones generales

El programa GMSH permite construir la geometría y asignarle las características físicas al problema mediante órdenes ejecutadas a través de los botones que presenta su interfaz gráfica, o también mediante la creación o modificación de un archivo en formato ASCII de extensión `.geo`. En este archivo se almacena la secuencia de comandos que forma la geometría del problema y la asignación de las características físicas y de su enmallado.

Es posible crear la geometría del problema a través de la modificación directa de archivo `.geo` o mediante los comandos de los botones en pantalla o utilizando las dos opciones, según convenga.

Cada orden realizada mediante las opciones del entorno gráfico para la creación de la malla, tendrá una línea en el archivo `.geo` asociada. Este archivo se puede consultar o modificar en cualquier momento, y resulta útil cuando se requiere, por ejemplo, deshacer la última acción realizada. Para esto basta borrar la última línea del archivo `.geo` y volver a cargarlo.

El archivo `.geo` se puede abrir a través del menú lateral de GMSH haciendo click en **Modules** ▷ **Geometry** ▷ **Edit script**. Al realizar modificaciones directamente en él, se deben guardar los cambios.

Si se desea volver a cargar el archivo `.geo`, se hace clic en **Modules** ▷ **Geometry** ▷ **Reload script**.

Para iniciar un nuevo archivo, se hace clic en **File** ▷ **New**, a continuación se selecciona la carpeta de trabajo y se escribe el nombre del archivo con extensión `.geo`.

En seguida se selecciona el tipo de librerías internas para la visualización. Para este tipo de problemas es conveniente seleccionar **OpenCASCADE**.

Por otro lado, en la mayoría de los cuadros de diálogo que genera el programa GMSH, la tecla de confirmación es **e**, mientras que la tecla para terminar es **q**.

7.1.2. Construcción de entidades geométricas

En el entorno gráfico de GMSH se pueden definir los puntos, líneas rectas o curvas, superficies y volúmenes que conforman la geometría del problema. Todas estas instrucciones tienen asociada una instrucción en el archivo de extensión `.geo`. En la [Figura 7.3](#) se muestran las opciones para la construcción de entidades del entorno gráfico de GMSH.

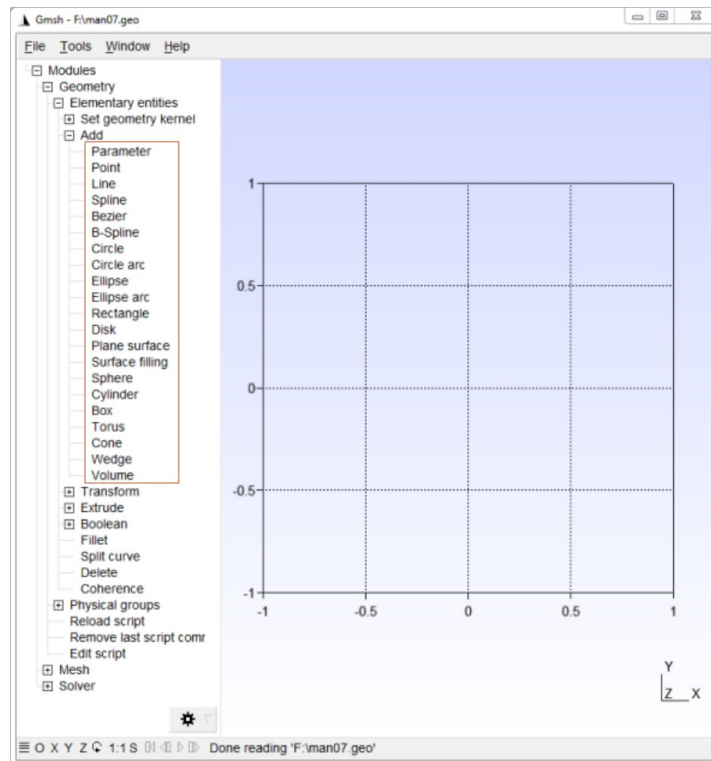


Figura 7.3: Opciones de construcción de entidades del entorno gráfico de GMSH.

7.1.2.1. Construcción de puntos

Para crear un punto se selecciona en el menú lateral **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Point**. A continuación se introducen las coordenadas del punto en la pestaña **Point** de la ventana **Elementary Entity Context** y se hace click en **Add**.

También es posible indicar gráficamente la posición del punto, indicándola con el puntero sobre la cuadrícula. Para modificar la precisión del puntero sobre la cuadrícula, se modifican los parámetros **X Y Zsnap** que aparecen en la ventana **Elementary Entity Context**.

Cada vez que se crea un nuevo punto, en el archivo de extensión **.geo** se adiciona una línea con la sintaxis que se muestra a continuación:

$$\text{Point}(ipunto) = \{xcoord, ycoord, zcoord, tamalla\}$$

En la anterior instrucción, *ipunto* es el identificador del punto, *xcoord*, *ycoord* y *zcoord* son las coordenadas x,y,z del punto y *tamalla* define el tamaño de los

elementos finitos sobre el punto. El parámetro *tamalla* en el caso de la construcción de mallas con elementos hexaédricos utilizando la instrucción **Transfinite**, no será tenido en cuenta, por lo que se puede dejar con valor igual a cero.

7.1.2.2. Construcción de líneas

Luego de definir los puntos, se construyen las líneas seleccionando en el menú **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Line**. A continuación se hace clic sobre el punto inicial de la línea y luego clic sobre el punto final. Se puede continuar creando nuevas líneas o terminar presionando la tecla **q**.

La instrucción generada en el archivo con extensión **.geo** es:

$$\text{Line}(ilinea) = \{inudo, jnudo\}$$

donde *ilinea* es el identificador de la línea, *inudo* es el identificador del nudo inicial y *jnudo* es el identificador del nudo final.

Adicionalmente, se pueden construir otros tipos de líneas curvas seleccionando **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ y a continuación seleccionar entre:

- **Spline** crea una línea curva que pasa por los puntos seleccionados. Su sintaxis es:

$$\text{Spline}(ilinea) = \{inudo, jnudo, knudo \dots\}$$

- **Bezier** crea una curva de Bézier, la cual es una curva que une el primero y el último punto seleccionado, y utiliza los puntos intermedios como puntos de control de su curvatura. Su sintaxis es:

$$\text{Bezier}(ilinea) = \{inudo, jnudo, knudo \dots\}$$

- **B-Spline** crea una línea que une el primer y el último punto, utilizando los puntos intermedios como control. Para ello utiliza una aproximación cúbica. Su sintaxis es:

$$\text{BSpline}(ilinea) = \{inudo, jnudo, knudo \dots\}$$

- **Circle** crea un arco circular. En su sintaxis se indican las coordenadas x, y, z del punto de inicio, el radio, el ángulo inicial y ángulo final del arco. Su sintaxis es de la forma:

$$\text{BSpline}(ilinea) = \{coorxi, cooryi, coorzi, radio, angi, angj\}$$

- **Ellipse** crea un arco de elipse. En su sintaxis se indican las coordenadas x, y, z del punto de inicio, el radio en x, el radio en y, el ángulo inicial y ángulo final del arco. Su sintaxis es de la forma:

$$\text{BSpline}(ilinea) = \{coorxi, cooryi, coorzi, radiox, \dots \\ radioy, anguli, angulj\}$$

7.1.2.3. Construcción de superficies

Para construir una superficie se selecciona **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Plane surface**. A continuación se da clic en las líneas rectas o curvas que forman el contorno de la superficie, y una vez seleccionadas todas las líneas se oprime la tecla **e**. Para finalizar la creación de superficies se presiona la tecla **q**.

Por cada superficie creada, en el archivo con extensión **.geo** se adicionan dos instrucciones:

$$\text{CurveLoop}(icurciclo) = \{ilinea, jlinea, klinea \dots\}$$

$$\text{PlaneSurface}(isuperf) = \{icurciclo\}$$

La primera instrucción establece un grupo de líneas conectadas por sus extremos, formando un ciclo de curvas. En ella *icurciclo* es el identificador del ciclo de curvas, y *ilinea, jlinea, ...* representan los identificadores de las líneas que conforman el ciclo.

La segunda instrucción establece la superficie a partir del ciclo de curva. En ella *isuperf* es el identificador de la superficie y el parámetro *icurciclo* es el identificador del ciclo correspondiente.

7.1.2.4. Construcción de volúmenes

La definición de los volúmenes se realiza seleccionando **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Volume**. A continuación se da clic en las superficies que conforman el contorno del volumen y luego se oprime la tecla **e**. Para finalizar la creación de volúmenes se presiona la tecla **q**.

Por cada volumen creado, se adicionan dos instrucciones al archivo de extensión **.geo** :

$$\text{SurfaceLoop}(isupciclo) = \{isup, jsup, ksup \dots\}$$

$$\text{PlaneSurface}(isuperf) = \{isupciclo\}$$

La primera instrucción establece un grupo de superficies conectadas por sus lados formando un ciclo de superficies. En ella *isupciclo* es el identificador del ciclo de superficies, y *isup*, *jsup*, ... representan los identificadores de las superficies que conforman el ciclo. La segunda instrucción establece el volumen a partir del ciclo de superficies. En ella *ivol* es el identificador del volumen y el parámetro *isupciclo* es el identificador del ciclo correspondiente.

7.1.3. Construcción de formas comunes

El programa GMSH permite además construir directamente formas geométricas, ya sean superficies o volúmenes, generando de manera automática los puntos, líneas o superficies requeridos. Algunas de éstas formas se muestran a continuación:

7.1.3.1. Creación de un rectángulo

Se puede crear desde **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Rectangle**. Esto abre una ventana desde donde se introducen las coordenadas x, y, z del extremo inferior izquierdo, así como su base y altura, representados como **DX** y **DY** en la ventana de creación desplegada. Una vez se definen éstos parámetros, se da clic en el botón **Add**. Para finalizar se presiona la tecla **q**.

Esta instrucción genera una línea al archivo con extensión **.geo** de la forma:

$$\text{Rectangle}(idsu) = \{extx, exty, extz, disx, disy, jnudo\}$$

donde *idsu* es el número identificador de superficie, *extx*, *exty*, *extz* son las coordenadas del extremo inferior izquierdo, *disx*, *disy* son la base y la altura del rectángulo y *rede* es un parámetro para redondear las esquinas.

7.1.3.2. Creación de un círculo o elipse

Se puede crear desde **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Disk**. Esto abre una ventana donde se introducen las coordenadas x, y, z del centro del círculo o elipse, así como su radio en X y radio en Y. Si el valor del radio en X es igual al radio en Y, se define un círculo, en caso contrario se define una elipse. Una vez se definen éstos parámetros, se da clic en el botón **Add**. Para terminar se presiona la tecla **q**.

Esta instrucción genera una línea al archivo con extensión `.geo` de la forma:

$$\text{Disk}(idsu) = \{cenx, ceny, cenx, radx, rady\}$$

donde *idsu* es el número identificador de superficie, *cenx*, *ceny*, *cenx* son las coordenadas del centro, *radx*, *rady* son los radios en dirección *x* y *y*.

7.1.3.3. Paralelepípedo regular

Para crearlo, se selecciona **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Box**. Esto despliega una ventana donde se introducen las coordenadas *x*, *y*, *z* del extremo inferior izquierdo del paralelepípedo, así como sus dimensiones en *x*, *y*, *z*, representados como **DX**, **DY** y **DZ** en la ventana desplegada. Una vez se han definido éstos parámetros, se da clic en el botón **Add**. Para terminar se presiona la tecla **q**.

Esta instrucción genera una línea al archivo con extensión `.geo` de la forma:

$$\text{box}(idvo) = \{extx, exty, extz, disx, disy, disz\}$$

donde *idvo* es el número identificador de volumen, *extx*, *exty*, *extz* son las coordenadas del extremo inferior izquierdo y *disx*, *disy*, *disz* son las dimensiones *x*, *y*, *z* del paralelepípedo.

7.1.3.4. Creación de un cilindro

Se puede crear desde **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Add** ▷ **Cylinder**. Esto despliega una ventana donde se introducen las coordenadas *x*, *y*, *z* del centro del círculo de la base en las casillas **Center base X**, **Center base Y** y **Center base Z**. Asimismo se introducen sus proyecciones en *X*, *Y* y *Z* en las casillas **Axis DX**, **Axis DY** y **Axis DZ**.

El radio del círculo de la base se ingresa en la casilla **Radius**. En la casilla **Angle** se define la porción del cilindro que se desea crear en radianes, siendo 2π el círculo completo. valores menores a éste generan la extrusión del sector del círculo indicado. Una vez se definen éstos parámetros, se da clic en el botón **Add**. Para terminar se presiona la tecla **q**.

Esta instrucción genera una línea al archivo con extensión `.geo` de la forma:

$$\text{Cylinder}(idvo) = \{cenx, ceny, cenx, axdx, axdy, axdz, radi, angl\}$$

donde *idvo* es el número identificador de volumen, *cenx*, *ceny*, *cenx* son las coordenadas del centro del círculo de la base, *axdx*, *axdy*, *axdz* son las proyecciones del eje longitudinal *x*, *y* y *z*. El parámetro *radi* es el radio del círculo de la base y el *angl* es el ángulo en radianes del sector circular.

7.1.4. Construcción de entidades geométricas mediante extrusión

Los procesos de extrusión que se describen en esta sección permiten generar un volumen a partir de la rotación o traslación de una o varias superficies.

7.1.4.1. Creación de volúmenes a partir de la extrusión por traslación de superficies

Para generar el volumen por traslación de superficies, se selecciona **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Extrude** ▷ **Traslate**. Lo anterior despliega una ventana donde se debe indicar la longitud de la extrusión por traslación para cada eje x, y, z , en las casillas **DX**, **DY** y **DZ** respectivamente.

A continuación en **Selection mode** se escoge **Surfaces**. Luego en el entorno gráfico se seleccionan las superficies y una vez seleccionadas se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Al realizar lo anterior, en el archivo de extensión **.geo** se genera una instrucción que tiene la forma:

$$\text{Extrude}\{dx, dy, dz\}\{\text{Surface}\{isup1, isup2, \dots\}; \dots\}$$

donde dx, dy y dz son las longitudes de la extrusión en direcciones x, y y z . $isup1, isup2, \dots$ son los identificadores de las superficies que serán extruidas.

7.1.4.2. Creación de volúmenes a partir de la extrusión por rotación de superficies

Para generar el volumen por rotación de superficies, se selecciona **Modules** ▷ **Geometry** ▷ **Elementary entities** ▷ **Extrude** ▷ **Rotate**. Se despliega una ventana donde se debe indicar las coordenadas x, y y z del origen del eje que define la rotación. Estas coordenadas x, y y z se introducen en las casillas **Axis point x**, **Axis point y** y **Axis point z**.

También es necesario definir las componentes x, y y z que definen la dirección del eje de rotación. Esto se hace en las casillas **Axis DX**, **Axis DY** y **Axis DZ**. A continuación se define el ángulo en radianes de la extrusión en la casilla **Angle**.

Una vez definidas los anteriores parámetros, en **Selection mode** se escoge **Surfaces**. Luego en el entorno gráfico se seleccionan las superficies y una vez seleccionadas se oprime la tecla **e**. Para finalizar se oprime la tecla **q**. De forma automática, en el archivo de extensión **.geo** se agrega una instrucción que tiene la forma:

$$\text{Extrude}\{\{dx, dy, dz\}, \{xp, yp, zp\}, ang\}\{\text{Surface}\{isup1, isup2, \dots\}; \dots\}$$

donde dx , dy y dz son componentes que definen la dirección del eje de rotación, xp , yp y zp son las coordenadas del punto inicial del eje de rotación, ang es el ángulo en radianes de la extrusión, y $isup1$, $isup2$... son los identificadores de las superficies que serán extruídas. La ubicación de estos comandos en el entorno gráfico de GMSH se muestra en la Figura 7.4 y la ventana de selección de parámetros para extrusión por rotación se muestra en la Figura 7.5.

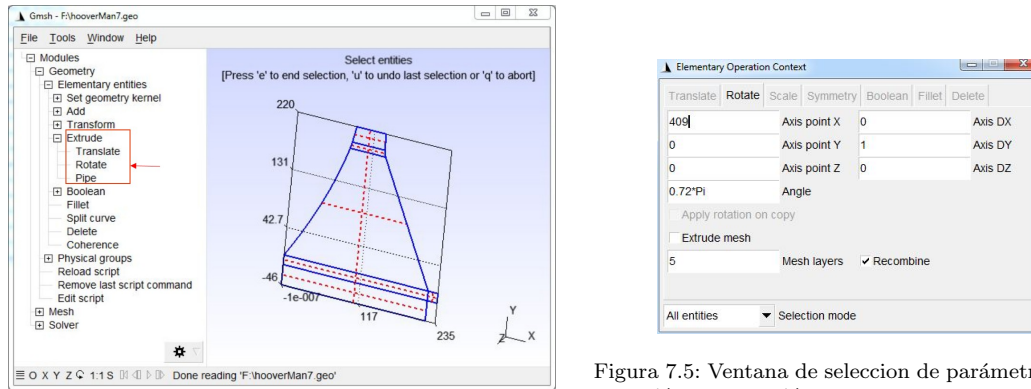


Figura 7.4: Ubicación de los comandos de extrusión por rotación.

Figura 7.5: Ventana de selección de parámetros para extrusión por rotación

7.1.5. Asignación de entidades físicas para el proceso en PEFiCA

En esta sección se indica la forma de asignar las propiedades mecánicas de los materiales, la definición de desplazamientos, la aplicación de cargas puntuales y las cargas distribuidas a los elementos de la geometría.

7.1.5.1. Propiedades mecánicas de los materiales

Las propiedades mecánicas de los materiales se asignan a los volúmenes. Es posible tener varios materiales dentro del dominio del problema, pero cada uno de los volúmenes que conforman el dominio del problema solo puede estar asociado a un solo tipo de material.

Para asignarlas, se debe desplegar la ventana **Physical groups** relacionada con volúmenes, seleccionando **Modules** \triangleright **Geometry** \triangleright **Physical groups** \triangleright **Add** \triangleright **Volume**.

Allí se asignan las propiedades del material al volumen mediante la siguiente sintaxis:

$$\text{CATE EYOU} = \text{eyou POIS} = \text{pois Gamm} = \text{gamm TIPR} = \text{tipr}$$

donde *eyou* es el módulo de Young, *pois* es la relación de Poisson, *gamm* es el peso específico y *tipr* es el tipo de problema, el cual para este tipo de problemas tridimensionales es $\text{TIPR} = 30$. No debe hacer espacios antes o después del caracter =.

A continuación se seleccionan los volúmenes que están asociados al material y se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Cuando se utiliza el procedimiento anterior, en el archivo de extensión **.geo** se agrega una instrucción que tiene la forma:

$$\text{Physical Volume}(\text{"CATE EYOU} = \text{eyou POIS} = \text{pois} \\ \text{Gamm} = \text{gamm TIPR} = \text{tipr"}) = \{ivol, jvol, \dots\};$$

donde *ivol*, *jvol* . . . son los identificadores de los volúmenes a los que se le asignará el material.

7.1.5.2. Definición de desplazamientos

Para asignar los desplazamientos conocidos del problema sobre superficies, se debe desplegar la ventana **Physical groups** relacionada con superficies, seleccionando **Modules** ▷ **Geometry** ▷ **Physical groups** ▷ **Add** ▷ **Surface**.

Si el desplazamiento conocido está ubicado en puntos, se debe desplegar la ventana **Physical groups** relacionada con puntos, seleccionando **Modules** ▷ **Geometry** ▷ **Physical groups** ▷ **Add** ▷ **Point**.

En la ventana desplegada se asignan los valores del desplazamiento con la siguiente sintaxis:

$$\text{DISP UX} = ux \text{ UY} = uy \text{ UZ} = uz$$

donde *ux*, *uy* y *uz* son los valores conocidos de las componentes de desplazamiento en direcciones *x*, *y* y *z*. Solo se debe indicar la componente de valor conocido. Si por ejemplo en una superficie sólo se conoce el valor del desplazamiento en *y*, la sintaxis es: $\text{DISP UY} = uy$. No deben haber espacios antes o después del caracter =.

A continuación se seleccionan las superficies o puntos que están asociados al material y se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Al seguir los pasos anteriores, en el archivo de extensión **.geo** se agrega una instrucción que tiene la forma:

$$\text{PhysicalSurface}(\text{"DISP UX} = ux \text{ UY} = uy \text{ UZ} = uz") = \{isup, jsup, \dots\};$$

donde $isup, jsup \dots$ son los identificadores de superficie a las que se les asignó el desplazamiento conocido.

Si el desplazamiento se asigna a puntos, la instrucción es que se agrega es:

$$\text{PhysicalPoint}(\text{"DISP UX} = ux \text{ UY} = uy \text{ UZ} = uz\text{"}) = \{ipun, jpun, \dots\};$$

donde $ipum, jpum \dots$ son los identificadores de los puntos.

7.1.5.3. Aplicación de cargas puntuales

Las cargas puntuales se pueden asignar sobre puntos o superficies. Para asignarle una carga puntual a una superficie, se debe desplegar la ventana **Physical groups** relacionada con superficies, seleccionando **Modules** ▷ **Geometry** ▷ **Physical groups** ▷ **Add** ▷ **Surface**.

Si la carga puntual está ubicada en puntos, se debe desplegar la ventana **Physical groups** relacionada con puntos, seleccionando **Modules** ▷ **Geometry** ▷ **Physical groups** ▷ **Add** ▷ **Point**.

En esta ventana se debe escribir la carga puntual con la siguiente sintaxis:

$$\text{LOAD FX} = fx \text{ FY} = fy \text{ FZ} = fz$$

donde fx, fy y fz son los valores de las componentes de la carga puntual en direcciones x, y y z . Si se aplica una sola componente, no es necesario escribir las que son iguales a cero. No deben haber espacios antes o después del carácter $=$.

En seguida se seleccionan las superficies o puntos que están asociados al material y se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Lo anterior genera en el archivo de extensión **.geo** una instrucción que tiene la forma:

$$\text{PhysicalSurface}(\text{"LOAD FX} = fx \text{ FY} = fy \text{ FZ} = fz\text{"}) = \{isup, jsup, \dots\};$$

donde $isup, jsup \dots$ son los identificadores de superficie a las que se les asignó el desplazamiento conocido.

Si la carga se asigna a puntos, la instrucción es que se agrega es:

$$\text{PhysicalPoint}(\text{"LOAD FX} = fx \text{ FY} = fy \text{ FZ} = fz\text{"}) = \{ipun, jpun, \dots\};$$

donde $ipum, jpum \dots$ son los identificadores de los puntos.

7.1.5.4. Aplicación de presiones

Para asignar las presiones conocidas del problema sobre superficies, se debe desplegar la ventana **Physical groups** relacionada con superficies, seleccionando **Modules** ▷ **Geometry** ▷ **Physical groups** ▷ **Add** ▷ **Surface**.

En esta ventana se debe escribir, dependiendo del tipo de presión la siguiente sintaxis:

- Si la presión es uniforme con componentes definidos en el sistema global de coordenadas:

$$\text{PRES WX} = wx \text{ WY} = wy \text{ WZ} = wz$$

donde wx , wy y wz son los valores de las componentes de presión uniforme en dirección x , y y z . No deben haber espacios antes o después del caracter $=$.

- Si la presión es uniforme y normal a la superficie:

$$\text{PRES WN} = wn$$

donde wn es el valores de la presión uniforme normal a la cara en la que actúa, considerada positiva si entra a la superficie.

- Si se aplica una presión hidráulica:

$$\text{PRES GAWA} = gawa \text{ HEWA} = hewa$$

donde $gawa$ es el peso específico del agua γ_w y $hewa$ es el nivel del agua con respecto al origen coordenado global h_w .

A continuación se seleccionan las superficies que están asociados a la presión y se oprime la tecla **e**. Para finalizar se oprime la tecla **q**. En la [Figura 7.6](#) se muestra un ejemplo de asignación de una presión hidroestática. El procedimiento anterior genera en el archivo de extensión `.geo` una instrucción que tiene la forma, según corresponda a los casos anteriores:

- Para presión uniforme en coordenadas globales:

$$\text{PhysicalSurface}(\text{"PRES WX} = wx \text{ WY} = wy \text{ WZ} = wz\text{"}) = \{isup, jsup, \dots\};$$

- Para presión uniforme normal a al superficie:

$$\text{PhysicalSurface}(\text{"PRES WN} = wn\text{"}) = \{isup, jsup, \dots\};$$

- Para presión hidroestática:

$$\text{PhysicalSurface}(\text{"PRES GAWA} = gawa \text{ HEWA} = hewa") = \{isup, jsup, \dots\};$$

donde $isup, jsup \dots$ son los identificadores de superficie a las que se les asignó el desplazamiento conocido.

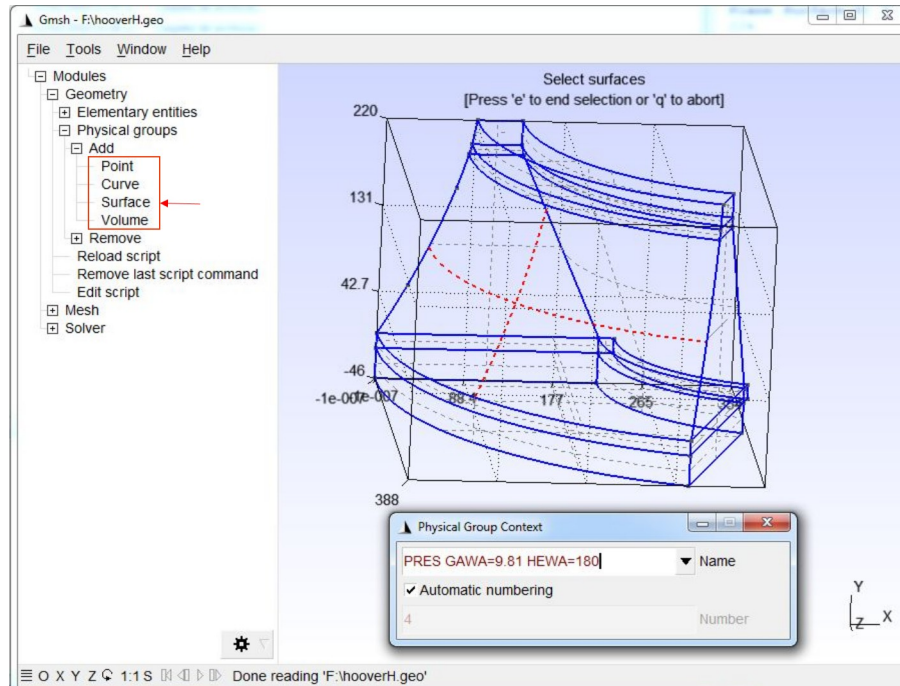


Figura 7.6: Ejemplo de asignación de una presión hidroestática sobre una superficie.

7.1.6. Preparación de entidades geométricas para el enmallado

Como se mencionó al principio de esta sección, para lograr crear una malla estructurada de elementos hexaédricos se requiere la asignación de la propiedad **Transfinite** a las líneas, superficies y volúmenes que conforman el dominio del problema.

7.1.6.1. Preparación de las líneas

Todas las líneas que conforman el dominio del problema debe tener asignado el parámetro **Transfinite Curve**, el cual determina la cantidad de divisiones que ésta

tendrá cuando se realice el enmallado. Esto se logra desplegando la ventana **Modules** ▷ **Mesh** ▷ **Define** ▷ **Transfinite** ▷ **Curve**.

A continuación se selecciona el número n de nudos que tendrá la línea en el enmallado. Cabe mencionar que el número de divisiones que tendrá la línea es $n - 1$.

También se selecciona el tipo de divisiones que se realizarán, donde la opción **Progression** está relacionada con una progresión geométrica que utiliza un coeficiente de progresión que se introduce en el cuadro **Parameter**. Por otro lado, la opción **Bump** establece un refinamiento hacia los extremos de la curva.

La opción para el tipo de divisiones seleccionada en los ejemplos realizados en este trabajo es **Progression**, con **Parameter** = 1, con lo cual se asignan divisiones uniformes de acuerdo al número de puntos seleccionado.

Para realizar una malla con elementos hexaédricos, es recomendable planear cuidadosamente el número de divisiones que se realizarán en cada línea, buscando que la futura malla tenga coherencia. Asimismo, se debe garantizar que las caras opuestas en las superficies cuadrilaterales tengan el mismo número de divisiones.

En seguida se seleccionan las curvas a las que se les asignará la misma división y se oprime la tecla **e**. Para finalizar se oprime la tecla **q**. Al realizar el procedimiento anterior, en el archivo de extensión **.geo** se crea una instrucción que tiene la forma:

$$\text{TransfiniteCurve}\{i\text{linea}, j\text{linea}, \dots\} = npunto \text{ Using Progression param}$$

donde $i\text{linea}, j\text{linea} \dots$ son los identificadores de las líneas a las que se les asigna la propiedad, $npunto$ es el número de puntos que va a tener la curva y $param$ es el parámetro de progresión.

7.1.6.2. Preparación de las superficies

Para asignar la propiedad **Transfinite Surface** a las superficies, se debe desplegar la ventana seleccionando **Modules** ▷ **Mesh** ▷ **Define** ▷ **Transfinite** ▷ **Surface**. A continuación se da clic en la superficie a la que se asignará la propiedad. Luego se da clic en cada uno de los vértices que tiene esta superficie, cuidando de recorrer el contorno de ésta sin omitir ningún punto. Una vez se seleccionan todos los vértices de la superficie, se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Para realizar el procedimiento anterior en algunos problemas con geometrías complejas es conveniente utilizar las opciones que ofrece GMSH de visualización exclusiva de ciertas entidades, para dejar visible solamente la superficie a la que se desee asignar la propiedad **Transfinite Surface**.

La asignación de la propiedad **Transfinite Surface** se logra seleccionando en la barra de menú superior **Tools** ▷ **Visibility**, lo cual despliega una ventana en donde

se pueden seleccionar los identificadores de las entidades que se quieren visualizar exclusivamente, en este caso la superficie. Luego se da clic en **Apply**, y solamente se visualizan las entidades seleccionadas.

Al realizar la asignación de la propiedad **Transfinite Surface**, en el archivo de extensión **.geo** se crea una instrucción que tiene la forma:

$$\text{Transfinite Surface}\{isup\} = \{ipun, jpun, kpun, \dots\}$$

donde *isup* es el identificador de la superficie a las que se les asigna la propiedad, *ipun, jpun, kpun, ...* son los identificadores de los puntos que conforman el contorno de la superficie.

7.1.6.3. Preparación de los volúmenes

La ventana de asignación de la propiedad **Transfinite Volume** se obtiene seleccionando **Modules** ▷ **Mesh** ▷ **Define** ▷ **Transfinite** ▷ **Volume**. En seguida se da clic en el volumen al que se le va a asignar la propiedad. A continuación se seleccionan los nudos que conforman el volumen.

Se recomienda realizar esta selección de vértices siguiendo las reglas de numeración del hexaedro mostradas en la [Subsección 3.6.1](#) para el caso de volúmenes con forma de hexaedros, y las reglas de numeración de la [Subsección 3.7.1](#) para el caso de volúmenes con forma de prisma o cuña. Una vez se seleccionan todos los vértices del volumen, se oprime la tecla **e**. Para finalizar se oprime la tecla **q**.

Luego de la asignación de la propiedad **Transfinite Volumen**, en el archivo de extensión **.geo** se crea una instrucción que tiene la forma:

$$\text{TransfiniteVolumen}\{ivol\} = \{ipun, jpun, kpun, \dots\}$$

donde *ivol* es el identificador del volumen al que se asigna la propiedad, *ipun, jpun, kpun, ...* son los identificadores de los puntos que conforman su contorno.

7.1.6.4. Preparación de los volúmenes

Luego de asignar la propiedad **Transfinite** a las curvas, superficies y volúmenes, que conforman el sólido del problema, se da la instrucción para que se combinen los elementos tetraédricos de la futura malla estructurada y se formen elementos hexaédricos y prismas, si se requiere.

Esto se logra al seleccionar **Modules** ▷ **Mesh** ▷ **Define** ▷ **Recombine**. A continuación se seleccionan todas las superficies del problema, manteniendo presionada la tecla **Ctrl** y manteniendo el clic para crear de esta manera un rectángulo de selección

que contenga todas las entidades del problema. A continuación se oprime la tecla **e** y luego para finalizar se oprime la tecla **q**.

Luego de realizar el proceso anterior, en el archivo de extensión **.geo** se crea una instrucción que tiene la forma:

$$\text{Recombine Surface}\{i_{sup}, j_{sup}, k_{sup}, \dots\};$$

donde $i_{sup}, j_{sup}, k_{sup}, \dots$ son los identificadores de las superficies seleccionadas.

7.1.7. Generación de la malla

Una vez asignadas las propiedades de los materiales, los desplazamientos, las cargas, y la propiedad **Transfinite** a las entidades geométricas que conforman el problema, se genera la malla de elementos finitos haciendo clic en **Modules** \triangleright **Mesh** \triangleright **3D**, u oprimiendo la tecla **3**.

Si luego de generar la malla se requiere modificar algún parámetro de la geometría o de la propiedad **Transfinite**, es posible recargar el archivo de extensión **.geo** mediante **Modules** \triangleright **Geometry** \triangleright **Reload script** o también oprimiendo la tecla del número **0**.

7.1.8. Creación del archivo **.msh**

Con la malla generada correctamente mediante los procedimientos descritos en los pasos anteriores, el programa GMSH permite exportar a un archivo de extensión **.msh** en formato ASCII de la versión 2 de GMSH, el cual puede ser leído por el programa PEFICA 2.0 con las modificaciones realizadas en este trabajo final de maestría. El procedimiento es el siguiente:

- Hacer clic en el menú superior en **File** \triangleright **Export**. Esto abrirá la ventana **Export**, la cual se muestra en la [Figura 7.7](#).
- Escribir el nombre del archivo junto con la extensión **.msh** en el cuadro de texto **Save As** de la ventana **Export**. En esta ventana se selecciona también la carpeta que contendrá éste archivo.
- Seleccionar el formato de exportación del archivo **.msh** como **Mesh-Gmsh MSH (*.msh)**. Luego hacer click en el botón **Save**.
- Seleccionar **Versión 2 ASCII** en la caja **Format** de la ventana **MSH Options** y dejar sin marcar las opciones **Save all elements** y **Save parametric coordinates**, como se muestra en la [Figura 7.8](#).

- Hacer clic en el boton **OK**

La conformación de este archivo se puede consultar en detalle en la referencia (Linero *et al.*, 2020).

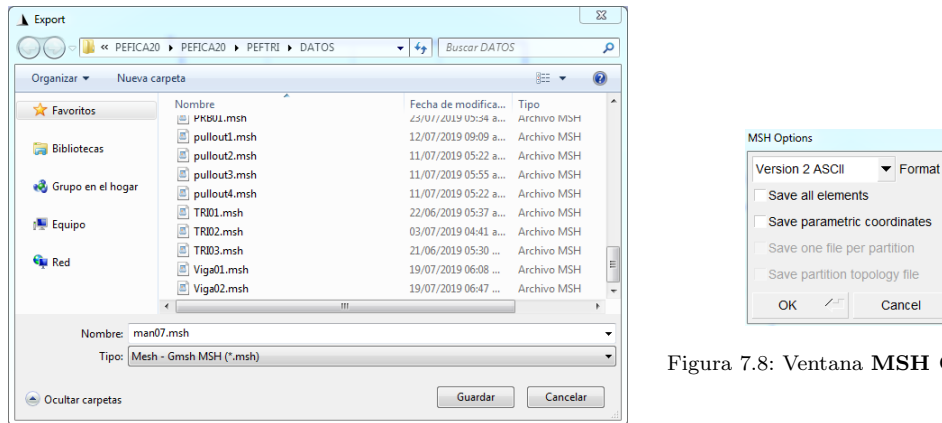


Figura 7.7: Selección del nombre del archivo en la ventana **Export**.

Figura 7.8: Ventana **MSH Options**.

7.2. Ejecución del programa en GNU-OCTAVE o MATLAB

Las funciones de cálculo están escritas en el lenguaje interpretado de GNU-OCTAVE, siendo compatibles con MATLAB. Esto significa que es necesario alguno de los dos programas para ejecutar éstas funciones.

Ya sea en GNU OCTAVE o en MATLAB, se requiere establecer el directorio de trabajo del programa. Este directorio es la carpeta `\PEFTRI`, la cual contiene el archivo `PEFICA.m`.

En GNU OCTAVE el directorio de trabajo se selecciona escogiendo la carpeta en el **Explorador de archivos**. En MATLAB el directorio de trabajo se puede seleccionar de la misma manera. Luego de establecer el directorio de trabajo y una vez construido el archivo de entrada, éste se puede analizar a través de la rutina **PEFICA.m** como una función de GNU-OCTAVE o MATLAB. La instrucción de llamada al programa se ingresa en la ventana de comandos, y es la siguiente:

```
» PEFICA <adat> <ople>
```

donde `<adat>` es el nombre del archivo de datos, escrito sin extensión. Este archivo debe estar contenido en la carpeta `\DATOS`.

El parámetro $\langle ople \rangle$ es la opción de lectura de datos y escritura de resultados. A continuación se muestran los valores que puede tomar:

- $\langle ople \rangle = 10$: es el valor por defecto. Se leen los datos de un archivo con extensión `.msh` de GMSH y se escriben los resultados de los esfuerzos y deformaciones promedio en los nudos en los archivos `.pos` y `.pos.opt` de GMSH
- $\langle ople \rangle = 12$: Se leen los datos de un archivo con extensión `.msh` de GMSH y se escriben los resultados de los esfuerzos y deformaciones promedio en los nudos en los archivos `.pos` y `.pos.opt` de GMSH. Se utiliza cuando hay más de un tipo de elemento finito o más de un material.

Luego de ejecutar la instrucción anterior, el programa muestra las etapas de cálculo y su tiempo empleado, en la misma ventana de comandos.

En la subcarpeta `\DATOS` se encuentran los archivos utilizados en los ejemplos mostrados en este trabajo.

7.3. Lectura de resultados en el programa GMSH

El módulo de post proceso del programa GMSH puede presentar de forma gráfica la configuración deformada del sólido y las reacciones en los apoyos. Asimismo, puede mostrar los resultados de desplazamiento, deformación o esfuerzo.

En esta sección se describirá de forma breve los principales aspectos de la visualización de los resultados utilizando el módulo de post proceso del programa GMSH. En la referencia (Linero *et al.*, 2020) se puede consultar con más detalle el uso de este programa para el postproceso.

El programa GMSH lee el archivo con extensión `.pos`, el cual está dividido en bloques que contienen la malla de elementos y los resultados de desplazamientos, reacciones, esfuerzos y deformaciones. También lee el archivo con extensión `.pos.opt` el cual contiene las opciones de visualización de resultados (Linero *et al.*, 2020).

El archivo que se abre desde el programa GMSH para visualizar el postproceso es de extensión `.pos`, el cual contiene la solución al problema. Al abrirlo, en la parte izquierda de la ventana principal de GMSH aparecen las diferentes opciones principales de visualización de los resultados contenidos en el archivo de extensión `.pos`.

Cada una de las opciones presentadas en la parte izquierda de la pantalla se puede activar o desactivar dando clic sobre ella. Esta acción activa o desactiva la visualización de cada resultado sobre la malla de elementos mostrada en la ventana principal.

Estas opciones varían según el número de categorías del problema. Cada grupo de vistas de una categoría contiene: las componentes de esfuerzo σ_{xx} , σ_{yy} , σ_{zz} , σ_{xy} , σ_{xz} y σ_{yz} , los esfuerzos principales σ_1 , σ_2 y σ_3 , el esfuerzo de Von Mises σ_{vm} , las componentes de deformación ϵ_{xx} , ϵ_{yy} , ϵ_{zz} , τ_{xy} , τ_{xz} y τ_{yz} , y las deformaciones principales ϵ_1 , ϵ_2 y ϵ_3 . Estas opciones se muestran en la [Figura 7.9](#)

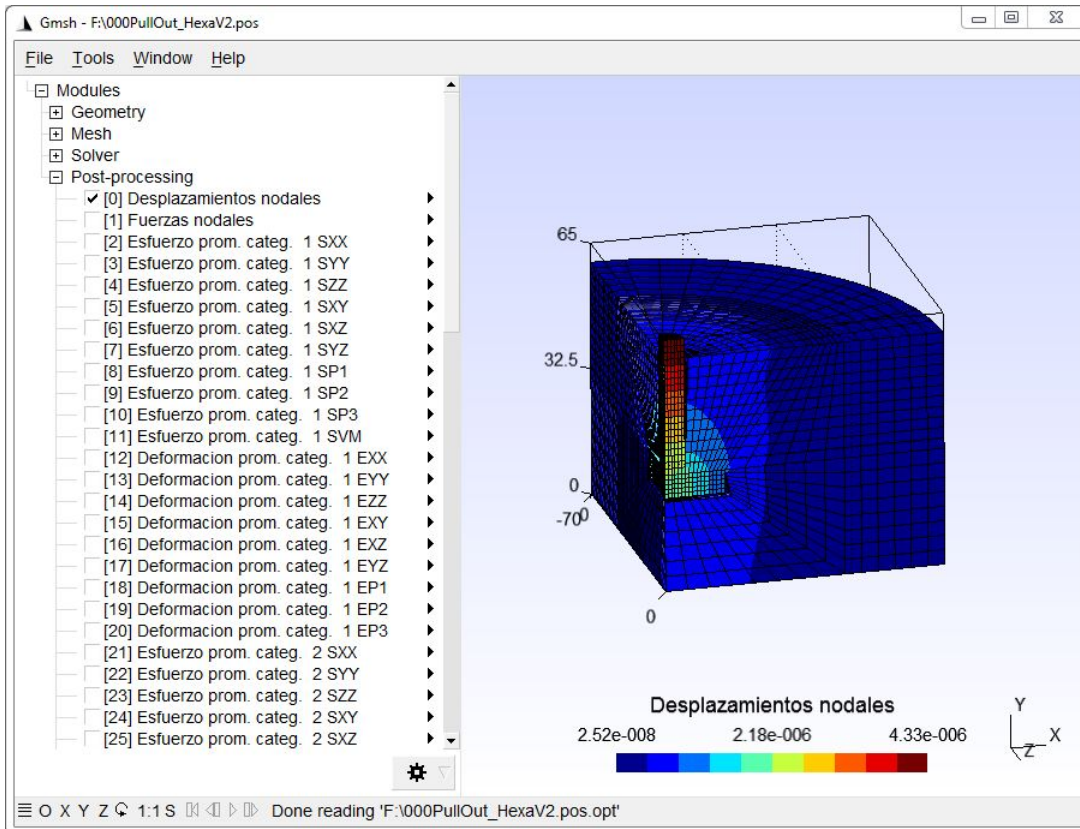


Figura 7.9: Entorno gráfico de GMSH para el post proceso.

7.3.1. Desplazamientos, condición deformada y fuerzas en los nudos

Luego de abrir el archivo de extensión .pos, el módulo de post proceso del programa GMSH muestra la norma del vector de desplazamiento, designada como la vista número 0. Adicionalmente, se muestra la malla de elementos finitos sin deformar.

Si se desea cambiar el factor de amplificación del vector de desplazamiento, para ver la deformada con una escala aumentada, se selecciona en el menú desplegable de la vista, la opción **Options**. Esto despliega la ventana de opciones. A continuación

se da clic en la pestaña **Aspect**. El factor de amplificación de la vista se puede modificar ingresando un nuevo valor en el cuadro **Displacement factor**.

Las fuerzas totales que actúan en los nudos de la malla se muestran en la vista número 1. Por defecto se muestran los vectores de fuerza entrando a los nudos mediante flechas. El tipo y el tamaño de las flechas se pueden modificar seleccionando la opción **Options** del menú desplegable de la vista **Fuerzas nodales**. Esta acción abre la ventana de opciones, donde se da clic en la pestaña **Aspect**. El tamaño de las flechas se puede modificar ingresando un nuevo valor mínimo y máximo en los cuadros de la opción **Arrow Size**.

7.3.2. Componentes de esfuerzo y deformación

Dependiendo de los tipos de elementos finitos utilizados y de la cantidad de materiales asignados al sólido, se asignó un número de categorías. Para cada una de estas categorías, el programa GMSH despliega una opción de visualización donde se muestran las componentes de esfuerzo σ_{xx} , σ_{yy} , σ_{zz} , σ_{xy} , σ_{xz} y σ_{yz} , los esfuerzo principales σ_1 , σ_2 y σ_3 , el esfuerzo de Von Mises σ_{vm} , las componentes de deformación ε_{xx} , ε_{yy} , ε_{zz} , τ_{xy} , τ_{xz} y τ_{yz} , y las deformaciones principales ϵ_1 , ϵ_2 y ϵ_3 .

Los resultados para una sola categoría son evaluados como el promedio en los nudos al ejecutar PEFICA con la opción $\langle ople \rangle = 10$. Para obtener los resultados en los nudos para múltiples categorías, se ejecuta PEFICA con la opción $\langle ople \rangle = 12$.

Cuando se presentan múltiples categorías, los resultados de cada una se pueden mostrar de forma simultánea. Al visualizar de forma simultánea el mismo resultado de dos categorías diferentes, el programa muestra una escala cromática independiente para cada una. La explicación detallada de las opciones que presenta el programa GMSH para la visualización de resultados en el postproceso de los resultados generados por el programa desarrollado, se puede encontrar en la referencia (Linero *et al.*, 2020).

Referencias

- ASTM C900. (2007). Standard test method for pullout strength of hardened concrete. *American Society for Testing and Materials*, 1-10. <https://doi.org/10.1520/C0900-14.2>
- Dahlblom, O., Peterson, A. & Petersson, H. (1986). CALFEM — A program for computer-aided learning of the finite element method. *Engineering Computations*, 3(2), 155-160. <https://doi.org/10.1108/eb023653>
- Dudley, C. R. & Owen, S. J. (2014). Degenerate Hex Elements. *Procedia Engineering*, 82, 301-312. <https://doi.org/10.1016/j.proeng.2014.10.392>
- Duque, C. & Mejía, M. (2007). *Herramientas de Software con licencia pública general para el modelado por elementos finitos* (Universidad Nacional de Colombia. Sede de Medellín. Facultad Nacional de Minas., Ed.; Vol. 74). 2006, Revista DYNA.
- Eaton, J., Bateman, D., Hauberg, S. & Wehbrin, R. (2017). *GNU Octave version 4.2.1 manual: a high-level interactive language for numerical computations*. <https://www.gnu.org/software/octave/doc/v4.2.1/>
- Felippa, C. A. (2017a). 11 The 8-Node Hexahedron (U. of Colorado, Ed.). En U. of Colorado (Ed.), *Advanced Finite Element Methods*. Boulder, Colorado, USA.
- Felippa, C. A. (2017b). 8 Solid Elements: Overview (University of Colorado, Ed.). En University of Colorado (Ed.), *Advanced Finite Element Methods*. Boulder, Colorado, USA, Department of Aerospace Engineering Sciences; Center for Aerospace Structures. <https://www.colorado.edu/engineering/cas/courses.d/AFEM.d/Home.html>
- Geuzaine, C. & Remacle, J. (2019). *Gmsh reference manual. The documentation for Gmsh 4.3.0. A finite element mesh generator with built-in pre- and postprocessing facilities*. <http://gmsh.info/doc/texinfo/gmsh.html>
- International Center for Numerical Methods in Engineering (CIMNE). (1987). *GID. Barcelona, España*. <https://www.gidhome.com/>
- Kohnke, P. (2009). *Theory Reference for the Mechanical APDL and Mechanical Applications* (Vol. 3304).

- Linero, D. (2010). *PEFiCA - Programa de elementos finitos a código abierto. versión 1.0*. Bogotá, Universidad Nacional de Colombia.
- Linero, D., Estrada, M. & Garzon, D. (2020). *Programación del método de los elementos finitos aplicado al análisis mecánico estático de sólidos*. Bogotá, Universidad Nacional de Colombia.
- Linero, D. & Garzón, D. (2010). *Elementos de la mecánica del medio continuo para cuerpos sólidos*. Universidad Nacional de Colombia. <https://books.google.com.co/books?id=Q0WboVF90FEC>
- Linero, D., Garzón, D. & Ramírez, A. (2013). *Análisis estructural mediante el método de los elementos finitos. Introducción al comportamiento lineal elástico* (1.^a ed.). Bogotá, Universidad Nacional de Colombia.
- Link, M. (2005). *MATFEM User's Guide*. Kassel, Alemania, University of Kassel.
- Marqués, P. (1995). *Software Educativo: guía de uso y metodología de diseño* (ESTEL, EditorialL, Ed.).
- Marqués, P. (2002). Evaluación y selección de software educativo. *Comunicación y Pedagogía*, (185), 31-37.
- Molina, M. (2006). *Programación en lenguajes estructurados* (McGraw-Hill Interamericana de España S.L, Ed.; 1.^a ed.). <http://assets.mheducation.es/bcv/guide/capitulo/8448148703.pdf>
- Simo, J. & Rifai, M. (1990). A class of mixed assumed strain methods and the method of incompatible Modes. *International Journal for numerical methods in engineering Vol 29*, 1595-1638. <https://doi.org/10.1002/nme.1620290802>
- Zienkiewicz, O. C. (2013). *The Finite Element Method: Its Basis and Fundamentals*. <https://doi.org/10.1016/b978-1-85617-633-0.00020-4>