

## SOFTWARE PARA LA EVALUACIÓN DE LA INFLUENCIA DE LA FORMULACIÓN DE ELEMENTOS FINITOS EN LA SIMULACIÓN DEL COMPORTAMIENTO DE ESTRUCTURAS SIMPLES - PARTE 2

### SOFTWARE FOR THE EVALUATION OF THE INFLUENCE OF THE FORMULATION OF FINITE ELEMENTS IN THE SIMULATION OF THE BEHAVIOR OF SIMPLE STRUCTURES – PART 2

Santiago Zera<sup>a</sup>, Daniel E. López<sup>a,b</sup> y Carlos J. Leiva<sup>a,b</sup>

<sup>a</sup>Facultad de Ingeniería, Universidad Nacional de Cuyo, Centro Universitario, Mendoza, Argentina, santizera@gmail.com.ar, <https://ingenieria.uncuyo.edu.ar/>

<sup>b</sup>Cátedra Estructuras Laminadas, daniel.lopez@ingenieria.uncuyo.edu.ar, carlos.leiva@ingenieria.uncuyo.edu.ar

**Palabras clave:** Método de Elementos Finitos, software educativo, cáscaras

**Resumen.** El uso de software basado en el Método de los Elementos Finitos (MEF) requiere conocer los distintos tipos de elementos y las variables implícitas en su formulación, como la discretización del campo incógnita, las funciones de forma, el esquema de integración numérica y la compatibilidad inter-elemental. El MEF es el método más utilizado para análisis estructural en diversas ramas de la ingeniería. Sin embargo, sus usuarios no siempre tienen en cuenta las variables involucradas ni la influencia que las mismas pueden tener en los resultados. En parte, esto se debe a que el software disponible no siempre permite interactuar con ellas. En este trabajo se describe un software desarrollado por los autores para la enseñanza del MEF que permite la interacción con estas variables para la evaluación de su influencia en los resultados de simulaciones estructurales. En la primera parte (S. Zera et al., Mecánica Computacional, 42:1587–1596, (2023)) se presentaron las capacidades del software para estudiar estructuras bidimensionales bajo estados planos de tensión y deformación y placas sometidas a flexión. En esta segunda parte se extienden sus capacidades para el estudio de estructuras de cáscara.

**Keywords:** Finite Element Method, educational software, shells

**Abstract.** The use of software based on the Finite Element Method (FEM) requires knowledge of the different types of elements and the variables implicit in their formulation, such as the discretization of the unknown field, the shape functions, the numerical integration scheme, and the inter-element compatibility. FEM is the most widely used method for structural analysis in various branches of engineering. However, its users do not always take into account the variables involved or the influence they may have on the results. In part, this is because the available software does not always allow interaction with them. This work describes a software developed by the authors for teaching FEM that allows interaction with these variables to evaluate their influence on the results of structural simulations. In the first part (S. Zera et al., Mecánica Computacional, 42:1587–1596, (2023)) the capabilities of the software to study two-dimensional structures under states of plane stress and strain and plate bending were presented. In this second part, its capabilities are extended for the study of shell structures.

## 1 INTRODUCCIÓN

El Método de los Elementos Finitos (MEF) es uno de los métodos numéricos más empleados en diversas ramas de la ingeniería para realizar análisis estructural. Su aplicación exige que el usuario tenga un conocimiento claro de las variables involucradas en su formulación, como el tipo de campo incógnita elegido (desplazamientos, tensiones, etc.), las funciones de forma empleadas para la discretización del mismo, y el esquema de integración numérica utilizado para calcular las matrices de rigidez, masa, etc. No obstante, es un hecho que muchos usuarios no siempre reconocen la relevancia de estas variables ni su impacto en los resultados de las simulaciones. Esto se debe en parte a que el software disponible generalmente no facilita la interacción directa del usuario con las variables que están implícitas en la formulación de los elementos.

En la primera parte de este proyecto (Zera et al., 2023) se presentó un software desarrollado por los autores que permite al usuario manipular en forma explícita algunas de las variables implícitas en la formulación del MEF para así poder analizar su influencia en el comportamiento de modelos estructurales bidimensionales simples bajo estados planos de tensión y deformación y con comportamiento de placa plana.

En esta segunda parte se toma como base el desarrollo realizado en la primera y se extienden las capacidades del mencionado software para permitir el análisis de estructuras de cáscara. Se pretende que las capacidades ofrecidas por el software permitan mejorar la comprensión de sus usuarios respecto al uso de MEF en distintas aplicaciones y sirvan para poner de manifiesto la influencia significativa que las variables propias de la formulación del método pueden tener en los resultados de la simulación del comportamiento de estructuras simples.

## 2 DESCRIPCIÓN DEL SOFTWARE

El software presentado en este trabajo permite realizar análisis estructural elástico lineal de estructuras de cáscara. El software, denominado **Z-MEF Shells**, permite manipular en forma explícita variables implícitas en la formulación de los elementos finitos, como son la discretización del campo incógnita (funciones de forma) y el esquema de integración utilizado para el cálculo de matrices de rigidez.

Como es sabido, cualquier proceso de cálculo mediante MEF consta de tres etapas principales: *pre proceso*, *cálculo* y *post proceso*. En el caso de Z-MEF, las etapas de pre proceso de los datos del problema y post proceso de los resultados se realizan dentro del ambiente que proporciona el software de libre acceso GiD<sup>1</sup>, desarrollado por CIMNE<sup>2</sup>. Para su vinculación con motores de cálculo desarrollados por los usuarios, GiD utiliza lo que se conoce como *problemtipe*, un conjunto de archivos ubicados en un mismo directorio, a partir de los cuales se definen los distintos parámetros que es posible ingresar y manipular en el modelo.

En el caso de Z-MEF Shells, el motor de cálculo está escrito en el lenguaje de programación Python. La estructura del programa es sencilla. Consta de un archivo principal **main.py** que invoca sucesivamente a tres objetos que se encargan de llevar adelante las distintas rutinas y cálculos. Estos objetos son:

- **Input:** Lee y almacena la información desde el archivo de entrada generado por GiD en la etapa de pre proceso.
- **Solver:** Toma la información almacenada en el objeto **Input** y genera la representación matemática del modelo: crea los distintos elementos, calcula sus matrices de rigidez y vectores de carga, ensambla el sistema de ecuaciones lineales y lo resuelve, obteniendo

---

<sup>1</sup> GiD SIMULATION, <https://www.gidsimulation.com/>

<sup>2</sup> CIMNE, Centro Internacional de Métodos Numéricos en Ingeniería, <https://www.cimne.com/>

los desplazamientos de la estructura, a partir de los cuales se calculan deformaciones, tensiones, etc. en cada elemento.

- **Output:** Este objeto toma los resultados obtenidos por el objeto **Solver** y crea el archivo de salida que luego es interpretado por la interfaz de post proceso de GiD.

El esquema de la [Figura 1](#) ilustra el proceso de cálculo y la vinculación entre Z-MEF Shells y GiD. Se indican también los archivos relacionados con las distintas etapas del programa.

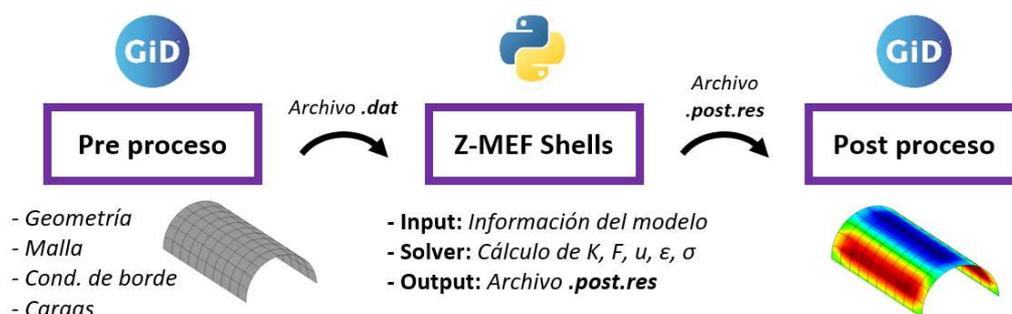


Figura 1: Esquema conceptual de la organización del código de Z-MEF Shells y su vinculación con GiD

A continuación se describen brevemente las características principales de la formulación adoptada para los elementos y se detallan las distintas capacidades del software en las etapas de *pre proceso*, *cálculo* y *post proceso*.

## 2.1 Introducción teórica. Variables implícitas en la formulación de elementos de cáscara plana

Como se ha mencionado, el software presentado en este trabajo permite el análisis estructural elástico lineal de estructuras de cáscara de superficie media arbitraria. Para ello se utilizan *elementos de cáscara plana*, de modo tal que la geometría de una cáscara con curvatura se aproxima mediante un conjunto de elementos planos. A medida que el tamaño de los elementos planos se reduce, mejor es la aproximación de la discretización respecto a la geometría real de la superficie media de la cáscara.

En particular se utilizan los elementos de cáscara plana de Reissner-Mindlin, cuya principal característica es que, para materiales homogéneos e isotrópicos (como los utilizados en este software), la matriz de rigidez local  $K$  de un elemento se obtiene simplemente como:

$$K = K_m + K_f + K_c$$

donde:

- $K_m$  es la matriz de rigidez de membrana, correspondiente a la matriz de rigidez de un elemento de tensión plana.
- $K_f$  es la matriz de rigidez de flexión, correspondiente, en este caso, a la matriz de rigidez de flexión de un elemento de placa basado en la teoría de Reissner-Mindlin.
- $K_c$  es la matriz de rigidez de corte, correspondiente también a la matriz de rigidez de corte de un elemento de placa de Reissner-Mindlin.

En la primera parte de este proyecto ([Zera et al., 2023](#)) se codificaron por separado dos módulos con elementos capaces de representar un estado de tensión plana (Z-MEF 2D) y elementos de placa (Z-MEF Placas). En esta segunda parte, los elementos implementados en dichos módulos se completan para obtener los *elementos de cáscara plana*, que pueden ser utilizados para modelar estructuras de lámina de superficie media arbitraria.

A nivel del elemento, las tres matrices  $K_m$ ,  $K_f$  y  $K_c$  contribuyen a la matriz de rigidez local  $K$  en forma desacoplada. El acoplamiento de los comportamientos de membrana y de flexión,

característica propia de este tipo de estructuras, se da al nivel de la estructura, cuando se han ensamblado los distintos elementos planos que conforman la malla.

La cantidad de puntos de integración a utilizar en las cuadraturas de Gauss con las que se evalúan cada una de estas tres matrices constituye una variable implícita en la formulación de este tipo de elementos. El presente desarrollo permite adoptar distinta cantidad de puntos de integración para  $K_m$ , para  $K_f$  y para  $K_c$ .

Otro aspecto relevante de esta formulación es el que se describe a continuación. Los grados de libertad de cada nodo son seis: tres desplazamientos y tres giros, medidos respecto a un sistema de ejes locales. La matriz de rigidez de membrana  $K_m$  aporta rigidez respecto a los dos desplazamientos en el plano del elemento; mientras que las matrices de rigidez de flexión  $K_f$  y de corte  $K_c$  aportan rigidez respecto al desplazamiento normal al plano y a los dos giros cuyos vectores se encuentran en dicho plano. De esta manera, en esta formulación, a nivel del elemento, el giro normal al plano, conocido como grado de libertad *drilling*, no tiene ninguna rigidez asociada.

En la simulación del comportamiento de estructura planas la rigidez nula en la dirección de los grados de libertad *drilling*, conduce a matrices de rigidez globales inestables. Para salvar esta situación, el enfoque adoptado consiste en asignar una rigidez ficticia proporcional al módulo de elasticidad  $E$  del elemento y a su volumen  $V$ , siendo la constante de proporcionalidad  $\alpha$  un valor que se adopta entre  $10^{-6}$  y  $10^{-2}$ , (Oñate, 1995; Celigüeta Lizarza, 2011). Así, el valor de esta constante constituye otra de las variables implícitas en la formulación del elemento y cuya influencia en los resultados podría ser significativa.

Por último, otra de las variables implícitas en la formulación del método es la cantidad de nodos utilizados para la discretización del campo incógnita, directamente asociada a las funciones de forma utilizadas para interpolar dicho campo en el interior de los elementos. En este sentido, el software permite la posibilidad de adoptar elementos cuadriláteros de 4, 8 y 9 nodos.

## 2.2 Pre proceso

Para comenzar a interactuar con el programa, en la interfaz de GiD se selecciona el *problemtipe* denominado *Z-MEF Shells*. Hecho esto, ya es posible generar/importar geometrías, asignar materiales, imponer condiciones de borde, cargas, y definir toda la información necesaria para conformar el modelo estructural.

### 2.2.1 Geometría

Es posible tanto generar geometrías desde cero, como importarlas desde archivos preparados en otros softwares específicos para ese fin. Sobre las entidades de esta geometría (vértices, superficies, bordes) se asignan luego las distintas condiciones que son parte de la información necesaria para la representación del modelo estructural (cargas, condiciones de borde, materiales, etc.).

### 2.2.2 Malla

Una vez que se cuenta con la geometría del modelo, ya es posible generar la malla de elementos finitos correspondiente. En este caso, el software cuenta con la posibilidad de utilizar elementos cuadriláteros lagrangeanos de 4 y 9 nodos; así como elementos cuadriláteros serendípticos de 8 nodos. Esta selección tiene influencia en el tipo de funciones de forma utilizadas para la discretización del campo incógnita (desplazamientos y giros nodales).

Si bien los elementos cuadriláteros de 8 y 9 nodos podrían modelar por sí mismos superficies con curvatura, el programa solo utiliza elementos de cáscara planos, de modo que todos los

nodos de un mismo elemento son forzados al momento de la generación de la malla a mantenerse en un mismo plano. La modelación de superficies con curvatura es posible a partir del ensamblaje de múltiples elementos planos.

Un aspecto a destacar es que, así como las cargas, condiciones de borde y materiales se pueden asignar sobre entidades geométricas, también es posible su asignación en forma directa a entidades de la malla (elementos, nodos y bordes).

### 2.2.3 Materiales

En esta versión del software se pueden utilizar materiales elásticos lineales e isotrópicos, definidos por su módulo elástico  $E$ , coeficiente de Poisson  $\nu$  y densidad  $\gamma$ . A su vez, el espesor de las cáscaras se asigna también como una propiedad del material.

### 2.2.4 Condiciones de bordes

Es posible aplicar condiciones de apoyo rígido, elástico (asignando un determinado valor de rigidez al apoyo) o de desplazamientos prescritos. Estas condiciones pueden imponerse tanto en el sistema de coordenadas global como en sistemas de coordenadas locales definidos por el usuario.

### 2.2.5 Cargas

Es posible aplicar fuerzas y momentos puntuales, así como cargas distribuidas sobre bordes o superficies. Tanto las cargas puntuales como las distribuidas se definen especificando sus componentes, ya sea en el sistema de coordenadas global, o en algún sistema de coordenadas locales definido por el usuario.

### 2.2.6 Unidades

La magnitud de fuerzas, desplazamientos, rigideces y demás propiedades físicas puede definirse en las distintas unidades del Sistema Internacional.

### 2.2.7 Esquema de integración

La formulación de los elementos adoptados tiene como una de sus variables implícitas la cantidad de puntos de integración utilizados para la evaluación de las matrices de rigidez de membrana  $K_m$ , de flexión  $K_f$  y de corte  $K_c$ . Este programa permite al usuario definir en forma sencilla y directa el esquema de integración numérica utilizado para la evaluación de estas matrices. Así, para cada una de ellas es posible elegir entre 1, 4 y 9 puntos de integración.

Se señala que, junto con la posibilidad de definir la cantidad de nodos de los elementos (4, 8 o 9 nodos), la elección de la cantidad de puntos de integración es una de las características principales de este software educativo.

## 2.3 Solver

El software presentado en este trabajo permite realizar el análisis de estructuras de cáscara de superficie media arbitraria. La teoría a partir de la cual se codificó el programa se basa en [Oñate \(1995\)](#), [Celigüeta Lizarza \(2011\)](#), [Zienkiewicz y Taylor \(2005\)](#) y las notas de clase del Curso 2022 de la Cátedra Estructuras Laminadas de la Facultad de Ingeniería de la Universidad Nacional de Cuyo.

El motor de cálculo está escrito en el lenguaje de programación Python y consta de un archivo principal desde el cual se invoca sucesivamente a tres objetos que: leen la información de entrada (archivo *.dat*), realizan los cálculos propios del MEF y escriben el archivo de salida

(archivo *.post.res*).

En primer lugar, el programa lee desde el archivo de entrada generado por GiD toda la información relacionada con el modelo (tipos de elementos, conectividades, materiales, condiciones, cargas, etc.) y luego pone en marcha todas las rutinas relacionadas con el MEF: crea los distintos elementos, calcula sus matrices de rigidez y vectores de carga.

En la estructura del código, cada tipo de elemento finito es representado por una clase de Python, y cada elemento de la malla del modelo se instancia como un objeto según el tipo de elemento que le corresponde (langrangeano de 4 o 9 nodos, o serendípito de 8 nodos). Cada uno de estos objetos cuenta con métodos para la evaluación de sus funciones de forma, matrices de rigidez, etc.

Para el cálculo de las distintas matrices de rigidez de cada elemento se utiliza una cuadratura de Gauss específica según la cantidad de puntos de integración indicada por el usuario en la etapa de pre proceso dentro de GiD.

A continuación, las matrices de rigidez y vectores de carga de los elementos se ensamblan para formar el sistema de ecuaciones lineales que representa el comportamiento de la estructura. Una vez resuelto dicho sistema, habiendo obtenido así los desplazamientos en cada nodo de la malla, se calculan deformaciones y tensiones en cada uno de los elementos. En este aspecto, el programa es capaz de calcular:

- Tensor de deformaciones
- Tensor de tensiones
- Tensiones principales y tensión de Von Mises
- Variables internas generalizadas: esfuerzos de corte, normal y momentos flectores

Estas magnitudes se evalúan respecto a ejes locales del elemento que pueden ser modificados a gusto por el usuario y se definen en la interfaz de GiD durante el pre proceso.

## 2.4 Post proceso

Una vez realizados los cálculos de desplazamientos, deformaciones y tensiones, el programa escribe el archivo de salida que luego es interpretado por GiD. A partir de este archivo, GiD es capaz de presentar la configuración deformada de la estructura, junto con mapas de color de los distintos resultados.

## 3 EJEMPLOS DE APLICACIÓN

Se presentan a continuación ejemplos de uso del programa Z-MEF Shells. En estos ejemplos se busca ilustrar algunos de los análisis que es posible realizar con herramientas de software educativo como la que se presenta en este trabajo.

### 3.1 Bóveda cilíndrica

En este primer ejemplo, extraído de [Celigieta Lizarza \(2011\)](#), se analiza la bóveda que se muestra en la [Figura 2](#), también conocida como *bóveda de Scordellis-Lo*. Se trata de una bóveda cilíndrica con desplazamientos restringidos en sus extremos y sometida a una carga en dirección vertical uniformemente distribuida sobre su superficie. El material tiene un módulo de elasticidad de  $4.32 \times 10^8$  y un coeficiente de Poisson  $\nu = 0$ . El espesor de la cáscara es 0.25.

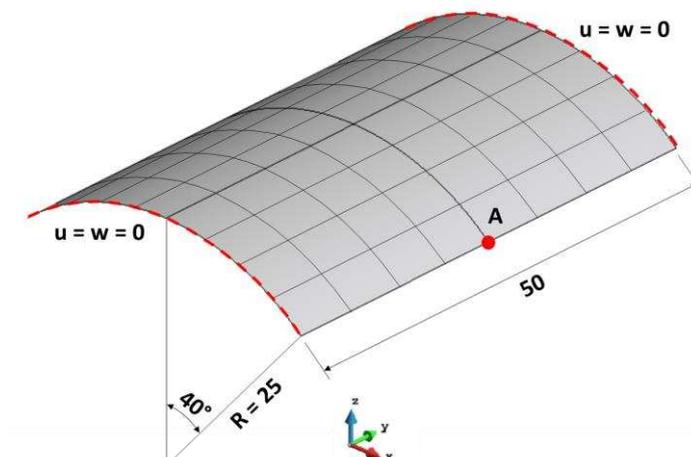


Figura 2: Bóveda cilíndrica de Scordellis-Lo

En este ejemplo se evalúa la influencia que tienen en el valor del desplazamiento vertical en el punto A las siguientes variables:

- **Cantidad de nodos:** Se analiza el comportamiento de elementos de 4, 8 y 9 nodos.
- **Cantidad de puntos de integración:** Se adoptan esquemas de integración *completa* y *reducida*.
- **Densidad de malla:** Se adoptan mallas estructuradas de 4x4, 8x8, 16x16 y 32x32.

El valor teórico del desplazamiento vertical en el punto A reportado por la literatura es de 0.3024. En la Figura 3 se presentan los valores obtenidos para dicho desplazamiento bajo las distintas combinaciones de las variables analizadas. Por su parte, en la Figura 4 se muestra la configuración deformada para el caso de un elemento de 9 nodos, con integración completa y malla de 32x32 elementos, el cual, según se desprende del gráfico de la Figura 3, es uno de los modelos considerados que mejor comportamiento presenta.

A partir de los valores mostrados en el gráfico de la Figura 3, se puede notar que los elementos de 8 y 9 nodos presentan comportamientos similares entre sí, los cuales no dependen tanto de la densidad de malla o del esquema de integración numérica adoptado. A su vez, se observa que los elementos de 4 nodos con integración completa presentan un comportamiento muy rígido; mientras que, por el contrario, aquellos elementos con integración reducida en las matrices de membrana y corte presentan un comportamiento demasiado flexible, especialmente para bajas densidades de malla.

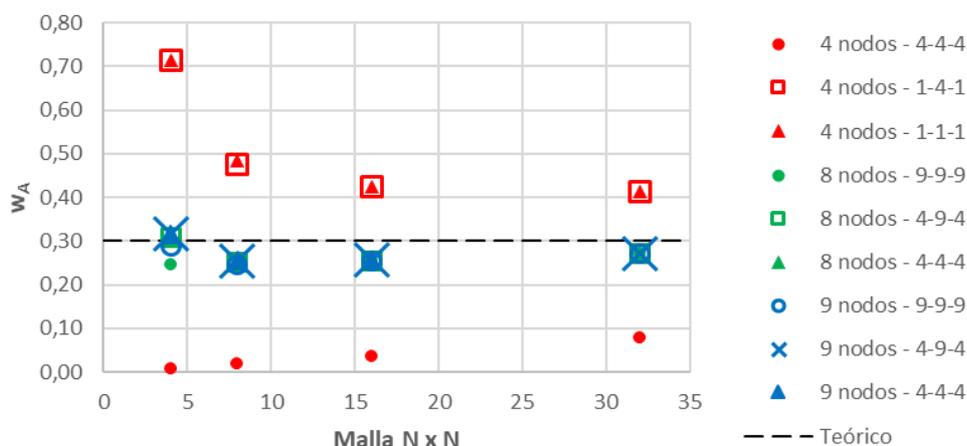


Figura 3: Desplazamiento en el punto A para discretizaciones con elementos con distinta cantidad de nodos y diferentes esquemas de integración numérica<sup>3</sup>

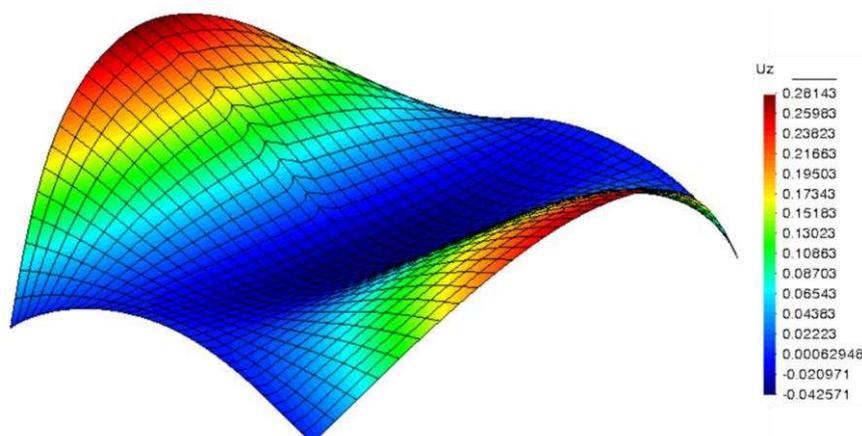


Figura 4: Configuración deformada para un modelo con elementos de 9 nodos, con integración completa y malla de 32x32 elementos.

De esta manera, los usuarios del software pueden utilizarlo para analizar la influencia que las distintas variables implícitas en el MEF pueden tener sobre los resultados.

### 3.2 Semiesfera con orificio y cargas puntuales

En este segundo ejemplo, extraído también de Celigüeta Lizarza (2011), se analiza una cáscara de forma semiesférica, de radio 10, con un orificio definido por una recta con inclinación de  $18^\circ$  respecto al eje vertical  $z$ , según se indica en la Figura 5. El material tiene módulo de elasticidad  $E = 6.825 \times 10^7$  y coeficiente de Poisson  $\nu = 0.3$ . La cáscara tiene un espesor de 0.04 y se encuentra sometida al esquema de cargas puntuales que se indica en la Figura 5. En la bibliografía se reporta que el valor teórico de desplazamiento radial en el punto A (ver Figura 5) es de 0.0935.

<sup>3</sup> En las leyendas de la Figura 3, los tres números separados por guiones representan, respectivamente, la cantidad de puntos de integración adoptada para el cálculo de las matrices de rigidez de membrana  $K_m$ , de flexión  $K_f$  y de corte  $K_c$ .

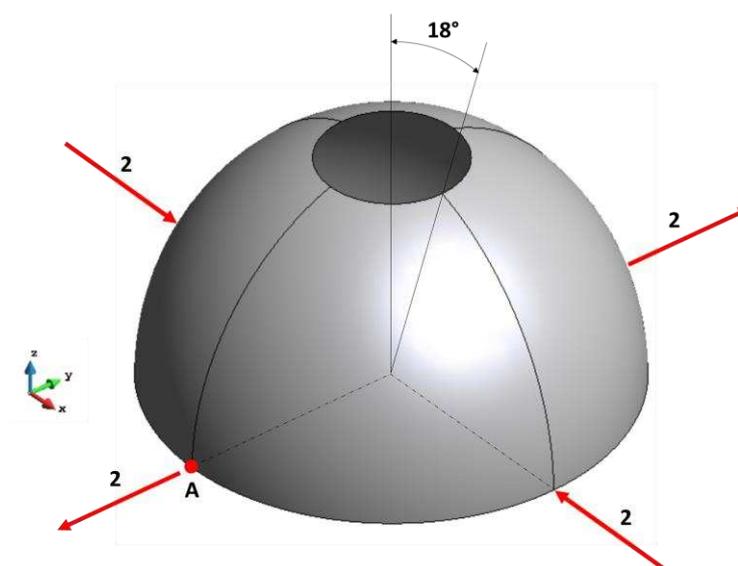


Figura 5: Semiesfera con orificio sometida a la acción de cargas puntuales

En la Figura 6 se presenta la configuración deformada de la estructura para una malla de  $8 \times 8$  elementos en cada octante, elementos de 4 nodos y dos esquemas de integración: completa (a) y reducida (b). Por un lado, en el caso con integración completa (a), el comportamiento es extremadamente rígido (desplazamiento radial en el punto A de 0.0003), lo cual es propio de elementos de 4 nodos basados en la teoría de Reissner-Mindlin utilizados para modelar estructuras tan esbeltas (en este caso, por ejemplo, la relación espesor/radio es de 0.004). En cambio, al analizar la deformada del modelo con integración reducida (b), el desplazamiento radial en el punto A (0.0915) se acerca mucho al valor teórico de referencia (0.0935); sin embargo, se observa que se presenta un fenómeno conocido en la literatura como *reloj de arena* (*hourglass* en inglés), originado por el esquema de integración reducida utilizado, el cual induce *modos de energía nula* en este tipo de elementos (Oñate, 1995; Irons, 1980). En este ejemplo se observa cómo, en un modelo que cuenta con pocas restricciones, este tipo de mecanismos puede propagarse a la totalidad de la malla, generando esa forma oscilante en la deformada y distorsionando los resultados.

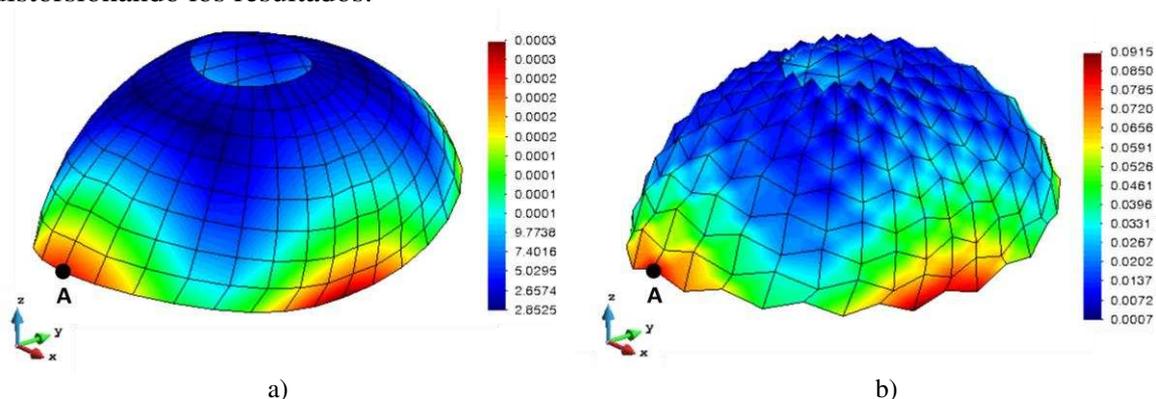


Figura 6: Configuración deformada. Discretización con elementos de 4 nodos con integración completa (a) y reducida (b)

De esta manera, se observa que el desarrollo presentado es una herramienta que permite captar en forma sencilla el comportamiento de distintas estructuras de cáscara como las ejemplificadas, posibilitando una mejor comprensión de la influencia que pueden tener las variables implícitas en la formulación de los elementos en los resultados numéricos obtenidos,

y ayudando a la formación del juicio crítico a la hora de la interpretación de resultados de modelos estructurales.

#### 4 DESARROLLOS FUTUROS

El código desarrollado hasta el momento, presenta un gran potencial para ser extendido, incorporando nuevos elementos finitos de tipo cáscara. Hasta el momento, como se ha descrito en este trabajo, el programa utiliza *elementos de cáscara plana*. Sin embargo, fácilmente podrían incorporarse elementos de cáscara basados en formulaciones distintas, como, por ejemplo, elementos planos con campos de deformación cortante impuesta (Oñate, 1995), entre otros (Celigüeta Lizarza, 2011). En este momento continúa el desarrollo para incorporar el elemento *semiloof* (Irons, 1980).

#### 5 CONCLUSIONES

En este trabajo se ha desarrollado un software de carácter educativo para el análisis por el método de elementos finitos de estructuras de cáscara. El mismo permite la manipulación en forma explícita de distintas variables implícitas en la formulación del método y los elementos utilizados, entre ellas las funciones de forma y el esquema de integración numérica. Con este trabajo se pone al alcance del usuario la posibilidad de manipular en forma directa y sencilla las variables mencionadas, a fin de contribuir a una mejor comprensión de la influencia que dichas variables tienen en los resultados de simulaciones numéricas del comportamiento estructural.

#### 6 AGRADECIMIENTOS

Los autores agradecen el apoyo económico de la Facultad de Ingeniería de la Universidad Nacional de Cuyo para llevar adelante este proyecto de investigación y poder viajar a exponer el trabajo.

#### REFERENCIAS

- Celigüeta Lizarza, J.T. *Método de los Elementos Finitos para Análisis Estructural*. Campus Tecnológico de la Universidad de Navarra, 2011.
- Irons, B., and Ahmad, S. *Techniques of Finite Elements*. Ellis Horwood Limited, 1980.
- Oñate, E., *Cálculo de Estructuras por el Método de Elementos Finitos. Análisis Estático Lineal*. CIMNE, 1995.
- Zera, S., López, D.E., y Leiva C.J. Software para la Evaluación de la Influencia de la Formulación de Elementos Finitos en la Simulación del Comportamiento de Estructuras Simples. *Congreso Argentino de Mecánica Computacional*, Volumen XL, 42:1587–1596, 2023.
- Zienkiewicz, O.C., and Taylor, R.L. *The Finite Element Method, Volume I*. Elsevier Butterworth-Heinemann, 2005.