

USO DE MALLAS DE TRIÁNGULOS ADAPTIVAS Y REFINAMIENTO LOCAL APLICADO A LA SIMULACIÓN DEL CRECIMIENTO DE UNA GRIETA

David I. Azócar^a, María Cecilia Rivara^b y Marcelo E. Elgueta^a

^a*Departamento de Ingeniería Mecánica, Universidad de Chile, Av. Beauchef 850, Santiago, Chile, dazocar@ing.uchile.cl, <http://dimec.uchile.cl>*

^b*Departamento de Ciencias de la Computación, Universidad de Chile, Av. Blanco Encalada 2120, Santiago, Chile, mcrivara@dcc.uchile.cl, <http://www.dcc.uchile.cl>*

Palabras clave: Fractura, Elementos finitos, Mallas Geométricas, LEPP.

Resumen. En este trabajo se muestra el uso de metodologías de mallado en dos dimensiones, para abordar un problema de la mecánica de la fractura, utilizando mallas de triángulos. El problema a resolver es el de un sólido elástico isotrópico sujeto a una carga constante y con una grieta en modo I de fractura, problema que se enmarca en la teoría de la mecánica de la fractura elástica lineal (MFEL). Para facilitar el trabajo, se ha desarrollado un software que permite el refinamiento de la malla en la zona cercana a la punta de la grieta, la fijación de condiciones de borde y de desplazamiento, y la generación de un archivo de salida con toda la información necesaria para encontrar el campo de esfuerzos y deformaciones por medio del método de elementos finitos utilizando el software comercial ABAQUS. Una vez obtenido este resultado, se realiza un post-proceso para calcular el valor del factor de intensificación de esfuerzos de la grieta y se modifica la malla original para simular el avance de dicha grieta. Para el refinamiento de la zona cercana a la punta de la grieta, se utilizan algoritmos de refinamiento LEPP-Bisección y LEPP-Delaunay. Los resultados se comparan con soluciones analíticas, así como con los obtenidos numéricamente utilizando elementos isoparamétricos en los cuales los elementos en la punta de la grieta son modificados desplazando el nodo medio en 1/4 de su posición original (quarter point elements), que es el método utilizado normalmente en los softwares CAE comerciales. El objetivo a largo plazo de este trabajo es desarrollar una herramienta que permita simular el crecimiento por fatiga de una grieta orientada al azar, simulando su avance y cambio de dirección si existe, utilizando mallas adaptivas.

1. INTRODUCCIÓN

En muchas aplicaciones de ingeniería, componentes estructurales deben ser sometidos a muy altos esfuerzos, pero minimizando su tamaño. Para lograr eso, se han desarrollado aleaciones con altos límites de fluencia, como por ejemplo aceros de alta resistencia, aleaciones base níquel o base titanio, etc. La forma en que fallan estos materiales es muy distinta a la de un acero estructural convencional. En estos casos no se produce deformación plástica, sino que el material falla por fractura frágil de una grieta ya existente, o que eventualmente se propagó por fatiga. El parámetro relevante al diseñar con estos materiales no es el límite de fluencia, sino la tenacidad a la fractura (K_{IC}). El procedimiento de diseño consiste en identificar por algún medio el tamaño de las grietas existentes, calcular el valor del factor de intensificación de esfuerzos (SIF, por sus siglas en inglés) en modo I (K_I) para dicha grieta, y verificar si superará el valor de K_{IC} del material seleccionado, en condiciones de operación. Para configuraciones geométricas sencillas, el valor de K_I puede ser obtenido por expresiones analíticas disponibles en la literatura, pero para geometrías más complejas el cálculo no es trivial, y debe ser hecho numéricamente.

El valor de K_I se puede obtener a partir de su relación con los campos de desplazamientos, esfuerzos y deformaciones cerca de la punta de la grieta, en el contexto de la teoría de la mecánica de la fractura elástica lineal (MFEL). Para obtener estos campos se utiliza por ejemplo el método de elementos finitos (Zienkiewicz y Taylor, 1991), en el cual a partir de una discretización del espacio utilizando una malla geométrica, se obtiene el valor de los desplazamientos en ciertos nodos asociados a la malla, los cuales se utilizan para obtener el campo en todo el espacio por medio de una extrapolación.

Cerca de la grieta, el campo de desplazamientos es proporcional a \sqrt{r} (Anderson, 1995), donde r es la distancia desde la punta de la grieta. En esta zona existen grandes variaciones (gradientes) de los campos. Por otro lado, el error local de la aproximación del campo obtenida por el método de elementos finitos depende del tamaño de los elementos en esa zona. Por lo tanto, al usar una malla cerca de la punta de la grieta del mismo tamaño que en el resto del cuerpo, en esta zona se obtendrá una aproximación con mucho error, y por ende, el valor de K_I obtenido también lo tendrá. Para solucionar este problema, existen básicamente dos opciones: crear una malla muy fina cerca de la punta de la grieta, o la utilización de elementos especiales que aproximen mejor la forma (\sqrt{r}) del campo.

En este artículo se muestra una metodología de cálculo de K_I , de refinamiento localizado de una grieta, y de simulación del crecimiento de una grieta de un problema de la mecánica de la fractura. Para eso se utiliza una aplicación computacional que permite la generación de un modelo de elementos finitos de un material sólido elástico isotrópico 2D que incluye una grieta en su interior, utilizando elementos finitos triangulares de segundo orden (6 nodos). El modelo creado se solucionará utilizando el software comercial ABAQUS, para luego mediante un post proceso obtener el valor de K_I . Estos valores se comparan con una solución analítica (Anderson, 1995), además de la solución obtenida utilizando los elementos especiales para grietas, incluidos en ABAQUS y que son los comúnmente utilizados. Para el refinamiento y la simulación del crecimiento de la grieta, se utilizan los Algoritmos basados en el LEPP, desarrollados por Rivara (1997).

2. FACTORES DE INTENSIFICACIÓN DE ESFUERZOS

La teoría de la mecánica de la fractura elástica lineal (MFEL) (Kanninen y Popelar, 1985; Anderson, 1995) modela la fractura suponiendo que no existe deformación plástica antes de la falla, es decir el material al alcanzar su límite de fluencia se fractura en forma frágil. En la

realidad siempre ocurre un poco de deformación plástica antes de la falla, pero en materiales con alto límite de fluencia, esta deformación plástica es muy pequeña y localizada en la punta de la grieta, así que para efectos del modelo, puede considerarse nula.

Según la MFEL, cerca de la grieta, el campo de desplazamientos es de la forma

$$u(r, \theta) = K \sqrt{\frac{r}{2\pi}} f(\theta) \quad (1)$$

La función $f(\theta)$ depende de la geometría de la configuración elegida. El factor K , llamado factor de intensificación de esfuerzos (SIF, por sus siglas en inglés), como su nombre lo dice, da una medida de cuánto aumenta el esfuerzo cerca de la grieta. Este factor depende del estado de esfuerzos y del tamaño de la grieta. En el caso general, una grieta puede experimentar tres modos de carga, llamados modo I, II y III. En el modo I la carga principal se aplica normal a la grieta, y tiende a abrirla; en el modo II, la carga es de corte, y tiende a deslizar una cara de la grieta con respecto a la otra; y en el modo III, la carga es de torsión. En consecuencia, un estado de esfuerzos general en una grieta puede describirse como una combinación de estos modos

$$u(r, \theta) = K_I \sqrt{\frac{r}{2\pi}} f_I(\theta) + K_{II} \sqrt{\frac{r}{2\pi}} f_{II}(\theta) + K_{III} \sqrt{\frac{r}{2\pi}} f_{III}(\theta) \quad (2)$$

Para efectos de su uso en ingeniería, K_I tiene una gran importancia. Si para una configuración dada, $K_I > K_{IC}$, el material fallará por fractura frágil. El parámetro K_{IC} se conoce como tenacidad a la fractura, y es una propiedad del material.

3. TÉCNICAS DE CÁLCULO DE LOS FACTORES DE INTENSIFICACIÓN DE ESFUERZOS

Para obtener el valor de K_I , se necesita conocer los campos cerca de la grieta. Para obtenerlos, una de las técnicas más utilizadas es el método de elementos finitos. Una vez obtenidos los campos, a partir de ellos y por medio de un post proceso, se obtienen los factores de intensificación de esfuerzos. Existen variadas metodologías para este post proceso, las más importantes se mencionan a continuación.

3.1. Extrapolación de desplazamientos

El método de extrapolación de desplazamientos consiste en calcular los SIF (en particular K_I) a partir del desplazamiento de los nodos cerca de la punta de la grieta. A partir de la MFEL (Kanninen y Popelar, 1985), se obtiene una expresión analítica que relaciona K_I y los desplazamientos, utilizando el sistema de referencia de la figura 1

$$\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \frac{K_I}{2\mu} \sqrt{\frac{r}{2\pi}} \begin{Bmatrix} \cos(\theta/2)[\kappa - 1 + 2\sin^2(\theta/2)] \\ \sin(\theta/2)[\kappa + 1 - 2\cos^2(\theta/2)] \end{Bmatrix} \quad (3)$$

Donde $\mu = \frac{E}{2(1+\nu)}$ es el módulo de corte, $\kappa = \frac{3-\nu}{4-\nu}$ para esfuerzo plano y $\kappa = 3 - 4\nu$ para deformación plana. Esta expresión es válida para cualquier valor de θ . Para facilitar los cálculos, lo más frecuente es evaluar la expresión en las caras de la grieta. En este caso, $\theta = \pi$, y el factor de intensificación de esfuerzos se calcula como

$$\lim_{r \rightarrow 0} \frac{2\mu}{\kappa + 1} \sqrt{\frac{2\pi}{r^2}} u_2 \quad (4)$$

Muy cerca de la grieta, puede considerarse que el estado de esfuerzos es de deformación plana. En este caso, la expresión se reduce a

$$K_I = \frac{E}{4(1-\nu^2)} \sqrt{\frac{2\pi}{r}} u_2 \quad (5)$$

Para mayor precisión, esta expresión se evalúa en los nodos. El valor buscado corresponde a $r = 0$, sin embargo el nodo correspondiente a $r=0$ tiene mucho error. Como solución a esto, se grafican valores de K versus r en los nodos, y se obtiene el valor buscado extrapolando linealmente estos valores hasta $r=0$.

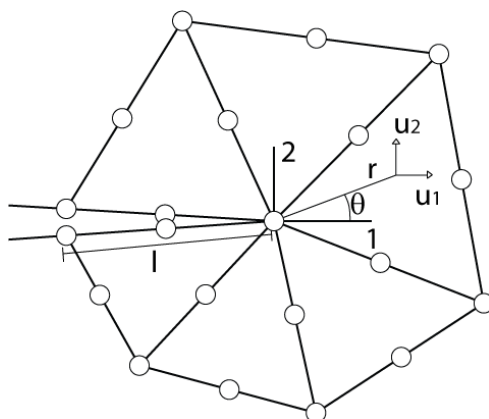


Figura 1: Elementos triangulares y sistema de coordenadas usado para describir el campo cerca de la punta de la grieta.

3.2. Integral J

El método de la integral J se basa en la energía involucrada en el crecimiento de la grieta. La expresión para la integral J (Rice, 1968) es

$$J = \int_{\Gamma} \left(U dy - t \frac{\partial d}{\partial x} ds \right) \quad (6)$$

Donde Γ es un camino que parte en la superficie inferior de la grieta, y llega a la superior, encerrando a esta; U es la densidad de energía de deformación; t es el vector de tracción; y d es el vector de desplazamientos. Esta integral es independiente del camino (Rice, 1968). Además, en el caso de la MFEL, el valor es idéntico a la tasa de liberación de energía de deformación $d\Pi/dA$, donde A es el área de la superficie de la grieta (Kanninen y Popelar, 1985). Por otro lado, esta expresión está relacionada con los factores de intensificación de esfuerzos

$$J = \frac{K^2}{E} \quad (7)$$

para esfuerzo plano, y

$$J = \frac{K^2}{E} (1 - \nu^2) \quad (8)$$

para deformación plana.

Por lo tanto, calculando numéricamente dicha integral se puede obtener el factor de intensificación de esfuerzos. La precisión de este cálculo, así como su independencia del camino,

depende de la precisión con la que se obtengan los campos. Por lo tanto las consideraciones sobre la malla cerca de la grieta siguen siendo las mismas. Aunque para un problema equivalente, el método de la integral J necesita una malla menos fina que el método de extrapolación de desplazamientos (Guinea et al., 2000). Su desventaja consiste en que se deben calcular los esfuerzos a partir de los valores nodales, en cambio en el método de extrapolación de desplazamientos, los valores nodales son evaluados directamente (Guinea et al., 2000).

4. MALLAS GEOMÉTRICAS

Una malla es una discretización geométrica formada por polígonos conectados entre sí. En el caso del método de elementos finitos, las mallas son utilizadas para discretizar una región geométrica en un grupo de elementos. Los tipos de mallas más utilizados (en 2D) son las mallas de triángulos y las mallas de cuadriláteros. En este trabajo, se utilizan elementos finitos triangulares de segundo orden, con 6 nodos, como muestra la figura 1. Un elemento de segundo orden significa que la interpolación del campo a partir de los valores nodales al resto del cuerpo se hará con funciones cuadráticas.

4.1. Elementos en la punta de la grieta

Dado que el método se basa en una discretización del objeto geométrico, la exactitud de la solución encontrada dependerá de la densidad de nodos. En las zonas donde el campo a calcular varía demasiado (como cerca de la punta de una grieta) la malla debe ser refinada. Por otra parte se sabe que la convergencia del método de elementos finitos requiere que no existan ángulos muy agudos en la malla (Zienkiewicz y Taylor, 1991). Si una malla tiene muchos elementos con ángulos pequeños, se dice que es de “mala calidad”.

El campo de desplazamientos cerca de la punta de una grieta varía como \sqrt{r} . Para obtener un nivel de error razonable, es necesario utilizar en esta zona una malla más fina que en el resto del cuerpo. En los inicios del desarrollo del método de elementos finitos, el refinamiento localizado era un problema no trivial. Barsoum (1976) introdujo un tipo de elemento especial, en el cual los nodos medios de los elementos que están en contacto con la punta de la grieta son desplazados 1/4 de su posición original. Con esto se genera una función de interpolación de la misma forma que el campo en esta zona, lo que mejora la interpolación. Esto puede hacerse con elementos isoparamétricos de 8 nodos (Barsoum, 1976), o en elementos triangulares de 6 nodos (Barsoum, 1977). En la figura 2 se ven las tres opciones posibles para la malla cerca de la punta de la grieta.

Al usar los elementos especiales, se puede obtener un error equivalente al obtenido con una malla normal, pero con una malla menos fina, lo que implica un ahorro de grados de libertad, además de generar una malla localizada sencilla. Esta técnica se utiliza hasta el día de hoy. A partir de la década del '80 (Rivara, 1984) comienzan a desarrollarse algoritmos para refinar mallas de triángulos, lo que facilita la discretización de geometrías complejas. Esto, sumado al desarrollo exponencial de las capacidades de computación, hace que el refinamiento localizado cada vez sea un problema menos complicado.

4.2. Triangulación Delaunay

Una triangulación (en 2D) (Goodman y O'Rourke, 1997) es una partición de un dominio geométrico, como un conjunto de puntos o un polígono, en triángulos que sólo se tocan en sus lados compartidos. Una triangulación T se dice que es Delaunay si para cualquier triángulo $t \in T$, el círculo circunscrito de t no encierra ningún otro triángulo de T (Goodman y O'Rourke,

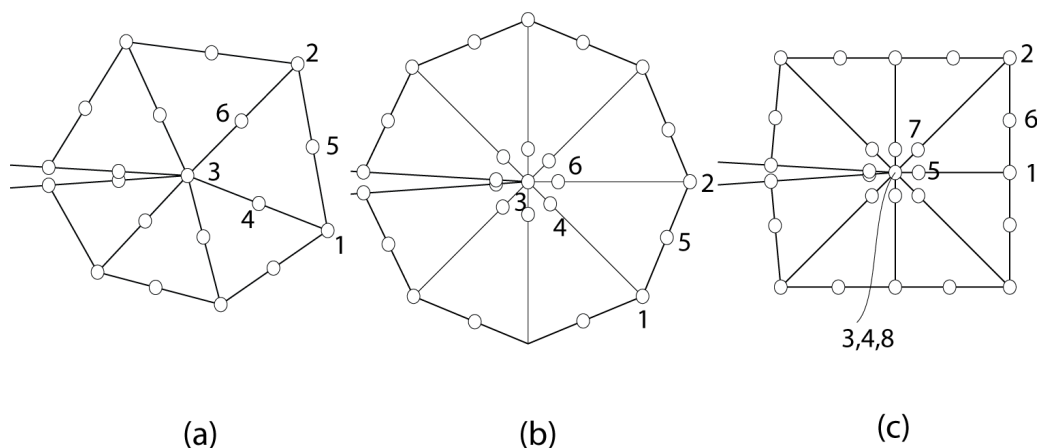


Figura 2: Elementos utilizados para la punta de la grieta. (a) Elemento triangular normal de 6 nodos; (b) Elemento triangular de 6 nodos con nodos medios desplazados; y (c) Elemento isoparamétrico de 8 nodos, con nodos medios desplazados $1/4$ de su posición y los 3 nodos de la punta colapsados.

1997). En la figura 3 se muestra una triangulación Delaunay. La propiedad mencionada permite implementar algoritmos de generación de mallas Delaunay, en los cuales se generan nuevos triángulos a partir de la inserción de nodos, y se verifica utilizando un *test del círculo circunscrito* si los nuevos triángulos generados mantienen la malla Delaunay. Si no lo son, estos se modifican para que la triangulación siga siendo Delaunay. Las triangulaciones Delaunay ya han sido usadas satisfactoriamente para calcular factores de intensificación de esfuerzos y simular la propagación de grietas (Phongthanapanich y Dechaumphai, 2004; Bittencourt et al., 1996).

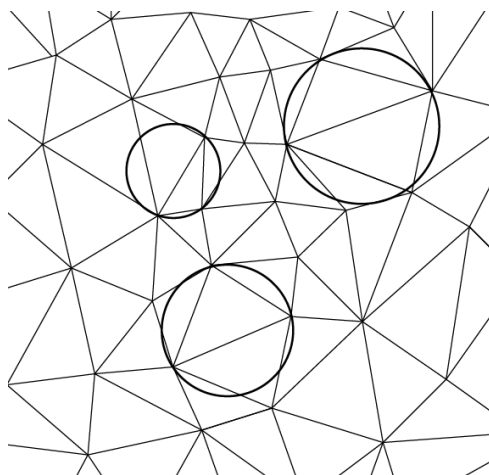


Figura 3: Triangulación Delaunay T de un conjunto de puntos. En una triangulación Delaunay, dado un triángulo t , el círculo circunscrito a t no debe tener en su interior a ningún otro triángulo de T .

4.3. Algoritmos de refinamiento LEPP

Dentro de los distintos procedimientos de refinamiento, existen los llamados algoritmos LEPP (Longest Edge Propagation Path) desarrollados por Rivara (1997). Al refinar un elemento, normalmente no sólo ese elemento es modificado sino algunos de los elementos que están cerca de él, para mantener una malla válida de buena calidad.

Para un triángulo t_0 en una triangulación T , se define el LEPP de t_0 como la lista ordenada de todos los triángulos $t_0, t_1, t_2, \dots, t_{n-1}, t_n$ tales que t_i es el triángulo vecino de t_{i-1} por el lado más

largo de t_{i-1} , para $i = 1, 2, \dots, n-1, n$. Una ilustración de lo anterior puede verse en la figura 4. Al refinar, sólo los elementos que están en el LEPP serán afectados. Estos algoritmos garantizan que siempre existirá una malla válida mientras se refina. Existen básicamente 3 variaciones de este algoritmo:

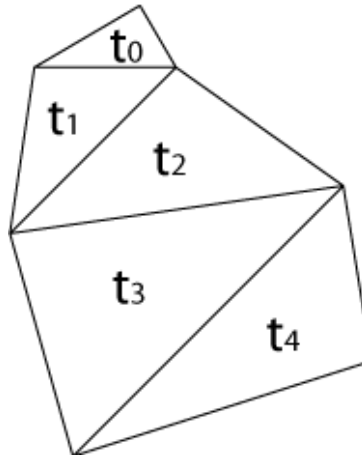


Figura 4: “Longest Edge Propagation Path (LEPP)”. t_0 es el triángulo a refinar, y los triángulos incluidos en el LEPP son los que se verán afectados por el refinamiento, para mantener una malla conforme. La última arista más larga se denomina arista terminal.

LEPP-Delaunay El algoritmo LEPP-Delaunay realiza sucesivamente la inserción Delaunay de un punto en el punto medio de la arista terminal, hasta que el triángulo objetivo es refinado. Este algoritmo mejora la calidad de la malla (la cual depende de la triangulación inicial). En la figura 5 se observa la secuencia de refinamiento del algoritmo.

LEPP-Delaunay(t, T)

Mientras t no sea modificado

 Encontrar el LEPP de t

 Realizar una inserción Delaunay del punto p (punto medio de la arista terminal del LEPP)

LEPP-Centroide En este caso, los puntos son insertados en el centroide del cuadrilátero formado por los dos últimos triángulos del LEPP, y luego se verifica que los nuevos triángulos sean Delaunay. El hecho de insertar el punto en el centroide y no en el punto medio de la arista común, trae como consecuencia que malla refinada sea de mejor calidad que la anterior, por lo que este algoritmo también se utiliza para mejorar la calidad. En la figura 6 se ilustra el algoritmo.

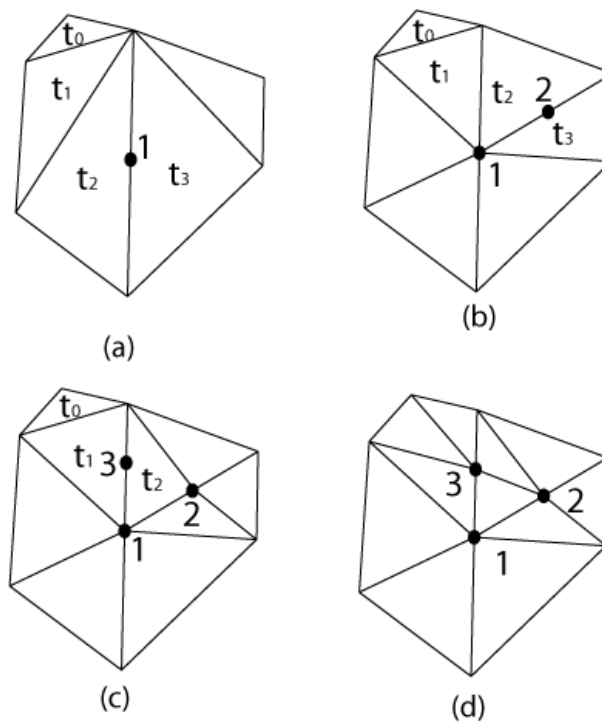


Figura 5: Refinamiento de un triángulo t_0 utilizando el algoritmo LEPP-Delaunay. En el Algoritmo LEPP-Delaunay, después de insertar un punto se verifica que los triángulos generados sigan siendo Delaunay. Las inserciones continúan hasta que el triángulo objetivo es modificado.

LEPP-Centroide(t, T)

Mientras t no sea modificado

 Encontrar el LEPP de t

 Realizar una inserción Delaunay del punto p

 (centroide del cuadrilátero formado por dos últimos triángulos del LEPP)

LEPP-Bisección En este caso, simplemente se biseccionan repetitivamente los triángulos que comparten la arista terminal asociada al LEPP, sin comprobar si los nuevos triángulos son Delaunay, hasta que el triángulo objetivo se refina. Este algoritmo no mejora la calidad de la malla, además de insertar más puntos, pero tiene la ventaja de generarse un mallado más rápido, puesto que no se hace ninguna comprobación si existen triángulos no Delaunay. Por lo tanto es apropiado para generar mallas en modelos muy grandes. Además permite implementar, en forma más simple que en los otros casos, un algoritmo de desrefinamiento sobre mallas generadas con este algoritmo. La figura 7 ilustra el refinamiento de un triángulo t_0 utilizando el algoritmo LEPP-Bisección.

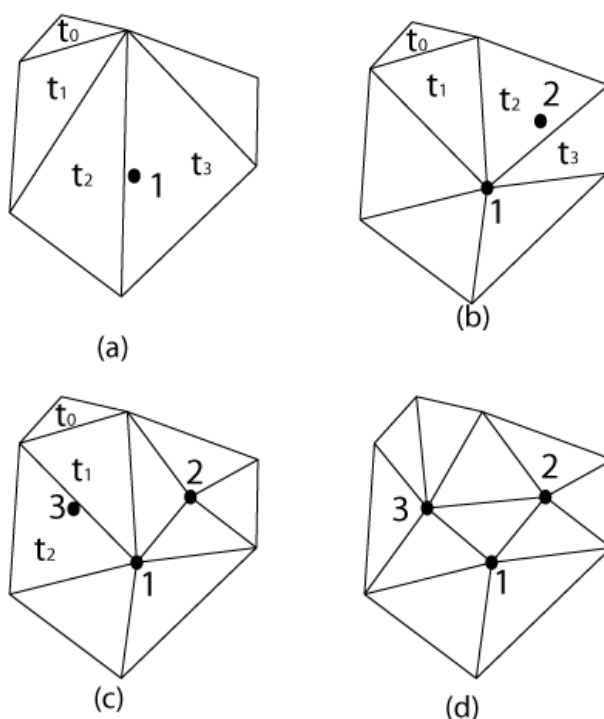


Figura 6: Refinamiento de un triángulo t_0 utilizando el algoritmo LEPP-Centroide. La diferencia con el algoritmo LEPP-Delaunay consiste en que el punto no se inserta en el punto medio de la arista terminal, sino en el centroide del cuadrilátero formado por los dos triángulos terminales.

LEPP-Biseccion(t,T)

Mientras t permanezca sin ser biseccionado

 Encontrar el LEPP de t

 Si t^* , el último triángulo del LEPP(t) es un triángulo de borde,
 biseccionar t^*

 Sino, biseccionar el par de triángulos terminales del LEPP(t)

4.4. Software de mallado

Con el objetivo de estudiar la generación de mallas y sus aplicaciones en ingeniería, se ha desarrollado una aplicación gráfica que permite realizar estas acciones en forma amigable. Las mallas son representadas mediante una implementación orientada a objetos, trabajo hecho originalmente por Calderon y Rivara (2004). Esta aplicación permite generar mallas de triángulos para una geometría 2D poligonal, con huecos poligonales. El mallado de geometrías curvas se proyecta como un trabajo posterior. Las grietas, como caso particular de un hueco poligonal, también pueden ser modeladas geoméricamente en forma explícita. La implementación original, además de permitir la representación y visualización gráfica de una malla de triángulos, incluía módulos para el refinamiento de la malla, utilizando los algoritmos LEPP-Delaunay, LEPP-Generalizado, LEPP-Bisección y LEPP-Centroide. Permitía además, la posibilidad de guardar la información de la malla en un archivo de texto, con un formato propio del software.

A partir de esta implementación original, y siguiendo el esquema orientado a objetos, se agregaron módulos que permitieran generar un modelo de elementos finitos, es decir representar (y por consiguiente guardar en un archivo de salida) información concerniente al material a

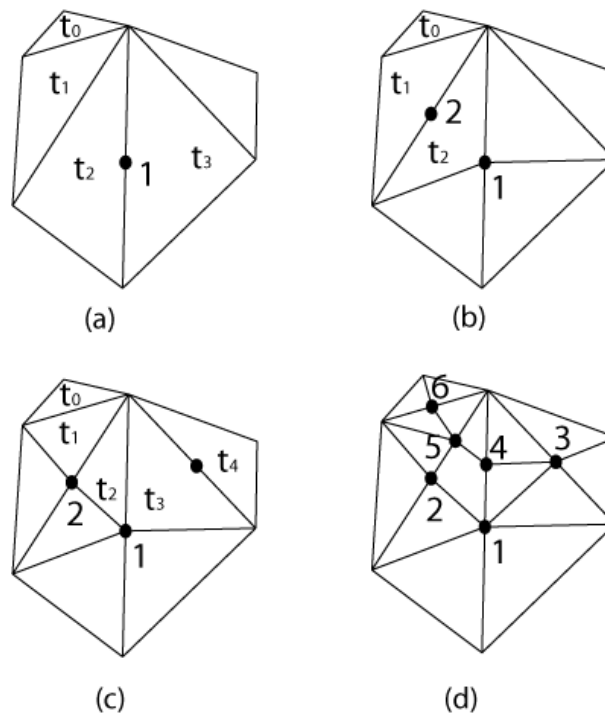


Figura 7: Refinamiento de un triángulo t_0 utilizando el algoritmo LEPP-Bisección. (a) triangulación inicial; (b) primer paso del proceso; (c) segundo paso del proceso; (d) triangulación final.

modelar, condiciones de desplazamiento y condiciones de carga. La implementación presentada en este texto, Permite modelar un sólido 2D cargado en forma estática. Con respecto al material, está limitada a sólidos elásticos isotrópicos. En cuanto a las condiciones de borde, pueden fijarse condiciones de desplazamiento constantes, así como condiciones de simetría, y cargas distribuidas constantes sobre una sección de la superficie del sólido. En la figura 8 se muestra una vista general del software.

5. RESULTADOS

A continuación se muestran los resultados de las pruebas hechas en este trabajo. Estas se dividen en dos partes, la primera corresponde al cálculo de K_I utilizando las mallas generadas con el software, y la segunda muestra la metodología utilizada para modificar la malla y así simular el avance de una grieta.

Para el cálculo de K_I , se prueba un caso utilizando condiciones de simetría, por lo que se simula sólo una cara de la grieta. Para refinar se utilizan los algoritmos LEPP-Delaunay y LEPP-Bisección. También se simula el mismo modelo pero sin simetría, refinando con el algoritmo LEPP-Delaunay. Para la simulación del crecimiento de la grieta, se prueban dos casos, uno utilizando el algoritmo LEPP-Delaunay, y otro utilizando el algoritmo LEPP-Bisección.

5.1. Cálculo del factor de intensificación de esfuerzos

El cálculo de K_I se hará sobre una configuración con solución analítica (Anderson, 1995) para poder validar los resultados. La configuración es la mostrada en la figura 9, el caso de una placa con una grieta en el centro. Dada la simetría, se resolverá 1/4 de la geometría. La solución

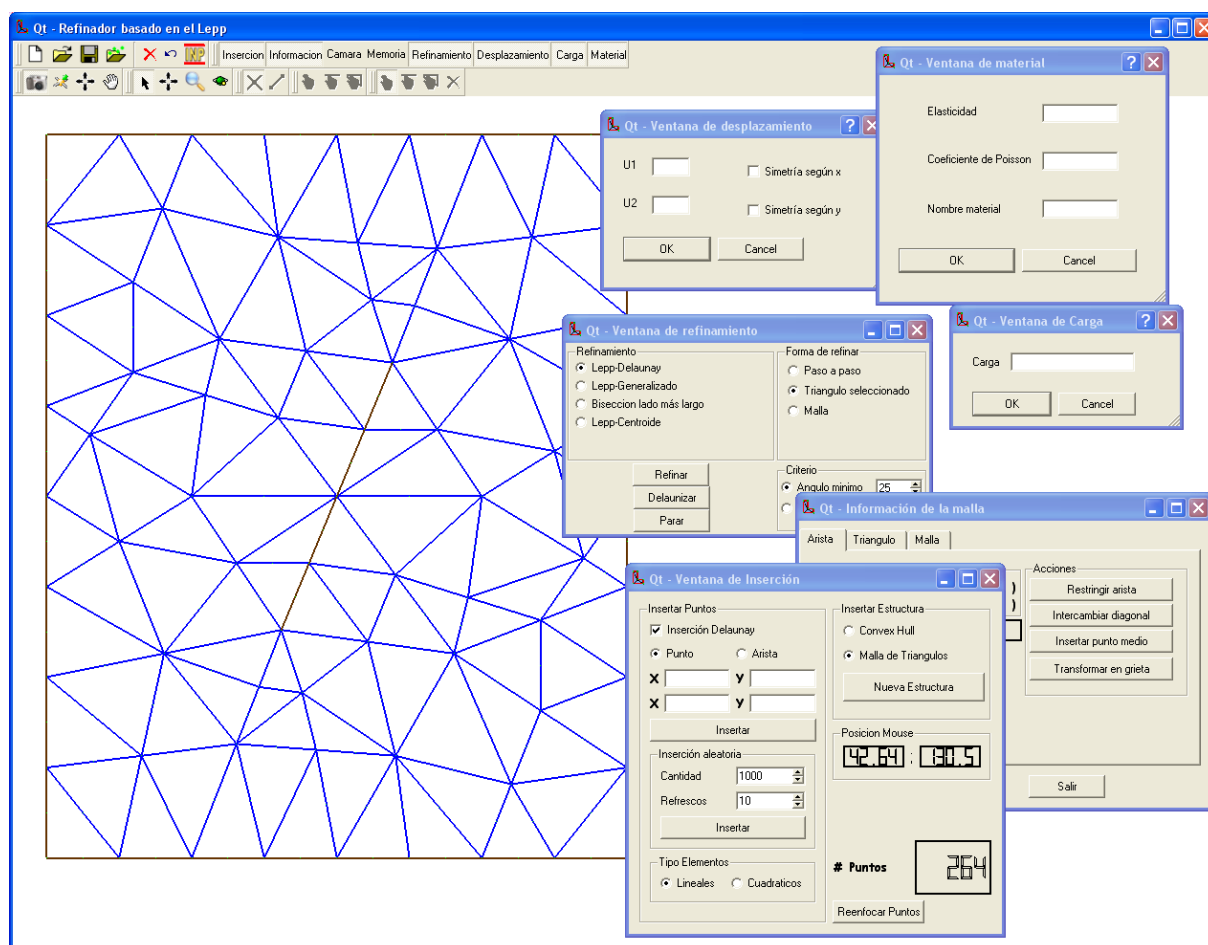


Figura 8: vista general del software.

analítica para este problema es

$$K_I = \sigma \sqrt{\pi a} \left[\sec \left(\frac{\pi a}{2W} \right) \right]^{0,5} \left[1 - 0,0025 \left(\frac{a}{W} \right)^2 + 0,06 \left(\frac{a}{W} \right)^4 \right] \quad (9)$$

Los valores de este problema son $a = 1[m]$, $W = 2[m]$, $E = 2 \cdot 10^{11}[Pa]$, $\nu = 0,3$ y $\sigma = 500 \cdot 10^6[Pa]$.

El caso con simetría se resolverá utilizando una malla generada con el software desarrollado, para varios niveles de refinamiento, como muestra la figura 10. El refinamiento se realizará con el algoritmo LEPP-Delaunay y con el algoritmo LEPP-Bisección. El cálculo de K_I se hará con el método de extrapolación de desplazamientos. Además, se calculará la misma configuración y con los mismos parámetros utilizando una malla isoparamétrica con elementos especiales generada por ABAQUS. Para calcular K_I , ABAQUS utiliza el método de la integral J. La malla utilizada se muestra en la figura 11. El primer nivel de refinamiento de la malla triangular tiene aproximadamente la misma cantidad de nodos que la malla isoparamétrica, esto con el objetivo de tener una idea de la diferencia que hay entre ambos métodos.

Además se resolverá la misma configuración pero sin utilizar la simetría. Esto con el objeto de probar el algoritmo de refinamiento en una grieta más genérica. La malla utilizada se muestra en la figura 12

Los resultados de la simulación, errores y parámetros se resumen en la tabla 1. La longitud l mencionada en la tabla es el largo de la arista que toca la cara de la grieta, como se muestra en

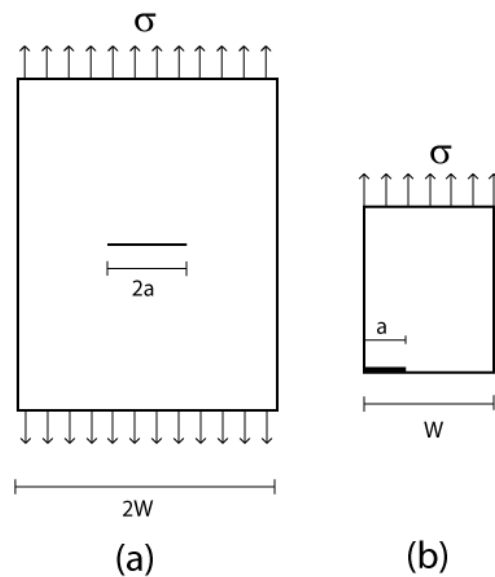


Figura 9: (a) El problema a resolver será el de una placa de ancho $2W$ con una grieta en el centro de tamaño $2a$. (b) Dada la simetría, es posible resolver sólo $1/4$ de la geometría.

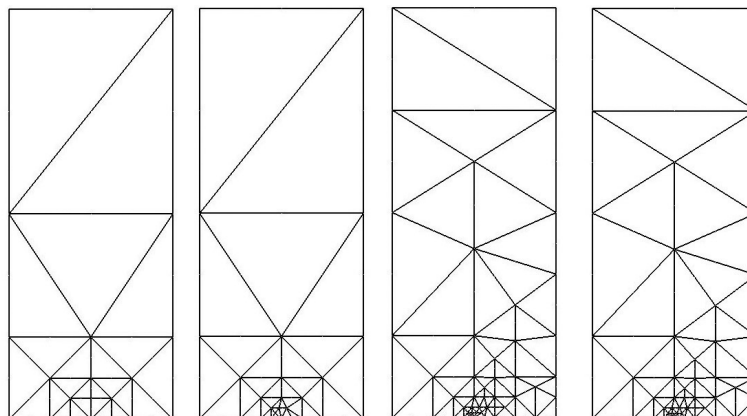


Figura 10: mallado utilizando $1/4$ de la geometría, utilizando distintos niveles de refinamiento. La malla más gruesa tiene aproximadamente la misma cantidad de nodos que la malla isoparamétrica con elementos especiales.

la figura 1.

Se observa en la tabla 1 que el método que utiliza elementos especiales obtiene un error de $0,21\%$. Se puede obtener un error equivalente utilizando elementos normales, refinando localmente la malla. El tamaño de los elementos especiales es de $l = 0,11[m]$, mientras que el tamaño con elementos normales es de $l = 0,015625[m]$, que es aproximadamente el 10% del elemento especial. Con este nivel de refinamiento se obtuvo un error de $0,1\%$. Para lograr este tamaño, el modelo que utiliza el algoritmo LEPP-Delaunay tiene más de tres veces la cantidad de nodos de la malla con elementos especiales, y al usar el algoritmo LEPP-Bisección el modelo tiene el doble de nodos.

5.2. Simulación del crecimiento de la grieta

Para modelar el crecimiento de una grieta, hay básicamente dos opciones: se puede rehacer completa la malla con la grieta más grande (Phongthanapanich y Dechaumphai, 2004), o modi-

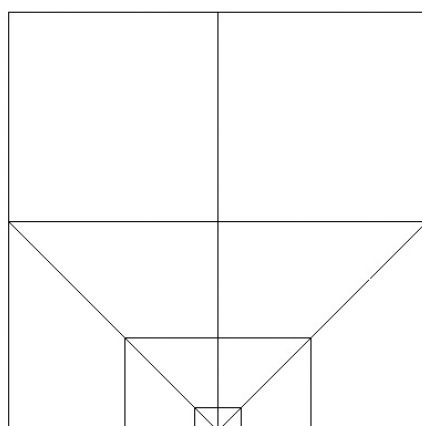


Figura 11: Malla generada con el software ABAQUS. Utiliza elementos isoparamétricos, con elementos especiales en la punta de la grieta. ABAQUS calcula K_I utilizando el método de la integral J.

CASO	Nodos	Elementos	l [m]	$K[MPa\sqrt{m}]$	Error (%)
Malla ABAQUS	59	14	0,111	1049,09	0,21
Malla triángulos, caso base	64	25	0,15	885,47	15,77
<i>LEPP-Delaunay</i>					
primer refinamiento	86	35	0,125	996,12	5,25
segundo refinamiento	175	76	0,0625	1069,42	1,7
tercer refinamiento	211	92	0,015625	1052,39	0,11
<i>LEPP-Bisección</i>					
primer refinamiento	82	33	0,125	890,09	15,33
segundo refinamiento	100	41	0,0625	975,2	7,24
tercer refinamiento	136	57	0,015625	1050,02	0,12
Grieta explícita	968	456	0,004	1047,92	0,32
Solución analítica				1051,27	0

Tabla 1: Resultados para el cálculo de K_I

ficar la malla original sólo en la zona donde la grieta crece (Bittencourt et al., 1996), que es lo que se hace en este trabajo.

El proceso de simulación del crecimiento de la grieta se divide en tres pasos: primero se inserta una arista en la malla de triángulos, que corresponde al nuevo segmento de la grieta; luego se transforma dicha arista en una grieta; y finalmente se modifica la malla en la zona colindante al nuevo segmento de la grieta, en caso de que la inserción de la arista genere elementos de mala calidad, o se necesite refinar en la nueva punta de la grieta. En este trabajo, se realizó dicho proceso, utilizando dos algoritmos distintos, el LEPP-Delaunay y el LEPP-Bisección. Para ambos casos, se partió de una geometría común, que es la que se muestra en la figura 13. Las dimensiones de la geometría son de $200[mm]$ de ancho y $100[mm]$ de alto, y la grieta tiene un largo de $10[mm]$. La malla está refinada en la punta de la grieta, que sería el caso si se estuviese simulando el crecimiento real de una grieta.

Se simuló una grieta que creció $0,6[mm]$ por paso, girando en 10° cada paso. Como criterio de refinamiento, se usó que la arista más larga de cada elemento que toca la punta de la grieta sea menor que $0,8[mm]$, que es el orden de magnitud de los elementos que existen en la punta

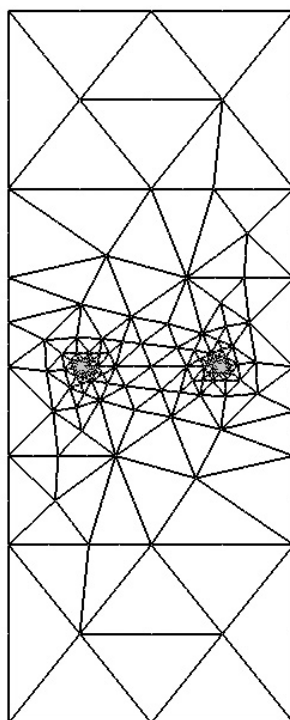


Figura 12: Mallado para la geometría completa, sin simetría.

antes de iniciar el crecimiento.

5.2.1. Crecimiento utilizando algoritmo LEPP-Delaunay

En el primer caso, se utilizó el algoritmo LEPP-Delaunay para refinar los elementos después del crecimiento, en caso de que fuesen más grandes que el criterio. No se realizó ninguna revisión de calidad de los elementos después del crecimiento. En la figura 14 se observa la malla resultante después del crecimiento. Los segmentos rojos corresponden a grieta propagada, y el segmento café corresponde a la grieta original.

5.2.2. Crecimiento utilizando algoritmo LEPP-Bisección

En este caso, se realizó lo mismo que en el caso anterior, pero se utilizó el algoritmo LEPP-Bisección para refinar la punta de la grieta después del crecimiento. En la figura 15 se observa la malla resultante después del crecimiento.

6. DISCUSIÓN

6.1. Cálculo del factor de intensificación de esfuerzos

La primera observación que se puede hacer al comparar los resultados del cálculo de K_I con los distintos métodos (tabla 1) proviene al comparar el caso base que utiliza triángulos normales con la malla con elementos especiales, dado que ambos casos tienen aproximadamente la misma cantidad de grados de libertad. Al utilizar los elementos especiales el error es de 0,21 %, y con los triángulos es de 15,77 %.

A medida que se refina la malla en la punta de la grieta, el error va disminuyendo paulatinamente, como era de esperarse. Se obtiene un error del mismo orden de magnitud que en el

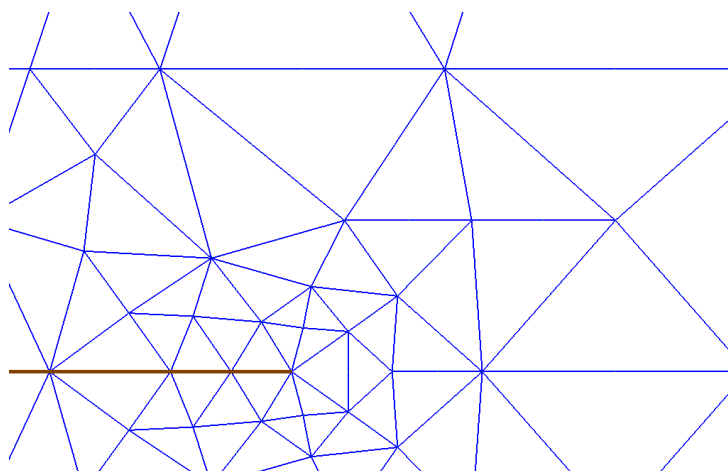


Figura 13: Mallado inicial para probar la metodología del crecimiento de la grieta. La malla está refinada localmente en la punta de la grieta, que sería lo que ocurriría en un crecimiento real. La línea café corresponde a la grieta.

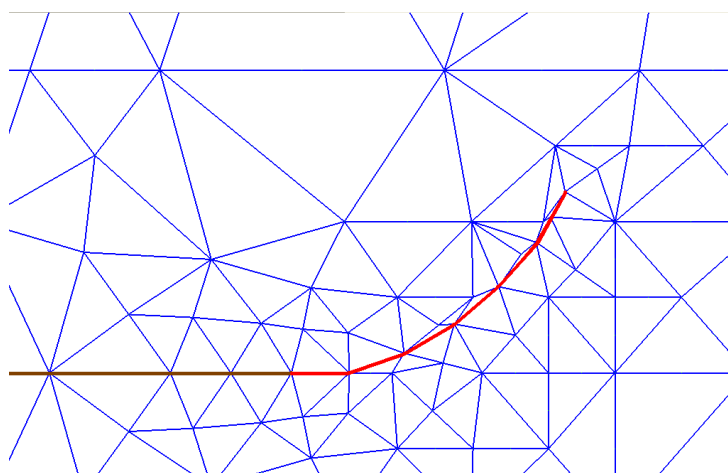


Figura 14: Malla resultante después del crecimiento de una grieta, utilizando el algoritmo LEPP-Delaunay. La grieta se hizo crecer en 6 pasos, de 0,6[mm] de largo, y cambiando de dirección en 10° cada paso. La línea café corresponde a la grieta original, y la línea roja a la propagación.

caso de los elementos especiales (0,11% después del tercer refinamiento) luego de disminuir el tamaño del elemento de $l = 0,15[mm]$ a $l = 0,015625[mm]$, es decir a un 10% del tamaño original, como muestra la tabla 1. La cantidad de nodos en el caso base es de 64. Después del tercer refinamiento al utilizar el algoritmo LEPP-Delaunay se obtienen 211 nodos, es decir más de 3 veces los nodos originales. Al utilizar el algoritmo LEPP-Bisección, se obtienen 136 nodos, poco más del doble. Como era de esperarse se necesitan más nodos para un error equivalente, si no se utilizan elementos especiales. Sin embargo, si se realizara el análisis de un caso real, donde las dimensiones de la grieta son mucho menores que las del cuerpo completo, el aumento relativo de grados de libertad es mucho menor, pues el refinamiento se hace en una zona muy localizada del modelo. El interés de utilizar elementos normales en vez de los elementos modificados radica en que a futuro se espera generalizar los algoritmos LEPP a tres dimensiones, pasando triángulos a tetraedros, dado que estos elementos se adaptan mucho mejor a geometrías complejas. La gran mayoría de los trabajos sobre cálculo de factores de intensificación de esfuerzos y propagación de grietas en 3D utilizan elementos isoparamétricos

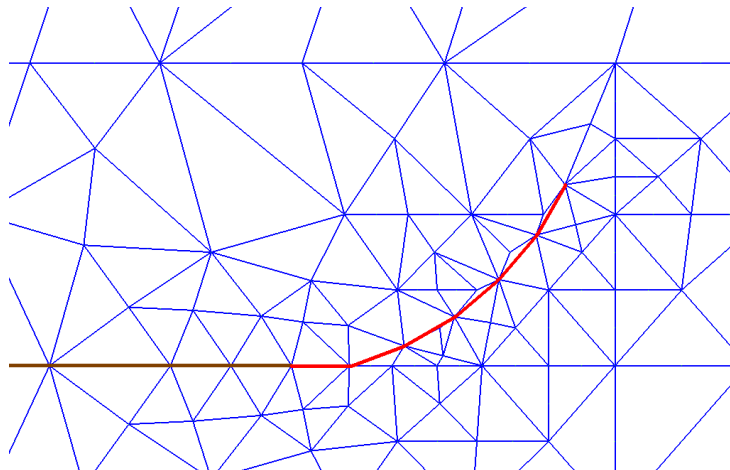


Figura 15: Malla resultante después del crecimiento de una grieta, utilizando el algoritmo LEPP-Bisección. La grieta se hizo crecer en 6 pasos, de $0,6[mm]$ de largo, y cambiando de dirección en 10° cada paso. La línea café corresponde a la grieta original, y la línea roja a la propagación.

prismáticos (“brick”) (Lin y Smith, 1999; Gosz y Moran, 2002; Okada et al., 2005), mientras que los trabajos utilizando tetraedros son escasos (Rajaram et al., 2000).

Además, se probó el mismo caso, pero utilizando el modelo completo, como muestra la figura 12. El objetivo de probar este caso no era el de corroborar la capacidad de cálculo, sino mostrar la capacidad del software desarrollado de poder generar una malla para un cuerpo con una grieta en su interior, y refinar localmente en ambas puntas. En la tabla 1 se ve que, como era de esperarse, se logró obtener un buen nivel de error refinando suficientemente las puntas de la grieta. Cabe destacar que deben refinarse ambas puntas de la grieta, dado que si en una punta la malla es demasiado gruesa, el efecto de ese error puede traspasarse a la otra punta, sobre todo si la grieta es pequeña, del mismo orden de magnitud que los elementos.

6.2. Simulación del crecimiento de la grieta

En las figuras 14 y 15 se muestran las mallas resultantes en la zona colindante a la grieta, después del crecimiento. Lo primero que puede observarse es la cantidad de nodos insertados. A simple vista se puede ver que el algoritmo LEPP-Delaunay inserta menos puntos que el algoritmo LEPP-Bisección. La malla inicial (figura 13) tiene 451 nodos. Después del crecimiento, con el algoritmo LEPP-Delaunay se obtienen 633 nodos, y con el LEPP-Bisección 677, es decir el algoritmo LEPP-Delaunay insertó 44 nodos más que el algoritmo LEPP-Bisección. Por lo tanto en lo que respecta a minimizar la inserción de nodos, el algoritmo LEPP-Delaunay es superior, de lo cual ya se tenía conocimiento (Rivara, 1997).

Otro punto importante tiene que ver con la calidad de las mallas generadas. En las figuras 14 y 15 se puede ver que ambos algoritmos generan algunos elementos de mala calidad a medida que la grieta avanza, siendo el algoritmo LEPP-Bisección más nocivo en este aspecto. En particular los elementos que van en la punta de la grieta están bastante distorsionados, lo cual puede afectar el error del cálculo de los SIF (Guinea et al., 2000). Cuando se simula el crecimiento de una grieta utilizando elementos triangulares, se desea que el nivel de refinamiento se conserve, puesto que en la grieta inicial ya se ha refinado lo suficiente para obtener un error adecuado. Esto bajo el supuesto que no es necesario refinar más a medida que la grieta crece. En el proceso de crecimiento de la grieta presentado en este trabajo, no se realiza mejoramiento de la calidad de la malla, puesto que hasta la fecha el software desarrollado sólo tiene implementado algoritmos

de refinamiento. Como se mencionó, los algoritmos LEPP-Delaunay y LEPP-Centroide pueden utilizarse para mejorar la calidad de una malla, pero en el caso del crecimiento de la grieta esto no es posible, puesto que no se desea que la malla se refine. Lo ideal sería implementar un método de reposicionamiento de nodos para ir mejorando la calidad de los elementos a medida que la grieta avanza. Esto se proyecta para un trabajo posterior.

De la discusión anterior se puede concluir que el algoritmo LEPP-Delaunay supera al algoritmo LEPP-Bisección tanto en la cantidad de nodos insertados como en la calidad de la malla generada. Sin embargo el algoritmo LEPP-Bisección tiene una ventaja interesante, puesto que se puede implementar fácilmente un algoritmo de des-refinamiento. En el caso de la propagación de una grieta, es necesario que en la zona de la punta de la grieta la malla sea fina, pero en el resto del cuerpo no. A medida que la grieta avanza, las zonas por las que ya pasó la punta de la grieta quedarán refinadas, por lo que un proceso de des-refinamiento en esa zona ayudaría a hacer más eficiente la metodología.

7. CONCLUSIONES

Se ha presentado un software para el desarrollo de metodologías de mallado y su aplicación a la mecánica de la fractura. Para mostrar su potencialidad, se ha utilizado para generar mallas con el fin de calcular K_I , refinar localmente la punta de una grieta, y simular el crecimiento de esta.

En lo que respecta al cálculo de K_I , se logró generar una malla suficientemente fina para calcular este parámetro utilizando elementos triangulares normales, tanto utilizando condiciones de simetría como mallando una grieta explícita. A partir de esto se proyecta un trabajo destinado a calcular modos combinados (K_I y K_{II}) para una grieta arbitraria. El interés de utilizar elementos triangulares normales y no elementos especiales, es porque a futuro se pretende generalizar este software a 3D utilizando mallas de tetraedros, pues los algoritmos LEPP pueden extenderse a tetraedros (Rivara y Palma, 1997). La generalización a 3D de los elementos especiales utiliza elementos prismáticos y no elementos tetraedros, por este motivo la mayoría de los autores utiliza este tipo de elementos al estudiar el cálculo de los factores de intensificación de esfuerzos y la propagación de grietas.

También se compararon dos procedimientos para simular el crecimiento de una grieta, uno usando el algoritmo LEPP-Delaunay, y otro usando el algoritmo LEPP-Bisección. Estos algoritmos se usan para refinar la nueva punta de la grieta, en caso de que esta presente elementos más grandes que los que habían antes del crecimiento. Al usar el algoritmo LEPP-Delaunay se obtienen elementos de mejor calidad, además de insertarse menos nodos que al usar el algoritmo LEPP-Bisección. De todas maneras, ambos algoritmos generan una cantidad considerable de elementos de mala calidad, por lo que es necesario mejorar esta metodología agregando un paso intermedio donde se revise y mejore la calidad de los elementos. Una buena opción es la de un reposicionamiento de los nodos. El algoritmo LEPP-Bisección presenta la ventaja de que puede implementarse en forma sencilla un algoritmo de des-refinamiento sobre una malla refinada.

Con la combinación de todas estas características se espera crear a mediano plazo un procedimiento robusto y eficiente de propagación de grietas por fatiga, y además generalizar a tres dimensiones.

Agradecimientos

Los autores agradecen a la Comisión Nacional de Investigación Científica y tecnológica, CONICYT, por el financiamiento de este trabajo, a través del proyecto Fondecyt 1040713.

REFERENCIAS

- T.L. Anderson. *Fracture Mechanics: fundamentals and applications*. CRC Press, 1995.
- R.S. Barsoum. On the use of isoparametric finite elements in linear fracture mechanics. *International Journal for Numerical Methods in Engineering*, 10:25–37, 1976.
- R.S. Barsoum. Triangular quarter-point elements as elastic and perfectly-plastic crack tip elements. *International Journal for Numerical Methods in Engineering*, 11:85–98, 1977.
- T.N. Bittencourt, P.A. Wawrzynek, A.R. Ingraffea, y J.L. Sousa. Quasi-automatic simulation of crack propagation for 2d lefm problems. *Engineering Fracture Mechanics*, 55:321–334, 1996.
- C. Calderon y M.C. Rivara. Herramienta computacional de apoyo a la investigación de algoritmos de triangulaciones. *Proceedings XII Jornadas Chilenas de Computación, Encuentro Chileno de Computación, Universidad de Tarapacá, Arica, 2004*.
- J.E. Goodman y J. O'Rourke. *Handbook of discrete and computational geometry*. CRC Press, 1997.
- M. Gosz y B. Moran. An interaction energy integral method for computation of mixed-mode stress intensity factors along non-planar crack fronts in three dimensions. *Engineering Fracture Mechanics*, 69:299–319, 2002.
- G. V. Guinea, J. Planas, y M. Elices. K_I evaluation by the displacement extrapolation technique. *Engineering Fracture Mechanics*, 66:243–255, 2000.
- M.F. Kanninen y C.H. Popelar. *Advanced Fracture Mechanics*. Oxford University Press, 1985.
- X.B. Lin y R.A. Smith. Finite element modelling of fatigue crack growth of surface cracked plates part i: The numerical technique. *Engineering Fracture Mechanics*, 63:503–522, 1999.
- H. Okada, M. Higashi, M. Kikuchi, Y. Fukui, y N. Kumazawa. Three dimensional virtual crack closure-integral method (vccm) with skewed and non-symmetric mesh arrangement at the crack front. *Engineering Fracture Mechanics*, 72:1717–1737, 2005.
- S. Phongthanapanich y P. Dechaumphai. Adaptive delaunay triangulation with object-oriented programming for crack propagation analysis. *Finite Elements in Analysis and Design*, 40:1753–1771, 2004.
- H. Rajaram, S. Socrate, y D.M. Parks. Application of domain integral methods using tetrahedral elements to the determination of stress intensity factors. *Engineering Fracture Mechanics*, 66:155–482, 2000.
- J.R. Rice. A path independent integral and the approximate analysis of strain concentrations by notches and cracks. *Journal of Applied Mechanics*, 35:376–86, 1968.
- M.C. Rivara. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering*, 20:745–756, 1984.
- M.C. Rivara. New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations. *International Journal for Numerical Methods in Engineering*, 40:3313–3324, 1997.
- M.C. Rivara y M. Palma. New lepp-algorithms for quality polygon and volume triangulation: implementation issues and practical behavior. *Trends in unstructured mesh generation, Eds: S.A. Cannan and S.Saigal, AMD-220:1–8, 1997*.
- O.C. Zienkiewicz y R.L. Taylor. *The finite element method*, volume I. McGraw Hill, 1991.