



**UNIVERSIDAD DE CHILE**  
**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**  
**DEPARTAMENTO DE INGENIERÍA MECÁNICA**

**APLICACIÓN DE LOS ALGORITMOS DE REFINAMIENTO LEPP EN LA  
PROPAGACIÓN DE GRIETAS**

**TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA  
MENCIÓN MECÁNICA**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL MECÁNICO**

**DAVID IGNACIO AZÓCAR PAREDES**

**PROFESOR GUÍA:  
MARCELO ELGUETA VERGARA**

**MIEMBROS DE LA COMISIÓN:  
MARÍA CECILIA RIVARA ZÚÑIGA  
DIEGO CELENTANO**

**SANTIAGO DE CHILE  
AGOSTO 2007**

RESUMEN DE LA TESIS  
PARA OPTAR AL TÍTULO DE  
MAGÍSTER EN CIENCIAS DE LA INGENIERÍA  
MENCION MECÁNICA  
POR: DAVID IGNACIO AZÓCAR PAREDES  
FECHA: 16/08/2007  
PROF. GUIA: Sr. MARCELO ELGUETA VERGARA

## **“APLICACION DE LOS ALGORITMOS DE REFINAMIENTO LEPP EN LA PROPAGACIÓN DE GRIETAS”**

En muchas aplicaciones de ingeniería es necesario utilizar materiales con alto límite de fluencia. En estos casos el material falla por fractura frágil, siendo la propiedad física relevante la tenacidad a la fractura, parámetro que marca el umbral para que se produzca la falla. Por otro lado, en una pieza sujeta a cargas cíclicas o variables una grieta existente que no es crítica eventualmente se propagará cuasi-estáticamente o por fatiga, pudiendo llegar a un valor crítico. En este caso interesa poder predecir el número de ciclos necesarios para llegar a un valor crítico, así como el camino que seguirá la grieta.

En este trabajo se presenta una metodología para resolver numéricamente el problema del cálculo de los factores de intensificación de esfuerzos y la propagación de una grieta en materiales sólidos isotrópicos elásticos bidimensionales (2D).

Para obtener los factores de intensificación de esfuerzos, mediante el método de elementos finitos se calcula el campo de desplazamientos del sólido. A partir de este resultado, los factores de intensificación de esfuerzos se calculan por medio del método de extrapolación de desplazamientos.

Para el manejo de las mallas se usa una herramienta interactiva basada en el algoritmo de Delaunay para obtener una malla inicial, y se usan los algoritmos Lepp-Delaunay para modificar/refinar la malla localmente en la grieta, mejorando la precisión del cálculo.

Las grietas son modeladas explícitamente, es decir como una entidad 1D (lineal) formada por dos superficies (curvas) libres que coinciden geoméricamente pero diferentes topológicamente. Luego del mallado, la grieta se representa como un conjunto de aristas conectadas entre sí. El crecimiento de la grieta se simula entonces agregando un nuevo segmento de grieta al final de ésta, modificando la malla geométrica original por medio de los algoritmos Lepp-Delaunay. Luego se simula en forma discreta la propagación de una grieta, resolviendo iterativamente los modelos de elementos finitos generados para cada paso.

Se logró implementar con éxito un método de cálculo de los factores de intensificación de esfuerzos, obteniendo errores del orden del 1 – 2% para varios problemas con solución conocida. Por otro lado, el método de propagación de grietas funcionó adecuadamente entregando el perfil del camino seguido por la grieta, validándose contra problemas con solución analítica y valores experimentales.

De los resultados obtenidos se concluye que los algoritmos LEPP-Delaunay tienen un gran potencial en la simulación de la propagación de grietas, debido a que permiten modificar localmente la malla a medida que la grieta avanza, ahorrándose el costo computacional del mallado completo entre un paso y otro de la propagación.

# Índice General

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Objetivos . . . . .	1
1.1.1	Objetivos generales . . . . .	1
1.1.2	Objetivos específicos . . . . .	1
1.1.3	Aspectos Metodológicos . . . . .	2
1.2	Estudio Bibliográfico . . . . .	2
1.2.1	Métodos numéricos en mecánica de la fractura . . . . .	2
1.2.2	Generación de mallas de triángulos . . . . .	5
<b>2</b>	<b>Mecánica de la Fractura</b>	<b>7</b>
2.1	Desarrollo histórico . . . . .	7
2.2	Mecánica de la Fractura Elástica Lineal . . . . .	8
2.2.1	Análisis de esfuerzos . . . . .	9
2.2.2	Análisis energético . . . . .	12
2.3	Propagación de grietas . . . . .	22
2.3.1	Ley de Paris . . . . .	22
2.3.2	Cambio de dirección de la grieta . . . . .	22
<b>3</b>	<b>Metodología numérica</b>	<b>25</b>
3.1	Elementos finitos . . . . .	25
3.1.1	Introducción . . . . .	25
3.1.2	Elementos finitos en la mecánica de sólidos . . . . .	26
3.1.3	Funciones de interpolación de los elementos . . . . .	29

3.1.4	Error . . . . .	31
3.2	Mallas geométricas . . . . .	32
3.2.1	Definiciones básicas . . . . .	32
3.2.2	Métodos de mallado . . . . .	34
3.2.3	Refinamiento . . . . .	35
3.2.4	Mejoramiento . . . . .	41
3.3	Técnicas de cálculo de los factores de intensificación de esfuerzos . . . . .	45
3.3.1	Elementos especiales con nodos desplazados . . . . .	46
3.3.2	Extrapolación de desplazamientos . . . . .	48
3.3.3	Integral J . . . . .	49
<b>4</b>	<b>Software</b>	<b>53</b>
4.1	Introducción . . . . .	53
4.2	Descripción general . . . . .	53
4.3	Implementación . . . . .	54
4.4	Funcionamiento . . . . .	59
4.4.1	Detalle de las funciones . . . . .	59
4.4.2	Pasos para crear y resolver un modelo . . . . .	65
4.5	Archivos de entrada y salida . . . . .	70
4.6	Post-proceso e interacción con ABAQUS . . . . .	73
<b>5</b>	<b>Resultados</b>	<b>77</b>
5.1	Introducción . . . . .	77
5.2	Mallado y refinamiento localizado . . . . .	77
5.2.1	Generación de mallas iniciales . . . . .	77
5.2.2	Refinamiento de la punta de la grieta . . . . .	80
5.2.3	Eliminación de triángulos malos . . . . .	87
5.3	Cálculo del factor de intensificación de esfuerzos . . . . .	87
5.3.1	Placa con grieta en el borde . . . . .	89
5.3.2	Grieta centrada . . . . .	91
5.3.3	Grieta en una muesca con forma de V . . . . .	91
5.3.4	Grieta en ángulo . . . . .	95
5.4	Simulación del crecimiento de la grieta . . . . .	95
5.4.1	Placa con grieta en la orilla . . . . .	99
5.4.2	Probeta apoyada en 4 puntos con grieta simple . . . . .	102
5.4.3	Probeta con dos apoyos y tres agujeros . . . . .	106

**6 Conclusiones**

**114**

**Bibliografía**

**118**

# Notación

Símbolo	Descripción
$\sigma_{ij}$	Tensor de esfuerzos , $(i, j) \in \{1, 2, 3\}$
$\epsilon_{ij}$	Tensor de deformaciones , $(i, j) \in \{1, 2, 3\}$
$r, \theta, z$	Coordenadas polares
$x, y, z$	Coordenadas cartesianas
$\sigma_{ij}$	Esfuerzos normales, $(i, j) \in \{x, y, z\}$ (cartesianas), $(i, j) \in \{r, \theta, z\}$ (polares)
$\tau_{ij}$	Esfuerzos de corte, $(i, j) \in \{x, y, z\}$ (cartesianas), $(i, j) \in \{r, \theta, z\}$ (polares)
$I, II, III$	Modos de carga
$K_I, K_{II}, K_{III}$	Factores de intensificación de esfuerzos
$U$	Vector desplazamiento
$u, v, w$	Componentes del vector desplazamiento
$E$	Módulo de Young, energía interna
$\nu$	Módulo de Poisson
$\mu$	Módulo de corte
$\Pi$	Energía potencial dada por la energía de deformación y fuerzas internas
$W_s$	Trabajo requerido para crear nueva superficie
$\gamma_s$	Energía de superficie
$\gamma_p$	Trabajo plástico por unidad de superficie creada
$\sigma_f$	Esfuerzo crítico para crear una grieta
$w_f$	energía de fractura total
$G$	Tasa de liberación de energía o fuerza de extensión de la grieta
$J$	Integral J
$w$	Densidad de energía de deformación
$T$	Vector de tracción
$n$	Vector normal a una superficie
$\Gamma$	Camino arbitrario para el cálculo de la integral J
$r_y$	Radio de la zona de la punta de la grieta donde se ha superado el límite plástico
$\sigma_y$	Esfuerzo de fluencia
$r_p$	Tamaño de la zona plástica
$a_e$	Largo equivalente de la grieta (corrección de la zona plástica de irwin)
$B$	Ancho de una probeta

Símbolo	Descripción
$K_{IC}$	Tenacidad a la fractura
$C, m$	Parámetros de la Ley de Paris
$N_i$	Funciones de forma
$u_i, v_i, w_i$	Componentes de desplazamiento $(u, v, w)$ en el nodo $i$
$\tilde{U}$	Vector desplazamiento aproximado por elementos finitos
$\tilde{\epsilon}$	Vector de deformaciones (2D) aproximado por elementos finitos
$\tilde{\sigma}$	Vector de esfuerzos (2D) aproximado por elementos finitos
$D$	Matriz de elasticidad
$L_i$	Coordenadas de área, $i \in \{1, 2, 3\}$
$e$	Error de aproximación por elementos finitos
$h$	Tamaño característico de un elemento finito
$p$	Orden del elemento finito
$t_i$	Triángulo $i$ (Elemento $i$ )
$q$	Indicador de calidad de un Elemento Finito
$P_{ij}$	Tensor Energía-Momentum de Eshelby
$a$	Largo de una grieta
$l$	largo característico de un elemento finito en la punta de una grieta

*“Las vacas no deben alimentarse con leche para producirla.”*

– Jorge Edwards



# Capítulo 1

## Introducción

### 1.1 Objetivos

A continuación se muestran los objetivos asumidos para este trabajo de tesis.

#### 1.1.1 Objetivos generales

A través del tema de tesis a desarrollar, se espera

- Implementar un método numérico que calcule los factores de intensificación de esfuerzos de una grieta en un sólido elástico isotrópico bidimensional, a partir de los resultados obtenidos con el método de elementos finitos, utilizando mallas de elementos triangulares en 2D y algoritmos de refinamiento de mallas basados en el LEPP (Longest Edge Propagation Path).
- Implementar una metodología que permita simular la propagación cuasi-estática o por fatiga de una grieta en un sólido elástico isotrópico en 2D.
- Implementar una aplicación computacional amigable que permita resolver problemas de mecánica de la fractura en 2D para geometrías arbitrarias.

#### 1.1.2 Objetivos específicos

Para llevar a cabo el objetivo general se consideran los siguientes objetivos específicos

- Implementar una aplicación computacional usando como base un software existente (Calderón & Rivara, 2004) que permita crear un modelo de elementos finitos utilizando mallas de triángulos, que pueda ser resuelto por un software de elementos finitos comercial.
- Implementar un método numérico para calcular el factor de intensificación de esfuerzos a partir de los resultados obtenidos del método de elementos finitos.
- Implementar computacionalmente una metodología que modifique localmente una malla de triángulos cerca de la punta de una grieta, de tal forma de iterativamente poder simular el crecimiento de una grieta, en 2D.

### 1.1.3 Aspectos Metodológicos

El trabajo presentado involucra las siguientes actividades:

- La modificación de un software ya existente, que permite crear y refinar mallas de triángulos (Calderón & Rivara, 2004). Dicho software se modificará para que pueda crear un modelo de elementos finitos, incluyendo propiedades del material (materiales elásticos isotrópicos), condiciones de desplazamiento y de carga, en un archivo de salida que pueda ser interpretado por el software comercial ABAQUS. El software permitirá además modelar grietas. Una grieta es modelada explícitamente como superficies libres. No se implementará un método que resuelva el modelo de elementos finitos.
- La implementación del método de extrapolación de desplazamientos, que permite calcular los factores de intensificación de esfuerzos.
- La implementación del método del máximo esfuerzo tangencial, que permite determinar el ángulo en el que crecerá la grieta, con respecto a su orientación original.
- La implementación de una metodología que modifique la malla, de tal forma de simular el crecimiento de la grieta. Para esto se utilizan los algoritmos Lepp-Delaunay, que ya están implementados en el software original.
- La realización de una serie de pruebas de cálculos de factores de intensificación de esfuerzos y propagación de grietas, para validar los resultados.

Todo lo anterior en conjunto permitirá cumplir el objetivo simular la propagación de una grieta.

## 1.2 Estudio Bibliográfico

Se incluye a continuación un estudio bibliográfico de trabajos con respecto al uso de los elementos finitos en la mecánica de la fractura. Esto implica el desarrollo de distintas metodologías para calcular los factores de intensificación de esfuerzos en 2D y 3D, el desarrollo en las técnicas de generación y refinamiento de mallas, y el desarrollo de métodos para modelar el crecimiento de una grieta.

### 1.2.1 Métodos numéricos en mecánica de la fractura

Chan *et al.* (1970) fueron los primeros en utilizar el método de elementos finitos para calcular los factores de intensificación de esfuerzos. Se observó que es difícil obtener la forma del campo cerca de la punta de la grieta utilizando elementos normales, pero que de todas maneras era factible, y se propuso el método de extrapolación de desplazamientos, que es usado hasta hoy.

Parks (1974) propone una metodología distinta, basada en la tasa de liberación de energía de deformación debido a la propagación de una grieta, que está ligada a la integral J (Rice, 1968). En este método, se realiza el análisis de un sistema con una grieta, y luego la grieta es extendida a una nueva posición moviendo los nodos,

repetiendo el análisis. El factor de intensificación de esfuerzos es obtenido al relacionarlo con la variación en la energía entre una configuración y otra. Este método es conocido como *Virtual Crack Extension*. Hellen (1975) probó este método más en detalle para 2D y 3D, en problemas lineales elásticos. de Lorenzi (1982) propone un método similar al propuesto por Parks, con aplicación en tres dimensiones. El mismo autor (de Lorenzi, 1985) continúa su trabajo, generalizando lo realizado e incluyendo efectos de tracciones en la superficie de la grieta y fuerzas externas.

Henshell & Shaw (1975) y Barsoum (1976) prueban que la forma  $1/\sqrt{r}$  que tienen el campo de esfuerzos y el de deformaciones cerca de la punta de la grieta, puede ser adquirida con el método de elementos finitos en forma muy precisa, si los puntos medios de los elementos que están en la punta de la grieta son desplazados  $1/4$  de su posición original (*Quarter point elements*), más cerca de la grieta. Este método permite calcular los factores de intensificación de esfuerzos con una malla menos fina que si se usaran elementos convencionales. Esto es válido para elementos isoparamétricos de 8 nodos (cuadriláteros) en 2D, y de 20 nodos (prismáticos) en 3D. Barsoum (1977) también utilizó este método para elementos triangulares en 2D.

Stern *et al.* (1976) introducen el método de la integral de la interacción de energía, para 2D, que se utiliza para estudiar modelos donde existen modos de carga combinados en la grieta. En este método, campos auxiliares se superponen a los campos que corresponden a la solución del problema. Con una elección adecuada de estos campos, pueden encontrarse ecuaciones que relacionan la superposición de los campos con los factores de intensificación de esfuerzos en forma separada.

Rybicki & Kanninen (1977) proponen un método de cálculo de los factores de intensificación de esfuerzos basados en el concepto de la integral de cierre de grieta (*crack closure integral*), para 2D. En este método, se relacionan los factores de intensificación de esfuerzos con la tasa de liberación de energía, la cual se calcula midiendo la energía necesaria para cerrar un segmento de la grieta. Este método utiliza los desplazamientos y fuerzas nodales obtenidas del método de elementos finitos, y no los esfuerzos y deformaciones, como en el método de Parks (1974). Shivakumar *et al.* (1988) generalizan este método a tres dimensiones.

Li *et al.* (1985) proponen una mejora al método de la integral J. Dado que la integral J es una integral de línea, ésta debe ser calculada numéricamente. El cálculo de esta integral presenta inconvenientes debido a la presencia de una singularidad en la punta de la grieta. Según Rice (1968), la integral J es independiente del camino, pero esto no se verifica numéricamente. Por esto, a partir del concepto de la función S, introducido por de Lorenzi (1982), que consiste en una función de mapeo entre dos configuraciones con diferentes largos de grieta; y utilizando el teorema de la divergencia, encuentra una integral de área equivalente (*Equivalent Domain Integral*) para calcular el valor de la tasa de liberación de energía de deformación. Este método cumple mejor la propiedad de la integral J de ser independiente del camino. Esto es válido para dos y tres dimensiones.

A partir de la idea de Li *et al.* (1985) de la integral equivalente, y el concepto de la función S de de Lorenzi, Nikishikov & Atluri (1987) desarrollan un método de integral de dominio equivalente que permite descomponer la integral J para calcular los factores de intensificación de esfuerzos en un estado de carga de modo mixto, es decir  $K_{II} \neq 0$ , para tres dimensiones.

Shih & Asaro (1988) utilizan el método de la interacción de Stern *et al.*, en conjunto con el método del

dominio equivalente de Li *et al.* para calcular los factores de intensificación de esfuerzos combinados en una grieta interfacial. Esta misma metodología puede ser generalizada para calcular una grieta en modo mixto en un material isotrópico homogéneo, lo que ocurre cuando una grieta tiene un ángulo de inclinación con respecto a la carga.

Raju & Shivakumar (1990) desarrollan un método de la integral de dominio equivalente, basándose en el trabajo de Nikishikov & Atluri, pero generalizando el concepto para calcular los factores de intensificación de esfuerzos en grietas presentes en la interfaz de dos materiales, así como en casos donde existen cargas en la superficie de la grieta. Shivakumar (1992) generaliza este concepto a 3D.

Bittencourt *et al.* (1996) desarrollan una metodología para simular el crecimiento cuasi-estático de una grieta, en 2D. Este método se basa en un remallado sólo de la zona cercana a la grieta, y no de todo el modelo. Para el refinamiento local, se utiliza un método basado en el concepto de “quadtree” para determinar el tamaño de cada elemento, cuya forma es posteriormente modificada (suavizada) reubicando los nodos. El tamaño de incremento de la grieta en cada paso de la propagación es controlado manualmente. Se prueba el método con resultados experimentales, y se observa que el tamaño del incremento de la grieta debe variar a lo largo de la simulación, y que este tamaño está ligado al radio  $K_I/K_{II}$ , asociado al radio de curvatura de la grieta.

Lin & Smith (1999) proponen una metodología para simular el crecimiento planar por fatiga de una grieta superficial en una placa, en 3D, que es uno de los problemas de fallas por fatiga más recurrentes. Este método considera múltiples grados de libertad para el crecimiento de una grieta superficial, lo que permite modelar grietas con un frente de forma arbitraria; en este trabajo se estudia el remallado del nuevo frente de grieta. Para crear un nuevo frente de grieta, la nueva posición se crea a partir del desplazamiento de algunos nodos de la grieta original, y luego reconstruyendo el frente de grieta con una curva spline cúbica. Se estudia la influencia de la forma del frente de la grieta, y de la malla en la punta de la grieta. Se utilizan los elementos isoparamétricos de 20 nodos propuestos por Barsoum (1976)

Carter *et al.* (2000) estudian la posibilidad de simular automáticamente el crecimiento de una grieta arbitraria en 3D, ya sea superficial o interna. Para esto, plantean un modelo para llevar a cabo este proceso, en particular, la forma en la que se debe modificar la malla geométrica para representar la grieta después del crecimiento. Se plantea que es posible simular el crecimiento sin necesidad de reconstruir la malla completamente, sino sólo modificando localmente el modelo.

Rajaram *et al.* (2000) utilizan el método de la integral de dominio equivalente de Li *et al.* en 3D, utilizando elementos finitos tetraedros de 10 nodos. En este trabajo se intenta calcular el factor de intensificación de esfuerzos de una grieta en modo I. En los elementos en el frente de la grieta, utilizan la metodología de desplazar los nodos a 1/4 de su posición. Concluyen que existen muchos inconvenientes para calcular los factores de intensificación de esfuerzos con tetraedros, dado que en algunos casos un vértice del elemento toca el frente de la grieta, y en otros casos una arista es la que toca. Esto produce oscilaciones inexistentes en la tasa de liberación de energía. Proponen una metodología para solucionar este problema, modificando las funciones de interpolación en los elementos cerca de la punta de la grieta. Hasta donde conoce el autor de esta tesis, es el único trabajo en el que se utilizan elementos tetraedros para calcular factores de intensificación de esfuerzos en 3D.

Guinea *et al.* (2000) estudia más en detalle el método de extrapolación de desplazamientos, en 2D. En particular, menciona el efecto del módulo de Poisson y el estado de esfuerzos en los resultados de los métodos energéticos, problemas que son evitados con el método de extrapolación de desplazamientos. Compara varias expresiones para obtener  $K_I$  con el método de extrapolación de desplazamientos, una de ellas que incluye términos del orden de  $l^{3/2}$ . Normalmente al usar el método de extrapolación de desplazamientos, se utilizan términos del orden de  $\sqrt{l}$ . En sus cálculos utilizan los elementos especiales introducidos por Barsoum (1976). Las expresiones utilizadas sólo son válidas para ese tipo de elementos. Se estudia la influencia de la forma de los elementos, el tipo de malla y el refinamiento.

Gosz & Moran (2002) desarrollan un método para calcular los factores de intensificación de esfuerzos en modos combinados en tres dimensiones, y con un frente de grieta no planar. Para resolver el modelo, utilizan los elementos finitos isoparamétricos de 20 nodos desarrollados por Barsoum (1976), y el método de la integral de dominio equivalente (Li *et al.*, 1985), en conjunto con el método de la interacción de Stern *et al.* (1976), igual que Shih & Asaro (1988), pero para 3D.

Miranda *et al.* (2003) calculan la vida a la fatiga y predicen el camino de la grieta en componentes estructurales 2D. Para esto, se utiliza la metodología desarrollada por Bittencourt *et al.* (1996) y se relaciona con criterios de diseño clásicos de la teoría de la fatiga.

Schöllmann *et al.* (2003) desarrollan un software para simular el crecimiento de una grieta. Se discretiza la geometría utilizando una malla de tetraedros, y en la zona del frente de grieta se superpone una pequeña malla de hexaedros. Con la malla de tetraedros se resuelve el método de elementos finitos, los cuales sirven de entrada para la malla de hexaedros que se utiliza para calcular los parámetros de la fractura.

Phongthanapanich & Dechaumphai (2004) utilizan el concepto de la triangulación Delaunay para simular la propagación de una grieta, en 2D. El método calcula los factores de intensificación de esfuerzos utilizando el método de extrapolación de desplazamientos, basado en el trabajo de Guinea *et al.* (2000). La dirección de propagación de esfuerzos se calcula utilizando la teoría del máximo esfuerzo circunferencial. Este método refina la malla haciendo inserciones Delaunay de puntos en el centroide de los triángulos en los cuales el error está sobre un valor máximo permitido (método adaptivo), por lo que en cada iteración el modelo es resuelto varias veces. Luego, se generan triángulos de buena calidad, reposicionando los nodos con una técnica de suavizamiento laplaciano. En cada iteración de la simulación del crecimiento, la malla es reconstruida completamente.

Okada *et al.* (2005b) modifican el método 3D de la integral de cierre de grieta de Shivakumar *et al.* (1988), relajando la restricción de que las caras de los elementos finitos en el frente de grieta tengan la misma área y que deban ser simétricos a lo largo del frente de la grieta, además de estar ordenados ortogonalmente. En este trabajo se utilizan elementos hexaedros de 20 nodos. Esta modificación es particularmente útil para usar este método en grietas con un frente que no es recto, como por ejemplo en una grieta elíptica superficial.

## 1.2.2 Generación de mallas de triángulos

Yerry & Shephard (1983) proponen un método de mallado con triángulos de geometrías arbitrarias basado en el concepto de “quadtree”<sup>1</sup>. Para esto, se discretiza la geometría con un “quadtree”, el cual se usa para

---

<sup>1</sup>Véase cap 3.2.2

determinar la posición de los nodos de la malla. Luego, la malla es suavizada reposicionando los nodos en el centroide del polígono formado por los triángulos que lo comparten. Este método genera mallas de buena calidad para geometrías arbitrarias.

Rivara (1984) presenta una técnica para refinar localmente una malla de triángulos ya existente. Este método se basa en la bisección de los lados más largos de los triángulos que se desea refinar. La ventaja de este método es que permite refinar iterativa y localmente la malla, manteniendo su calidad a través del proceso. Como se mencionó, este método permite refinar mallas ya existentes, por lo que es útil para métodos adaptivos o para modelos en los cuales la geometría debe ser modificada. El mismo autor generaliza este trabajo a 3D (Rivara & Levín, 1992)

Potyondy *et al.* (1995) desarrolla una metodología de mallado, basado en el trabajo de Yerry & Shephard (1983), con aplicación a la propagación de grietas. En este método, la zona cercana a la punta de la grieta es remallada, incluyendo en la punta elementos especiales (Barsoum, 1977), y en el resto de la zona elementos triangulares normales. A partir de esta metodología, se implementa el método de crecimiento de grieta de Bittencourt *et al.* (1996).

Rivara (1997) introduce el concepto de LEPP (longest edge propagation path) utilizado para el refinamiento de mallas de triángulos. El LEPP de un triángulo  $t$  se define como el conjunto de triángulos sucesivos unidos por su arista más larga. Basado en este concepto, se implementa un método de refinamiento en el cual sólo los triángulos incluidos en el LEPP son modificados, al intentar refinar  $t$ . A partir de este concepto se pueden implementar algoritmos de refinamiento que conservan o mejoran la calidad de la malla geométrica, lo cual es particularmente útil para métodos adaptivos y para la propagación de grietas, donde la geometría es modificada constantemente, además de necesitarse un refinamiento localizado. El mismo autor continúa el trabajo, generalizando el concepto del LEPP a 3-D (Rivara & Palma, 1997), para mallas de tetraedros.

Shewchuk (1998) presenta un algoritmo de refinamiento de mallas de triángulos y tetraedros. Éste consiste en insertar nodos en el centro de la circunferencia circunscrita a un triángulo de mala calidad (esfera circunscrita en el caso de un tetraedro). Esto permite refinar y mejorar la calidad de una malla. Este algoritmo se aplica en mallas Delaunay, y la inserción de puntos conserva la propiedad Delaunay.

Cavalcante-Neto *et al.* (2001) desarrollan una metodología de mallado de geometrías arbitrarias en 3 dimensiones, con grietas de formas arbitrarias en su interior. Este método utiliza la técnica del frente de avance (*advancing front*), combinado con una discretización de la geometría con un octree, para determinar el tamaño local de los elementos.

Zuo *et al.* (2005) desarrollan una metodología de mallado de geometrías con grietas arbitrarias en 3 dimensiones. La metodología desarrollada se basa en el método del frente de avance, el cual es mejorado para solucionar problemas inherentes a geometrías con grietas, como son los nodos y superficies que coinciden geoméricamente (dado que las grietas son geometrías 2D), así como la intersección de volúmenes en el proceso de remallado.

## Capítulo 2

# Mecánica de la Fractura

La mecánica de la fractura consiste, en términos generales, en estudiar en los mecanismos involucrados en la ruptura de un material. El principal motivo por el que un material falla es por la presencia de defectos en su interior o superficie, como poros o grietas. Por esto, la mecánica de la fractura estudia en particular lo que ocurre en las zonas colindantes a estos defectos, pues aquí se inicia el proceso de fractura.

La mecánica de la fractura tiene gran importancia en el diseño mecánico, y jugó un rol preponderante en los avances tecnológicos del siglo XX. Los avances en la mecánica de la fractura han permitido utilizar materiales más sofisticados, como aleaciones de alta resistencia mecánica y térmica o materiales compuestos, donde el comportamiento mecánico es distinto al esperado normalmente en un material como el acero de construcción. Estas aleaciones permitieron construir equipos térmicos que funcionan a mayor temperatura, o construir elementos mecánicos más pequeños y livianos, lo que permitió el desarrollo, por ejemplo, de la industria automotriz y aeroespacial (turbinas, motores, componentes mecánicos en alas de aviones, etc.).

La mecánica de la fractura actualmente sigue evolucionando, en particular por el desarrollo de la informática, que permite resolver numéricamente problemas complejos que hace 20 ó 30 años eran imposibles de resolver.

### 2.1 Desarrollo histórico

Los problemas surgidos de la fractura de estructuras y elementos mecánicos, así como el interés por solucionarlos ha existido desde el inicio de la humanidad. Sin embargo el estudio específico de este fenómeno comenzó mucho después. Los primeros intentos de estudiar este fenómeno provienen de Leonardo Da Vinci, quien estudió la resistencia de unos cables de acero en función de su largo. Observó que la resistencia de dichos cables disminuía con su largo. Esto está asociado a la presencia de defectos en el cable, pues mientras más largo es éste, la probabilidad de que exista un defecto es mayor. Sin embargo esta es una aproximación cualitativa. El estudio cuantitativo de la fractura no ocurrió hasta el siglo XX.

El primer trabajo cuantitativo que conecta la presencia de defectos con la fractura fue realizado por Griffith (1920). Su trabajo consistió en un análisis de esfuerzos de un agujero elíptico y lo relacionó con la propagación inestable de una grieta. A partir de un balance de energía, basado en la primera ley de la termodinámica, plantea una teoría de la fractura. Según esta teoría, un defecto se vuelve inestable (es decir,

ocurre la fractura) cuando el cambio de la energía de deformación debido al incremento en el largo de la grieta compensa el aumento de la energía superficial debido al mismo incremento. Esta teoría fue corroborada con experimentos en probetas de cristal. Sin embargo no pudo ser corroborada en metales, pues el supuesto de que la energía proviene de la energía de superficie sólo se cumple para materiales idealmente frágiles.

Después de la Segunda Guerra Mundial, en los laboratorios de investigación naval en Estados Unidos, Irwin (1948) extendió la teoría de Griffith a los metales, incluyendo el efecto de la deformación plástica local. Varios años antes, Westergaard (1939) desarrolló una técnica para analizar los esfuerzos y desplazamientos cerca de la punta de una grieta aguda. A partir de este trabajo, Irwin (1957) demostró que los campos de esfuerzos y desplazamientos pueden ser descritos por un parámetro característico, que está relacionado con la tasa de liberación de energía. Posteriormente este parámetro fue conocido como el factor de intensificación de esfuerzos. Estos trabajos dieron lugar a numerosas aplicaciones, como la prevención de fracturas en rotores, fuselajes de aviones, etc.

Paris *et al.* (1961) aplican los conceptos de la mecánica de la fractura al crecimiento de grietas por fatiga. Plantean que el número de ciclos de carga y el crecimiento de una grieta por fatiga están relacionados con los factores de intensificación de esfuerzos. Sin embargo pasó mucho tiempo antes que esta idea fuera aplicada en la práctica, dado que los ingenieros de la época veían con suspicacias esta teoría.

Los trabajos mencionados hasta ahora son válidos para materiales (en particular metales) que fallan por fractura frágil. Estos trabajos forman la base de lo que se conoce como la teoría de la mecánica de la fractura elástica lineal (LEFM, por sus siglas en inglés). Esta teoría deja de ser válida cuando existe una cantidad considerable de deformación plástica previa a la fractura. Wells (1961) da los primeros pasos hacia una teoría para la fractura plástica, al observar que en la fractura en materiales muy dúctiles, las caras de la grieta se separan. Esta observación dio paso a otro parámetro, llamado desplazamiento de apertura de la punta de la grieta (crack tip opening displacement, CTOD).

Rice (1968) propone otro parámetro para caracterizar el comportamiento no lineal cerca de la grieta, la llamada integral J, una integral de línea que está relacionada con la tasa de liberación de energía en materiales no lineales, la cual a la vez está relacionada con los campos de esfuerzos y desplazamientos cerca de la grieta. La integral J es usada actualmente para caracterizar la tenacidad a la fractura en aceros y otros materiales. La relación matemática entre los esfuerzos, la tenacidad a la fractura y el tamaño de los defectos fue introducida por Shih & Hutchinson (1976). A partir de estos trabajos se basa la teoría para estudiar el comportamiento no lineal en la fractura.

A partir de la década del '80, con el desarrollo de la computación, comienzan a desarrollarse las bases para el uso de métodos numéricos en la mecánica de la fractura. Una reseña detallada de esto ya fue mostrada en el estudio bibliográfico.

## **2.2 Mecánica de la Fractura Elástica Lineal**

La mecánica de la fractura elástica lineal (LEFM, por sus siglas en inglés) modela el comportamiento a la fractura dentro de los límites de la teoría de la elasticidad lineal, es decir materiales que no presentan comportamientos no lineales como la deformación plástica, o ésta está confinada a una zona muy pequeña,



en la punta de la grieta, de tal forma que no afecta el comportamiento global del material. Las bases de esta teoría corresponden a los trabajos realizados hasta 1960, según la reseña histórica anterior. A pesar de esto, la LEFM no es una teoría obsoleta. Por una parte, al estudiar esta teoría se logran comprender los conceptos fundamentales de la fractura, necesarios para avanzar hacia otros tópicos más sofisticados. Por otro lado, en los materiales donde la fractura es esencialmente frágil, es decir, existe poca o nula deformación plástica previo a la fractura, la LEFM permite modelar adecuadamente su comportamiento. Estos materiales aparecen con bastante frecuencia en la ingeniería, como es el caso de las aleaciones metálicas con alto límite de fluencia. En estos materiales, la falla se produce por la fractura rápida de una grieta que se vuelve inestable desde un punto de vista energético. Esta grieta puede ser un defecto proveniente del proceso de fabricación, o una grieta que se propagó por fatiga a partir de un defecto más pequeño. La propagación cuasi-estática de grietas y la propagación por fatiga pueden ser modeladas muy bien por los supuestos planteados en la LEFM.

### 2.2.1 Análisis de esfuerzos

Es posible encontrar expresiones analíticas para el campo de desplazamientos y los tensores de esfuerzos y deformaciones en cuerpos con grietas. Las expresiones mostradas en esta sección son válidas bajo los siguientes supuestos

- estado de esfuerzos en dos dimensiones (esfuerzo plano o deformación plana)
- material isotrópico
- deformación cuasiestática e isotérmica
- sin fuerzas externas

Las expresiones para casos más sofisticados (por ejemplo materiales anisotrópicos) están fuera del alcance de esta tesis, por lo tanto no se incluyen. Una descripción más detallada de la deducción de las expresiones mencionadas en este capítulo se puede encontrar en Anderson (1995, cap. 2.6 y Ap. 2) y Kanninen & Popelar (1985, cap. 3.1 y 3.2).

#### Expresión general para los esfuerzos cerca de la punta de la grieta

Williams (1957) fue uno de los primeros en publicar una expresión analítica general para un cuerpo con una grieta. Si se define un sistema de coordenadas polar con el origen centrado en la punta de la grieta (figura 2.1), los esfuerzos para cualquier cuerpo lineal elástico con una grieta están dados por

$$\sigma_{ij} = \left( \frac{k}{\sqrt{r}} \right) f_{ij}(\theta) + \sum_{m=0}^{\infty} A_m r^{\frac{m}{2}} g_{ij}^{(m)}(\theta) \quad (2.1)$$

donde  $\sigma_{ij}$  es el tensor de esfuerzos,  $r$  y  $\theta$  son las coordenadas polares,  $k$  es una constante, y  $f_{ij}$  es una función adimensional de  $\theta$ . Los términos de la sumatoria dependen de la geometría, pero en cualquier configuración geométrica, la solución siempre incluye un término proporcional a  $1/\sqrt{r}$ . Cuando  $r \rightarrow 0$ , los términos de la sumatoria tienden a 0, mientras que el término fuera de la sumatoria tiende a infinito. Es decir, en cualquier configuración geométrica, el tensor cerca de la punta de la grieta varía como  $1/\sqrt{r}$ .

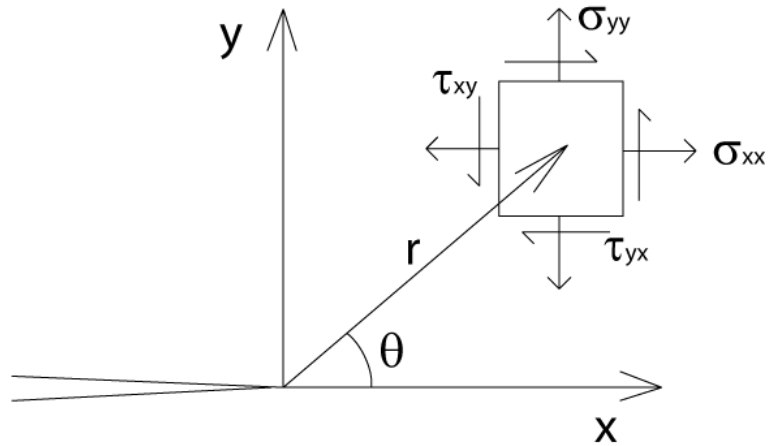


Figura 2.1: Sistema de coordenadas utilizado para deducir las expresiones de los campos de esfuerzos y desplazamientos. El origen está centrado en la punta de la grieta, y el eje x apunta en la dirección de la grieta.

### Factores de intensificación de esfuerzos

Una grieta puede experimentar 3 modos de carga (figura 2.2). En el modo I la carga principal es aplicada normal a la cara de la grieta, tendiendo a abrirla. En el modo II, la carga es de corte y tiende a deslizar una cara de la grieta con la otra. En el modo III, el esfuerzo de corte tiende a deslizar las caras en forma perpendicular al plano x-y.

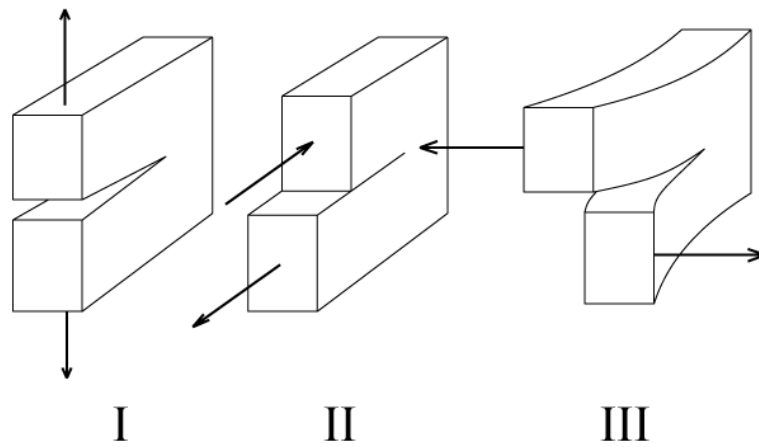


Figura 2.2: Modos de carga a los que puede estar sujeto una grieta: El modo I tiende a abrir la grieta, El modo II tiende a deslizar una cara de la grieta con la otra, y el modo III desliza las caras de la grieta en forma perpendicular al plano x-y.

El estado general de esfuerzos es una combinación de estos modos. La contribución individual de cada modo al campo de esfuerzos es aditivo, por lo tanto se puede escribir

$$\sigma_{ij}^{total} = \sigma_{ij}^I + \sigma_{ij}^{II} + \sigma_{ij}^{III} \quad (2.2)$$

Se define el *factor de intensificación de esfuerzos* como  $K = k\sqrt{2\pi}$ . El factor de intensificación de esfuerzos se denota con un sub-índice para indicar el modo de carga,  $K_I$ ,  $K_{II}$  y  $K_{III}$ . Cerca de la punta de la grieta, los términos de la sumatoria de la ecuación 2.1 son nulos, y sólo el término singular sobrevive. Por lo tanto, el campo puede escribirse como

$$\lim_{r \rightarrow 0} \sigma_{ij}^I = \frac{K_I}{\sqrt{2\pi r}} f_{ij}^I(\theta) \quad (2.3)$$

$$\lim_{r \rightarrow 0} \sigma_{ij}^{II} = \frac{K_{II}}{\sqrt{2\pi r}} f_{ij}^{II}(\theta) \quad (2.4)$$

$$\lim_{r \rightarrow 0} \sigma_{ij}^{III} = \frac{K_{III}}{\sqrt{2\pi r}} f_{ij}^{III}(\theta) \quad (2.5)$$

Para el caso plano en dos dimensiones, se pueden deducir expresiones analíticas para los campos de esfuerzos y deformaciones cerca de la punta de la grieta (Kanninen & Popelar, 1985, cap. 3.1.2). Los esfuerzos para el modo *I* están dados por

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix} = \frac{K_I}{\sqrt{2\pi r}} \cos(\theta/2) \begin{Bmatrix} 1 - \text{sen}(\theta/2)\text{sen}(3\theta/2) \\ \text{sen}(\theta/2)\text{cos}(3\theta/2) \\ 1 + \text{sen}(\theta/2)\text{sen}(3\theta/2) \end{Bmatrix} \quad (2.6)$$

y los desplazamientos

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \frac{K_I}{2\mu} \sqrt{\frac{r}{2\pi}} \begin{Bmatrix} \text{cos}(\theta/2)[\kappa - 1 + 2\text{sen}^2(\theta/2)] \\ \text{sen}(\theta/2)[\kappa + 1 - 2\text{cos}^2(\theta/2)] \end{Bmatrix} \quad (2.7)$$

El factor de intensificación de esfuerzos modo *I*,  $K_I$  se define como

$$K_I = \lim_{r \rightarrow 0} \{ \sqrt{2\pi r} \sigma_{yy} |_{\theta=0} \} \quad (2.8)$$

Para el modo *II*

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix} = \frac{K_{II}}{\sqrt{2\pi r}} \begin{Bmatrix} -\text{sen}(\theta/2)[2 + \text{cos}(\theta/2)\text{cos}(3\theta/2)] \\ \text{cos}(\theta/2)[1 - \text{sen}(\theta/2)\text{sen}(3\theta/2)] \\ \text{sen}(\theta/2)\text{cos}(\theta/2)\text{cos}(3\theta/2) \end{Bmatrix} \quad (2.9)$$

y los desplazamientos

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \frac{K_{II}}{2\mu} \sqrt{\frac{r}{2\pi}} \begin{Bmatrix} \text{sen}(\theta/2)[\kappa + 1 + 2\text{cos}^2(\theta/2)] \\ -\text{cos}(\theta/2)[\kappa - 1 - 2\text{sen}^2(\theta/2)] \end{Bmatrix} \quad (2.10)$$

El factor de intensificación de esfuerzos modo II,  $K_{II}$  se define como

$$K_{II} = \lim_{r \rightarrow 0} \{ \sqrt{2\pi r} \sigma_{xy} |_{\theta=0} \} \quad (2.11)$$

Los factores de intensificación de esfuerzos definen la amplitud de la singularidad, es decir los esfuerzos crecen en forma proporcional a  $K$ , en la zona cerca de la grieta. Se puede ver en las ecuaciones anteriores que si se conoce  $K$ , los esfuerzos y deformaciones en esta zona quedan completamente determinados. De ahí la importancia de este parámetro en la mecánica de la fractura.

Es posible relacionar los factores de intensificación de esfuerzos con las energías involucradas en la fractura. Esta relación es fundamental para el desarrollo de los métodos numéricos para abordar problemas de ingeniería.

### 2.2.2 Análisis energético

A continuación se muestra una aproximación a una teoría de la mecánica de la fractura desde un enfoque energético, es decir de las fuerzas y potenciales involucrados en la creación y propagación de una grieta. Primero se muestra el trabajo realizado por Griffith (1920) para materiales idealmente frágiles. Luego se muestran las modificaciones hechas por Irwin (1948) para generalizar los conceptos de Griffith a metales, así como la introducción del concepto de la tasa de liberación de energía, de gran relevancia en el posterior desarrollo de los métodos numéricos. Para una descripción más detallada de estos conceptos, se recomienda leer a Anderson (1995, cap. 2.3 y 2.4) y Kanninen & Popelar (1985, cap. 3.3.1)

#### El modelo de Griffith

Según la primera ley de la termodinámica, cuando un sistema pasa de un estado fuera del equilibrio a uno de equilibrio, la energía debe disminuir. A partir de este concepto, Griffith (1920) postula un modelo energético para explicar la formación de una grieta. Por lo tanto, una grieta puede formarse, o crecer si es que ya existe, sólo si este proceso causa que la energía total del sistema disminuya o permanezca constante.

En la figura 2.3 se observa una grieta de largo  $2a$ , en una placa infinita de ancho  $B$ , en condiciones de esfuerzo plano. Si existe un incremento  $dA$  en el área de la grieta ( $A = 2aB$ ), el balance de energía para este incremento, bajo condiciones de equilibrio es

$$\frac{dE}{dA} = \frac{d\Pi}{dA} + \frac{dW_s}{dA} = 0 \quad (2.12)$$

donde  $E$  es la energía total,  $\Pi$  es la energía potencial dada por la energía de deformación interna y las fuerzas externas, y  $W_s$  es el trabajo requerido para crear nuevas superficies. Se puede mostrar (Anderson, 1995, cap. 2) que

$$\Pi = \Pi_0 - \frac{\pi\sigma^2 a^2 B}{E} \quad (2.13)$$

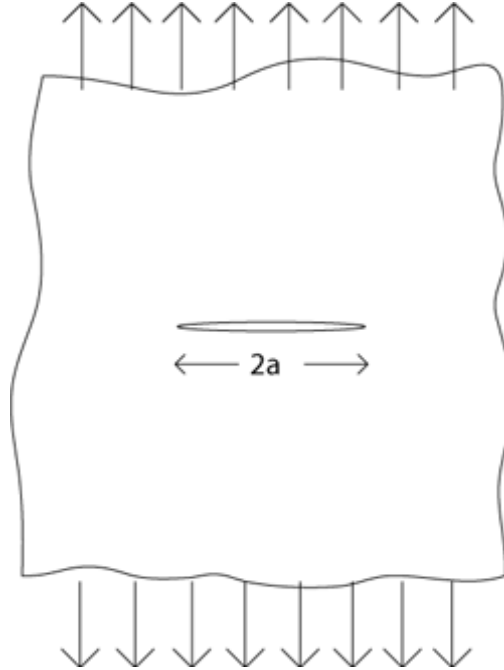


Figura 2.3: Grieta de ancho  $2a$  en una placa infinita, sujeta a esfuerzos remotos.

donde  $B$  es el espesor de la placa, y  $E$  es el Módulo de Young del material. Como para formar una grieta se deben formar dos superficies, la expresión para  $W_s$  es

$$W_s = 4aB\gamma_s \quad (2.14)$$

donde  $\gamma_s$  es la energía de superficie del material. Por lo tanto

$$-\frac{d\Pi}{dA} = \frac{\pi\sigma^2 a}{E} \quad (2.15)$$

$$\frac{dW_s}{dA} = 2\gamma_s \quad (2.16)$$

Igualando 2.15 y 2.16, el valor del esfuerzo crítico para que se cree una grieta es

$$\sigma_f = \left( \frac{2E\gamma_s}{\pi a} \right)^{1/2} \quad (2.17)$$

Generalizando para 3D, para una grieta de superficie circular se obtiene la siguiente expresión (Anderson, 1995, cap. 2)

$$\sigma_f = \left( \frac{\pi E \gamma_s}{2(1-\nu^2)a} \right)^{1/2} \quad (2.18)$$

donde  $a$  es el radio del círculo y  $\nu$  es el Coeficiente de Poisson.

### Modificación al modelo de Griffith

El modelo planteado por Griffith sólo es válido para materiales idealmente frágiles. Irwin (1948) modificó este modelo para incluir el efecto de la deformación plástica. La expresión para el esfuerzo crítico, incluyendo el efecto de la deformación plástica es

$$\sigma_f = \left( \frac{2E(\gamma_s + \gamma_p)}{\pi a} \right)^{1/2} \quad (2.19)$$

donde  $\gamma_p$  es el trabajo plástico por unidad de área de superficie creada.

$\gamma_s$  refleja la energía asociada a la ruptura de enlaces atómicos para formar una superficie. Sin embargo, cuando una grieta se propaga en un metal, no sólo se rompen los enlaces, sino que además existe movimiento de dislocaciones.  $\gamma_p$  da cuenta de esta energía extra, debido al movimiento de dislocaciones. Se puede generalizar la expresión de Griffith para cualquier tipo de de disipación de energía

$$\sigma_f = \left( \frac{2Ew_f}{\pi a} \right)^{1/2} \quad (2.20)$$

donde  $w_f$  es la energía de fractura, ya sea elástica, plástica, viscoelástica o viscoplástica. Esta expresión es válida para materiales donde el comportamiento global sea elástico, y el efecto plástico esté confinado a la punta de la grieta. En caso de no ser así, los supuestos para llegar a esta expresión no son válidos, además de que el valor de  $w_f$  no necesariamente es constante.

### La tasa de liberación de energía

Irwin (1956) introduce el concepto de la tasa de liberación de energía, concepto que es más conveniente para resolver problemas de ingeniería. La tasa de liberación de energía,  $G$ , se define como

$$G = -\frac{d\Pi}{dA} \quad (2.21)$$

Dado que  $G$  es la derivada de un potencial, también se llama fuerza de extensión de la grieta. Según 2.15, la tasa de liberación de energía para la placa infinita con una grieta de largo  $2a$  es

$$G = \frac{\pi\sigma^2 a}{E} \quad (2.22)$$

La extensión de la grieta ocurre cuando  $G$  alcanza un valor crítico

$$G_c = \frac{dW_s}{dA} = 2w_f \quad (2.23)$$

donde  $G_c$  es una medida de la tenacidad a la fractura del material. La energía potencial de un sólido elástico es

$$\Pi = U - F \quad (2.24)$$

donde  $U$  es la energía de deformación almacenada en el cuerpo, y  $F$  es el trabajo debido a fuerzas externas.

### Relación entre $K$ y $G$ , integral de cierre de grieta

Se puede demostrar (Kanninen & Popelar, 1985, cap. 3.3.1) que la tasa de liberación de energía y los factores de intensificación de esfuerzos están relacionados por la expresión

$$G = \frac{K_I^2}{E'} + \frac{K_{II}^2}{E'} + \frac{K_{III}^2}{2\mu} \quad (2.25)$$

donde

$$\mu = \frac{E}{2(1+\nu)} \quad (2.26)$$

es el módulo de corte,

$$E' = E \quad (2.27)$$

para esfuerzo plano, y

$$E' = \frac{E}{1-\nu^2} \quad (2.28)$$

para deformación plana.

Irwin (1957) demostró esta relación, introduciendo el concepto de la integral del cierre de grieta (*crack-closure integral*), de gran relevancia para los métodos numéricos para calcular los parámetros de fractura. Considérese una grieta de largo  $a + \Delta a$  sujeta a una carga en modo I, como muestra la figura 2.4(a). Si se aplica un esfuerzo compresivo entre  $x = 0$  y  $x = \Delta a$  lo suficientemente grande para cerrar la grieta (figura 2.4(b)), el trabajo asociado a ese cierre puede relacionarse con la tasa de liberación de energía

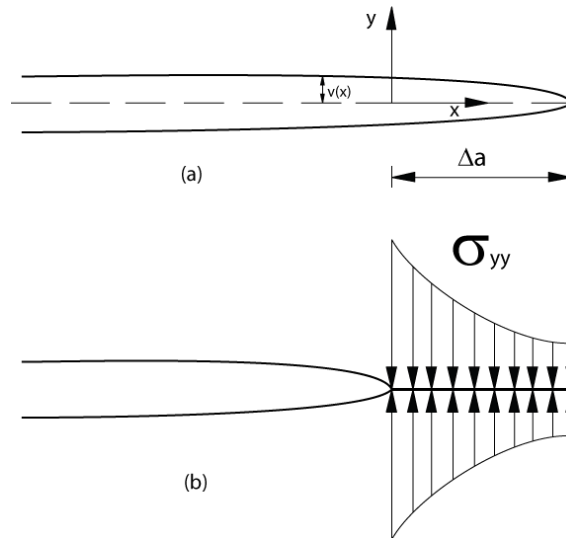


Figura 2.4: Sobre una grieta de largo  $a + \Delta a$  (a) se aplica una distribución de esfuerzos para cerrar una parte de ella, de largo  $\Delta a$ . (b). El trabajo necesario para cerrar la grieta puede relacionarse con la tasa de liberación de energía.

$$G = \lim_{\Delta a \rightarrow 0} \left( \frac{\Delta U}{\Delta a} \right)_{\text{carga fija}} \quad (2.29)$$

donde  $U$  es el trabajo para cerrar la grieta, que está dado por

$$\Delta U = \int_{x=0}^{x=\Delta a} dU(x) \quad (2.30)$$

El trabajo es el área bajo la curva fuerza-desplazamiento

$$dU(x) = 2 \frac{1}{2} F_y(x) v(x) = \sigma_{yy} v(x) dx \quad (2.31)$$

Reemplazando las expresiones de 2.6 y 2.7 para  $\theta = \pi$ , se obtiene que

$$G = \lim_{\Delta a \rightarrow 0} \frac{(\kappa + 1) K_I(a) K_I(a + \Delta a)}{4\pi\mu\Delta a} \int_0^{\Delta a} \sqrt{\frac{\Delta a - x}{x}} dx \quad (2.32)$$

$$= \frac{(\kappa + 1) K_I^2}{8\mu} = \frac{K_I^2}{E'} \quad (2.33)$$

donde  $K_I(a)$  y  $K_I(a + \Delta a)$  corresponden a los factores de intensificación de esfuerzos para la grieta de largo  $a$  y  $a + \Delta a$  respectivamente.

Se puede realizar un análisis similar para los modos *II* y *III*, eligiendo los esfuerzos y desplazamientos relevantes para cada caso.

Dado que la energía es un escalar, pueden sumarse las contribuciones de los tres modos, como muestra la ecuación 2.25.

## La integral J

Rice (1968) introduce una integral de línea independiente del camino, de mucha importancia en la mecánica de la fractura. Esta expresión es válida para la fractura elástica lineal y la fractura elasto-plástica. Dado un camino  $\Gamma$  arbitrario (en sentido contrario a las manecillas del reloj) alrededor de la punta de una grieta, empezando en su superficie inferior y terminando en su superficie superior, como muestra la figura 2.5, la integral J está dada por la expresión

$$J = \int_{\Gamma} \left( w dy - T_i \frac{\partial u_i}{\partial x} ds \right) \quad (2.34)$$

donde  $w$  es la densidad de energía de deformación,  $T_i$  son las componentes del vector de tracción,  $u_i$  son las componentes del vector de desplazamiento y  $ds$  es el diferencial de línea. La densidad de energía de deformación está dada por

$$w = \int_0^{\epsilon_{ij}} \sigma_{ij} d\epsilon_{ij} \quad (2.35)$$



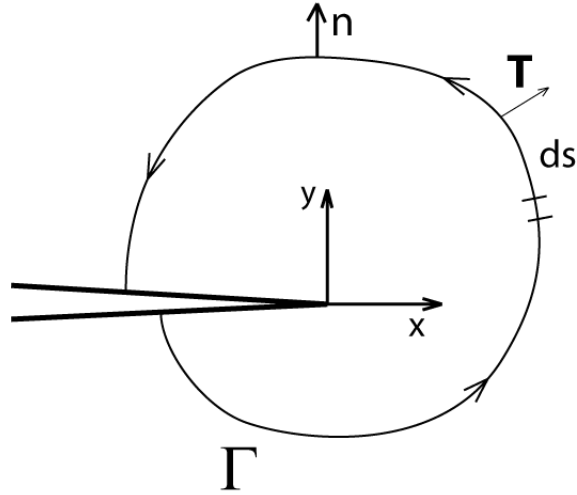


Figura 2.5: Contorno arbitrario  $\Gamma$  alrededor de la punta de una grieta, utilizado para la deducción de las expresiones asociadas a la *integral J*.

La tracción es un vector de esfuerzos normal al contorno. Las componentes están dadas por

$$T_i = \sigma_{ij}n_j \quad (2.36)$$

donde  $n_j$  son las componentes del vector unitario normal a  $\Gamma$ .

La deducción de la expresión para la integral J, así como la demostración de su independencia del camino, se encuentran en Kanninen & Popelar (1985, 3.3.2).

La integral J es igual a la tasa de decrecimiento de la energía potencial, es decir

$$J = -\frac{d\Pi}{da} \quad (2.37)$$

Por lo tanto, para un material elástico lineal, la integral J es equivalente a la tasa de liberación de energía, G, y por lo tanto, a los factores de intensificación de esfuerzos

$$J = G = \frac{1}{E'}(K_I^2 + K_{II}^2) + \frac{1}{2\mu}K_{III}^2 \quad (2.38)$$

### Plasticidad en la punta de la grieta

Como se mencionó, la teoría de la LEFM considera que no existe deformación plástica previa a la fractura. Esto sólo ocurre en materiales idealmente frágiles. En los materiales recurrentes en la ingeniería, siempre existe deformación plástica previo a la fractura. Sin embargo, es posible utilizar los conceptos de la LEFM a materiales que no son idealmente frágiles, bajo el supuesto que la plasticidad está limitada a una zona muy

pequeña, en la punta de la grieta. En estricto rigor, la zona plástica debe ser más pequeña que la zona dominada por el parámetro  $K$ , es decir la zona cerca de la punta de la grieta donde los esfuerzos son proporcionales a  $1/\sqrt{r}$ . El tamaño de esta zona puede estimarse, mediante la llamada aproximación de Irwin (1960). La ecuación 2.6, para  $\theta = 0$ , se reduce a

$$\sigma_{yy} = \frac{K_I}{\sqrt{2\pi r}} \quad (2.39)$$

Reemplazando en la ecuación 2.39 el límite de fluencia,  $\sigma_y$ , se obtiene la zona de la punta de la grieta en la cual se ha superado el límite elástico

$$r_y = \frac{1}{2\pi} \left( \frac{K_I}{\sigma_y} \right)^2 \quad (2.40)$$

En un material real, en esta zona se producirá deformación plástica, y los esfuerzos se redistribuirán como se muestra en la figura 2.6.

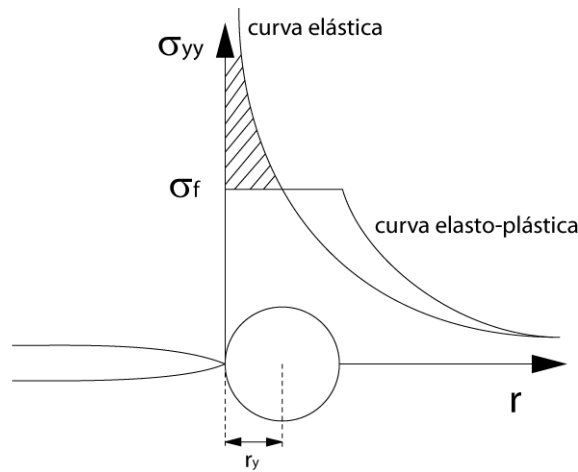


Figura 2.6: Distribución de los esfuerzos en el caso perfectamente elástico, y en el caso con deformación plástica en la punta de la grieta.

Dado que para  $0 \leq r \leq r_y$  se alcanzó el límite de fluencia, una aproximación del comportamiento consiste en suponer que el esfuerzo está uniformemente distribuido en esta zona. La energía que no se incluye bajo este supuesto, es decir, el área achurada en la figura 2.6, se reubica a la derecha, por lo que la zona plástica es mayor que  $r_y$ . El valor de la zona plástica es, para esfuerzo plano

$$r_p = 2r_y = \frac{1}{\pi} \left( \frac{K_I}{\sigma_y} \right)^2 \quad (2.41)$$

En forma equivalente, para deformación plana, la zona plástica está dada por

$$r_p = \frac{1}{3\pi} \left( \frac{K_I}{\sigma_y} \right)^2 \quad (2.42)$$

Un material que presenta deformación plástica de este tipo es ligeramente más tenaz que uno idealmente frágil. Irwin (1960) propone que se puede modelar el efecto en la tenacidad de la zona plástica utilizando la teoría de la LEFM, suponiendo un largo equivalente de la grieta,  $a_e$ , definido como

$$a_e = a + r_y \quad (2.43)$$

que se conoce como la *corrección de la zona plástica de Irwin*. A partir de este largo efectivo, se pueden calcular los factores de intensificación de esfuerzos a partir de la teoría de la LEFM y obtener resultados acordes con la realidad. Dado que el factor de intensificación de esfuerzos depende de  $a_e$ , la solución se busca en forma iterativa.

A partir de un análisis de esfuerzos basado en el criterio de Von Mises (Anderson, 1995, cap. 2.8.4), (Kanninen & Popelar, 1985, cap. 3.4), se puede obtener la forma de la zona plástica, para estados de esfuerzo plano y deformación plana. La figura 2.7 muestra cualitativamente la forma que toma la zona plástica para los modos I, II y III respectivamente, obtenidas a partir de un análisis elástico. Es necesario realizar un análisis elasto-plástico para obtener la forma más exacta. Dicho análisis está fuera del alcance de esta tesis.

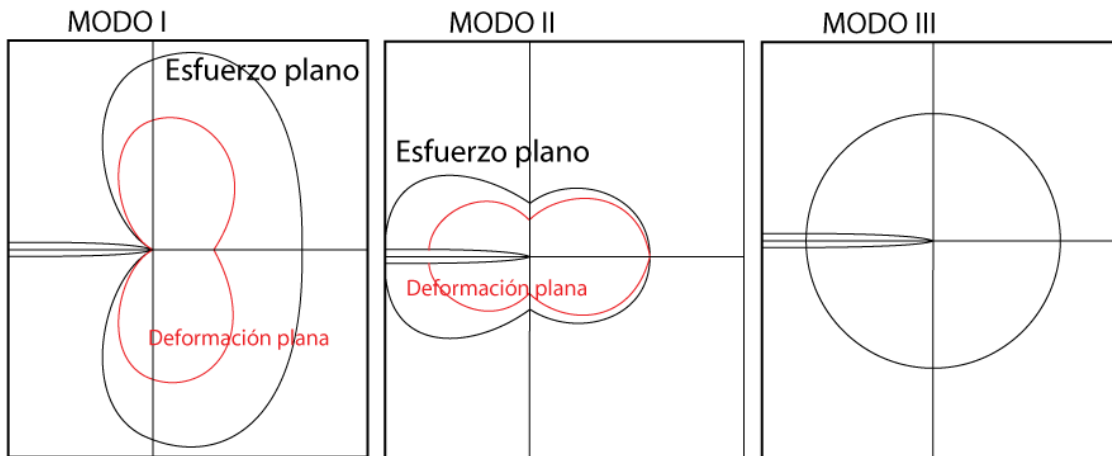


Figura 2.7: Forma de la zona plástica cerca de la punta de la grieta obtenida a partir de un análisis elástico, para cada uno de los modos de carga.

### Esfuerzo plano y deformación plana

En un estado general de esfuerzos en tres dimensiones, se tiene un estado intermedio entre esfuerzo plano y deformación plana. Dado un espécimen de ancho  $B$ , y con una grieta que llega hasta ambas superficies del material, la forma de la zona plástica es cualitativamente como muestra la figura 2.8. En la zona más central de la grieta, la deformación según  $z$  está restringida debido a que está rodeado de material, por lo

tanto, en esta zona el estado de esfuerzos predominante es de deformación plana. Cerca de la superficie, dado que son superficies libres, no existen esfuerzos transversales que eviten la deformación, por lo que el estado predominante es de esfuerzo plano. La figura 2.9 muestra cómo varían los esfuerzos y deformaciones a lo largo de la dirección transversal. Se puede inferir de dicho gráfico que mientras más grande sea  $B$ , la zona de deformación plana es mayor con respecto a la zona de esfuerzo plano. En las grietas internas, al no existir paredes libres en la dirección transversal, el estado predominante es el de deformación plana. Este supuesto es relevante a la hora de plantear métodos numéricos para calcular factores de intensificación de esfuerzos en 3D.

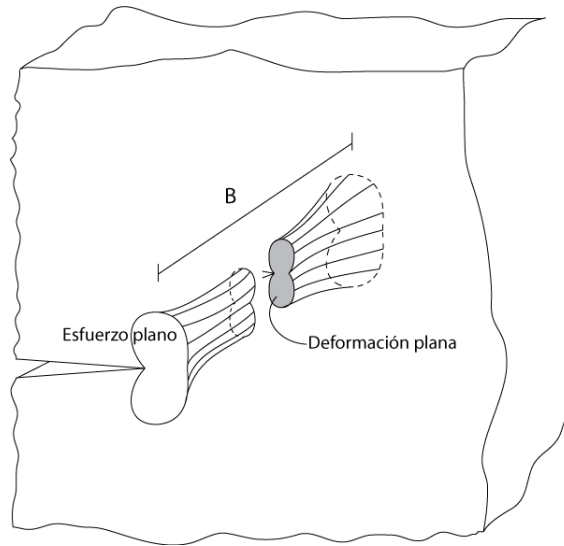


Figura 2.8: Forma de la zona plástica en un sólido en tres dimensiones. En las zonas cercanas a la superficie, el estado predominante es el de esfuerzo plano, mientras que en el interior, es de deformación plana.

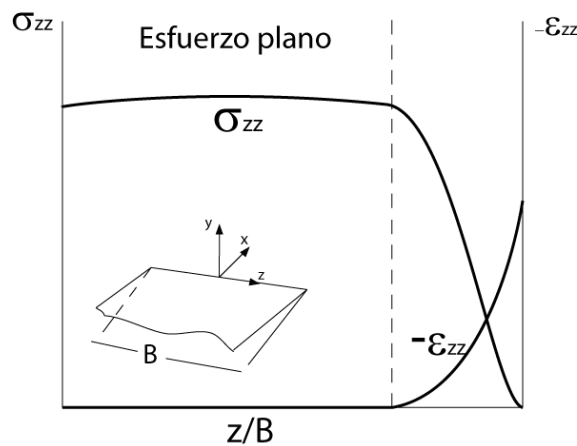


Figura 2.9: Variación de los esfuerzos y deformaciones transversales a lo largo del espesor.

## Tenacidad a la fractura

Para medir experimentalmente la tenacidad a la fractura, normalmente se utiliza una probeta llamada probeta compacta (*compact tension specimen*), cuya geometría se muestra en la figura 2.10. En dicha probeta, se hace propagar una grieta por fatiga desde la punta, y sobre ella se mide la tenacidad a la fractura. Previo a la fractura, existe deformación plástica. El tamaño de esa zona no se conoce antes de realizar el ensayo, y éste depende de las dimensiones de la probeta. Si el tamaño de la zona plástica es del orden de magnitud del tamaño de la probeta, entonces el estado de esfuerzos es de esfuerzo plano, por lo que la tenacidad medida depende de estas dimensiones, en particular del espesor de la probeta,  $B$ . La tenacidad de un material está asociada directamente a la cantidad de deformación plástica previa a la fractura. Como se observa en la figura 2.7, para un estado de esfuerzos de deformación plana, se tiene una zona de deformación plástica menor que en el estado de esfuerzo plano, por lo que la tenacidad es menor. A medida que se aumenta el espesor de la probeta, el estado de esfuerzos se vuelve más cercano a la condición de deformación plana, por lo que la tenacidad disminuye. Esto ocurre hasta que el estado de esfuerzos se hace predominantemente de deformación plana. Cuando ocurre esto, un aumento en  $B$  no tiene mayor efecto en la tenacidad a la fractura medida. Al alcanzar la zona de deformación plana, el valor crítico del factor de intensificación de esfuerzos medido se conoce como *tenacidad a la fractura* ( $K_{IC}$ ), parámetro que es independiente de las dimensiones. Los valores de la tenacidad a la fractura tabulados en los distintos libros de la especialidad corresponden a valores medidos en estado de esfuerzo plano.

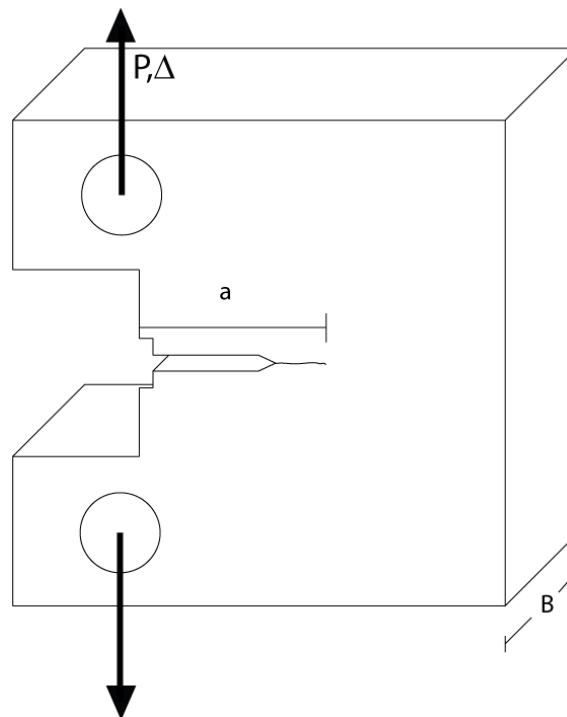


Figura 2.10: Probeta compacta (*compact tension specimen*), usada para medir experimentalmente la tenacidad a la fractura. La muesca se fabrica mediante mecanizado, y a partir de ella se hace propagar una grieta por fatiga, para poder realizar el ensayo.

## 2.3 Propagación de grietas

A partir de lo estipulado en la teoría de la LEFM, se puede modelar el crecimiento de una grieta para casos de crecimiento cuasi-estático y de propagación por fatiga. El caso de la propagación de grietas por fatiga es particularmente interesante, dado que es una de las principales causas de falla en materiales de alta resistencia.

### 2.3.1 Ley de Paris

Si un material con una grieta en su interior es sometido a una carga cíclica, sufrirá la propagación de dicha grieta hasta que ésta alcance un tamaño crítico y el material falle por fractura frágil. Este proceso se conoce como falla por fatiga. Una de las primeras observaciones importantes a la hora de relacionar la fatiga con la teoría de la mecánica de la fractura, es que dicho proceso es controlado no por la variación en los esfuerzos, sino por la variación en los factores de intensificación de esfuerzos  $\Delta K$  presentes en la grieta en cada ciclo. Paris *et al.* (1961) Planteó una expresión analítica que relaciona el crecimiento de la grieta, la cantidad de ciclos involucrados y los factores de intensificación de esfuerzos

$$\frac{da}{dN} = C\Delta K^m \quad (2.44)$$

Esta expresión es conocida como la *ley de Paris*. Se observa de dicha expresión que el proceso de crecimiento es controlado por la variación de los factores de intensificación de esfuerzos. Los parámetros  $C$  y  $m$  dependen del material a modelar, y se obtienen experimentalmente ajustando las mediciones a una curva potencial. En la figura 2.11 se observa un gráfico típico donde se ajusta la medición experimental de  $\frac{da}{dN}$  v/s  $\Delta K$ , para un acero SAE 1020.

### 2.3.2 Cambio de dirección de la grieta

Cuando una grieta preexistente se propaga en modo mixto (es decir  $K_{II} \neq 0$  y/o  $K_{III} \neq 0$ , la grieta cambia de dirección durante el proceso de propagación, dependiendo de la distribución de esfuerzos local. Existen varios criterios para analizar cuál será el cambio de ángulo de una grieta frente a un estado de carga mixto, que se muestran a continuación.

#### Criterio del máximo esfuerzo tangencial

En coordenadas polares, el campo de esfuerzos cerca de la punta de la grieta está dado por

$$\sigma_{\theta\theta} = \frac{1}{\sqrt{2\pi r}} \cos(1/2\theta) \left( K_I \cos^2(1/2\theta) - \frac{3}{2} K_{II} \sin(\theta) \right) \quad (2.45)$$

$$\tau_{r\theta} = \frac{1}{\sqrt{2\pi r}} \cos(1/2\theta) [K_I \sin(\theta) + K_{II}(3\cos(\theta) - 1)] \quad (2.46)$$

La dirección de propagación puede obtenerse imponiendo la condición  $\partial\sigma_{\theta\theta}/\partial\theta = 0$  o  $\tau_{r\theta} = 0$  (Erdogan & Sih, 1963), con lo que se obtiene la siguiente ecuación

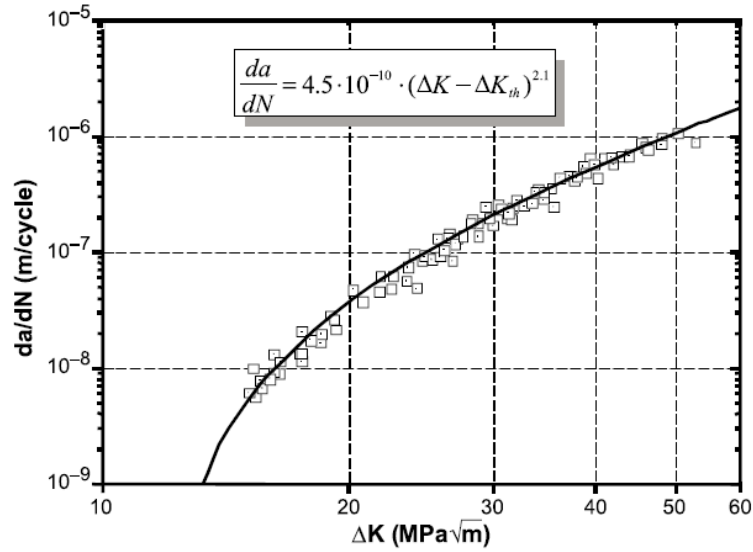


Figura 2.11: Mediciones experimentales de  $\frac{da}{dN}$  v/s  $\Delta K$ , y ajuste de los datos a una curva potencial para obtener los parámetros C y m de la ley de Paris. En la figura se muestra el caso de un acero SAE 1020 (obtenido de Castro *et al.* (2003)).

$$K_I \sin \theta + K_{II} (3 \cos \theta - 1) = 0 \quad (2.47)$$

con  $\theta$  medido con respecto al plano de la grieta. Esta ecuación puede ser resuelta numéricamente, por ejemplo con el método de Newton-Raphson.

### Criterio de la máxima tasa de liberación de energía

Dado un segmento de grieta de tamaño  $a$  que aparece en una grieta anterior como muestra la figura 2.12, con un ángulo  $\theta$  con respecto al plano de ésta, los factores de intensificación de esfuerzos de la nueva grieta dependen de los factores de intensificación de esfuerzos de la grieta anterior

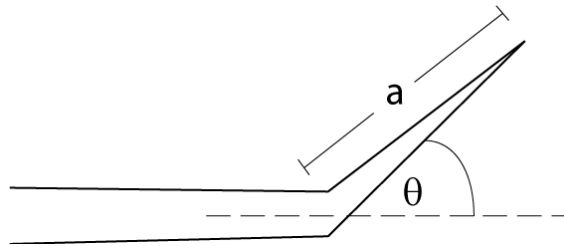


Figura 2.12: Crecimiento de un segmento de grieta de largo  $a$  a partir de otra grieta más grande.

$$\begin{aligned} K_I^k &= c_{11} K_I + c_{12} K_{II} \\ K_{II}^k &= c_{21} K_I + c_{22} K_{II} \end{aligned} \quad (2.48)$$

Los coeficientes  $c_{ii}$  están dados por Hayashi & Nemat-Nasser (1981). El criterio de la máxima tasa de liberación de energía plantea que el ángulo  $\theta^*$  es tal que maximiza la tasa de liberación de energía para la nueva configuración de la grieta, es decir

$$G^k = \frac{1}{E'}(K_I^{k2} + K_{II}^{k2}) \quad (2.49)$$

Estos dos criterios son los utilizados normalmente en metodologías numéricas. En esta tesis se utiliza el primero de ellos.

En este capítulo se ha revisado el desarrollo histórico de la mecánica de la fractura, una descripción más amplia sobre la mecánica de la fractura lineal, y finalmente la propagación de grietas. Los aspectos recién enumerados abarcan los conocimientos esenciales para poder abordar y desarrollar esta tesis, en lo que respecta al planteamiento físico del problema. A continuación se muestran los aspectos relacionados con la metodología numérica.



## Capítulo 3

### Metodología numérica

En los últimos 20 años, el desarrollo de la informática ha permitido resolver problemas de la ingeniería que antes eran difíciles o casi imposibles, lo que trajo un importante impacto en el diseño. En particular la mecánica de sólidos, y la mecánica de la fractura, se vieron beneficiados de estos avances. Para problemas sofisticados de diseño, como geometrías o estados de carga complejos, las soluciones analíticas son difíciles de obtener. En estos casos, pueden resolverse las ecuaciones involucradas mediante aproximaciones obtenidas por métodos numéricos. El método de elementos finitos es la metodología más recurrente en la práctica tanto académica como industrial. Por este motivo, las metodologías de cálculo del factor de intensificación de esfuerzos presentadas en esta tesis utilizan la solución del campo de desplazamientos y los tensores de esfuerzos y deformaciones obtenidos con el método de elementos finitos.

Por otro lado, para obtener soluciones satisfactorias con el método de elementos finitos, el tema de la discretización del cuerpo geométrico es relevante. Se incluye en este capítulo una sección dedicada a las mallas geométricas, donde se explica a grandes rasgos los aspectos teóricos detrás de la generación de mallas de triángulos en 2D. Se introducen conceptos relacionados con la calidad de las mallas, la generación y el refinamiento de éstas, las ventajas y desventajas de las distintas metodologías, y su vínculo con la mecánica de la fractura.

#### 3.1 Elementos finitos

##### 3.1.1 Introducción

A grandes rasgos, el método de elementos finitos consiste en discretizar un espacio mediante una malla geométrica para encontrar una solución aproximada a un problema complejo, como por ejemplo una ecuación diferencial. Por ejemplo, si lo que se busca es un campo en el espacio (2D) de la forma

$$U = U(x, y) \quad (3.1)$$

que satisface una ecuación diferencial, el método de elementos finitos consiste en transformar el problema continuo en uno discreto, donde se busca una solución de la forma

$$\tilde{U} = \sum C_i F_i(x, y) \quad (3.2)$$

es decir, una solución aproximada  $\tilde{U}$  de  $U$ , en función de una base finita de funciones conocidas  $F_i$ . Esto reduce el problema a un sistema de ecuaciones lineales

$$Ax = b \quad (3.3)$$

donde

$$x = [c_1, c_2, \dots, c_{n-1}, c_n]^T \quad (3.4)$$

Las constantes  $c_i$  están relacionadas con las constantes  $C_i$  necesarias para encontrar la solución. Dependiendo del tipo de problema a solucionar (mecánica de sólidos, mecánica de fluidos, electromagnetismo, etc) la forma en la que se plantea el sistema de ecuaciones es distinto. En esta tesis se explicará el planteamiento para la mecánica de sólidos.

### 3.1.2 Elementos finitos en la mecánica de sólidos

#### Introducción

Para efectos de esta tesis, interesa utilizar el método de elementos finitos en la mecánica de sólidos. Dado que el trabajo realizado se enmarca en la teoría de la mecánica de la fractura elástica lineal, interesa en particular el uso de elementos finitos en problemas de elasticidad Zienkiewicz & Taylor (2000, cap. 2).

La aproximación de un problema continuo de elasticidad por medio del método de elementos finitos puede describirse en los siguientes pasos

- El continuo se discretiza por medio de una malla geométrica, en un grupo de superficies (2D) o volúmenes (3D), llamados elementos finitos.
- Los elementos están conectados entre sí por medio de un conjunto de nodos ubicados en sus bordes o en su interior. Las incógnitas del problema de elementos finitos consisten en los desplazamientos en los nodos.
- El campo de desplazamientos en todo el continuo es obtenido a partir de los desplazamientos en los nodos. Se elige un grupo de funciones que define únicamente los desplazamientos en el interior de cada elemento finito, a partir de los nodos pertenecientes a éste.
- A partir de los desplazamientos, definidos en el punto anterior, se puede definir únicamente el estado de deformaciones. A partir del estado de deformaciones, más las propiedades del material y alguna deformación inicial (por ejemplo deformaciones por efectos térmicos) se obtiene el estado de esfuerzos en cada elemento finito.
- Se determina un sistema de fuerzas concentradas en los nodos para formar una relación de rigidez, que es en el fondo la matriz de coeficientes del sistema de ecuaciones a resolver para encontrar los valores nodales.

Los problemas de elasticidad en sólidos abarcan desde problemas 2D de esfuerzo plano y deformación plana, problemas axisimétricos, placas, hasta problemas genéricos tridimensionales. Esta tesis abarca los casos 2D de esfuerzo plano y deformación plana.

### Formulación de los elementos finitos

Siguiendo el procedimiento enunciado en la sección anterior, el primer paso es discretizar el continuo en un conjunto de elementos finitos conectados entre sí por medio de nodos, como muestra la figura 3.1. El ejemplo mostrado en esta sección es un problema 2D, y está discretizado por medio de elementos finitos triangulares de tres nodos. Pueden obtenerse expresiones similares en el caso de 3D, así como expresiones para elementos no triangulares.

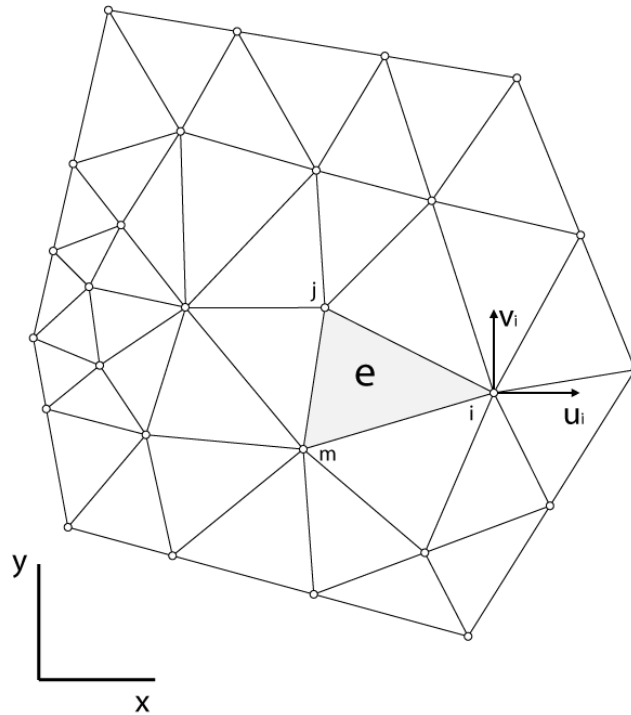


Figura 3.1: Discretización del continuo en una malla geométrica de elementos finitos, y definición de nodos y desplazamientos en el elemento.

Un elemento finito  $e$  puede contener un número variable de nodos. En el ejemplo mostrado en la figura 3.1 los elementos tienen 3 nodos, ubicados en los vértices, aunque pueden tener más nodos. El desplazamiento  $U$  en cualquier punto  $(x,y)$  (en 2D) puede ser aproximado por un vector  $\tilde{U}$

$$U \approx \tilde{U} = \sum_k N_k a_k^e = [N_i, N_j, \dots] \begin{Bmatrix} a_i \\ a_j \\ \vdots \end{Bmatrix} = N a^e \quad (3.5)$$

donde las componentes de  $N_i$ , llamadas *funciones de forma*, son funciones asociadas a cada nodo y  $a^e$  es un vector con los desplazamientos nodales para el elemento  $e$ . Para el caso 2D

$$U = \begin{Bmatrix} u(x,y) \\ v(x,y) \end{Bmatrix} \quad (3.6)$$

representan los desplazamientos horizontal y vertical, y

$$a_i = \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} \quad (3.7)$$

los desplazamientos en el nodo  $i$ .

Una vez conocido el campo de desplazamientos en cualquier punto en el interior del elemento finito  $e$ , puede obtenerse el estado de deformaciones en su interior. La relación entre desplazamientos y deformaciones puede escribirse en forma matricial

$$\tilde{\varepsilon} \approx \varepsilon = Su \quad (3.8)$$

donde  $S$  es un operador lineal. Dado que el desplazamiento se expresa a partir de las funciones de forma y los valores nodales, utilizando la ecuación 3.5 se puede escribir

$$\tilde{\varepsilon} \approx \varepsilon = Ba \quad (3.9)$$

donde

$$B = SN \quad (3.10)$$

es decir el operador  $S$  actúa sobre las funciones de forma, que son conocidas. En el caso de esfuerzo plano

$$\varepsilon = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x}, 0 \\ 0, \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y}, \frac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix} \quad (3.11)$$

La relación entre los esfuerzos y las deformaciones, en el caso de un material lineal elástico, está dada por

$$\sigma = D(\varepsilon - \varepsilon_0) + \sigma_0 \quad (3.12)$$

donde  $\varepsilon_0$  corresponde a deformaciones previas producidas por ejemplo por cambios de temperatura y  $\sigma_0$  corresponde a algún esfuerzo residual conocido, y  $D$  es una matriz de elasticidad que contiene las propiedades del material. Los problemas con esfuerzos residuales y deformaciones previas están fuera del alcance de esta tesis.

Para el caso 2D plano, el esfuerzo está dado por el vector

$$\sigma = \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \quad (3.13)$$

y la matriz D, para materiales lineales isotrópicos y esfuerzo plano, está dada por

$$D = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \quad (3.14)$$

y para deformación plana

$$D = \frac{E}{(1+\nu)(1+2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & (1-2\nu)/2 \end{bmatrix} \quad (3.15)$$

Los materiales anisotrópicos están fuera del alcance de esta tesis.

### 3.1.3 Funciones de interpolación de los elementos

#### Introducción

Como se menciona en la sección anterior, el campo de desplazamientos así como el estado de esfuerzos y deformaciones en el interior de un elemento finito se obtiene interpolando a partir de los valores nodales. Esta interpolación depende del tipo de elemento finito y el orden de las funciones de forma. En la figura 3.2 Se muestran dos tipos de elementos en 2D, rectangulares y triangulares, y elementos tetraedros y prismáticos (“brick”) en 3D, para primer y segundo orden de interpolación. El orden de la interpolación está dado por la cantidad de nodos que tiene el elemento. Por ejemplo, en el caso de un elemento triangular, una interpolación de primer orden (es decir, utilizando como funciones de forma polinomios de grado 1) se obtiene utilizando tres nodos por elemento, uno en cada vértice. Una interpolación de segundo orden (es decir, utilizando funciones de forma cuadráticas) se obtiene utilizando seis nodos, tres en los vértices y tres en los puntos medios de las aristas. Se pueden utilizar interpolaciones del orden que se desee, tanto para elementos triangulares o rectangulares. En el caso de la elasticidad, lo más común es utilizar elementos de segundo orden. En esta tesis se utilizan elementos triangulares de 6 nodos.

#### Elementos triangulares

Las funciones de interpolación para un elemento finito están asociadas a cada nodo. Es decir, en una interpolación de orden cuadrático hay 6 nodos, por lo tanto la interpolación es la combinación lineal de 6 funciones de interpolación. De esto último se desprende que las funciones deben ser linealmente independientes.

Para introducir las funciones de forma de segundo orden primero se definen las *coordenadas de área*. Éstas son funciones que definen únicamente una coordenada en el interior de un elemento triangular a partir de las coordenadas de los nodos. Se definen las coordenadas  $L_1$ ,  $L_2$  y  $L_3$  para un triángulo 123 (figura 3.3) por una relación lineal entre este sistema de coordenadas y las coordenadas cartesianas

$$\begin{aligned} x &= L_1x_1 + L_2x_2 + L_3x_3 \\ y &= L_1y_1 + L_2y_2 + L_3y_3 \\ 1 &= L_1 + L_2 + L_3 \end{aligned} \quad (3.16)$$

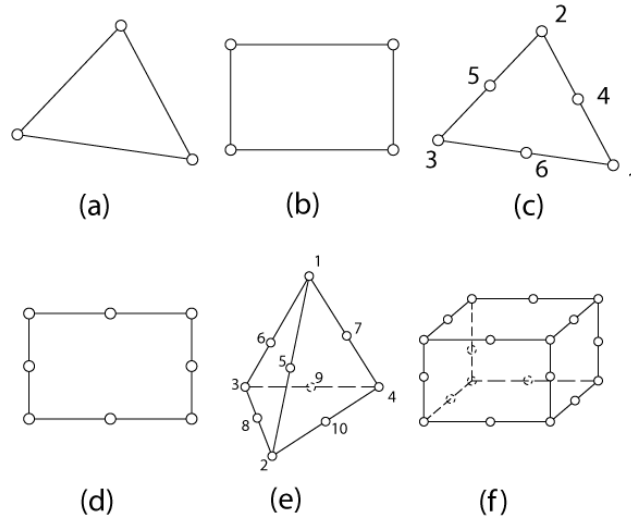


Figura 3.2: Tipos de elementos utilizados y orden de la interpolación. (a) elemento triangular lineal. (b) Elemento rectangular lineal. (c) Elemento triangular cuadrático. (d) Elemento tetraedro cuadrático. (e) Elemento prismático cuadrático. (f)

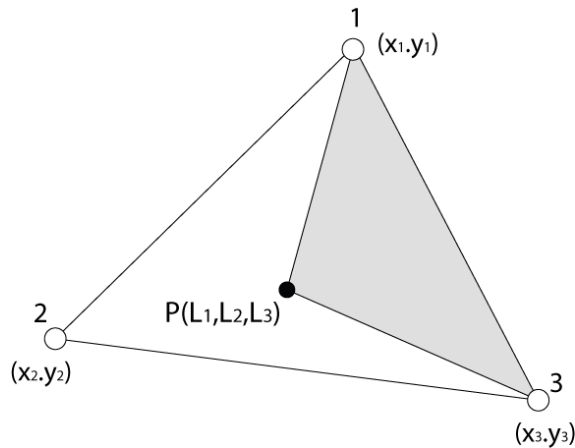


Figura 3.3: Coordenadas de área.

Para las coordenadas de área se cumple que en el punto  $i$ ,  $L_i = 1$  y  $L_{j \neq i} = 0 \forall i = 1, 2, 3$ . Una buena definición de  $L_i$  es el radio entre el área sombreada de la figura 3.3 (es decir el área del triángulo formado por el punto P y los dos puntos  $j \neq i$ ) y el área total de triángulo. Por ejemplo para  $L_1$

$$L_i = \frac{\text{area}(P23)}{\text{area}(123)} \quad (3.17)$$

Por eso el nombre de coordenadas de área. Resolviendo la ecuación 3.16

$$L_1 = \frac{a_1 + b_1x + c_1y}{2\Delta} \quad (3.18)$$

$$L_2 = \frac{a_2 + b_2x + c_2y}{2\Delta} \quad (3.19)$$

$$L_3 = \frac{a_3 + b_3x + c_3y}{2\Delta} \quad (3.20)$$

donde

$$\begin{aligned} a_1 &= x_2y_3 - x_3y_2 \\ b_1 &= y_2 - y_3 \\ c_1 &= x_3 - x_2 \end{aligned} \quad (3.21)$$

lo mismo para las otras constantes, rotando cíclicamente los índices 1, 2 y 3.

El área de un triángulo puede calcularse como

$$\Delta = \text{area}_{123} = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \quad (3.22)$$

El área de un triángulo (123) se debe calcular enumerando los índices en sentido contrario a las manecillas del reloj (figura 3.3 c), para que el determinante sea un número positivo.

Una vez definidas las coordenadas de área, se pueden definir las funciones de forma para un elemento triangular de segundo orden. Éstas se definen en función de las coordenadas de área. Siguiendo la numeración de la figura 3.2 las funciones de forma están dadas por (Zienkiewicz & Taylor, 2000, cap. 8.8.2)

$$\begin{aligned} N_1 &= (2L_1 - 1)L_1 \\ N_2 &= (2L_2 - 1)L_2 \\ N_3 &= (2L_3 - 1)L_3 \\ N_4 &= 4L_1L_2 \\ N_5 &= 4L_2L_3 \\ N_6 &= 4L_3L_1 \end{aligned} \quad (3.23)$$

El desplazamiento puede encontrarse utilizando estas funciones de forma según la definición de 3.5.

### 3.1.4 Error

El método de elementos finitos entrega una aproximación de la solución real del problema por medio de una discretización del continuo. Se define el error como la diferencia entre la solución aproximada y la exacta, siendo

$$\begin{aligned} e &= U - \tilde{U} \\ e &= \varepsilon - \tilde{\varepsilon} \\ e &= \sigma - \tilde{\sigma} \end{aligned} \quad (3.24)$$

el error local para el desplazamiento, deformación y esfuerzo, respectivamente. En la práctica es más útil definir el error mediante una norma, un valor integrado sobre una zona del continuo. En el caso de la elasticidad, la *norma energética* se expresa equivalentemente como (Zienkiewicz & Taylor, 2000, cap 14.1)

$$\begin{aligned}\|e\| &= \left[ \int_{\Omega} (\varepsilon - \tilde{\varepsilon}) D (\varepsilon - \tilde{\varepsilon}) d\Omega \right]^{1/2} \\ \|e\| &= \left[ \int_{\Omega} (\varepsilon - \tilde{\varepsilon}) D (\sigma - \tilde{\sigma}) d\Omega \right]^{1/2} \\ \|e\| &= \left[ \int_{\Omega} (\sigma - \tilde{\sigma}) D^{-1} (\varepsilon - \tilde{\varepsilon}) d\Omega \right]^{1/2}\end{aligned}\tag{3.25}$$

donde  $D$  es la matriz de elasticidad definida anteriormente para 2D y 3D. Se intuye que el error de la aproximación depende de la densidad de nodos que existe en la discretización, o equivalentemente, del tamaño característico  $h$  de los elementos utilizados. A medida que  $h \rightarrow 0$ , la aproximación se acerca a la solución exacta. Se puede mostrar (Zienkiewicz & Taylor, 2000, cap. 2.6) que el error asociado a un elemento de tamaño  $h$  es de orden  $O(h^{p+1})$ , donde  $p$  es el orden de la función de interpolación. Por lo tanto, en el caso de utilizar funciones cuadráticas, el error es de orden  $O(h^3)$ . Sin embargo, en caso que exista una singularidad, la convergencia del error es distinta, y es de orden  $O(h^\lambda)$ , con  $\lambda = 0,5$  para una grieta casi cerrada hasta  $\lambda = 0,71$  para una esquina en ángulo recto.

### 3.2 Mallas geométricas

El problema de generar una malla para resolver un problema con el método de elementos finitos involucra varios aspectos, como son el tipo de malla a utilizar, la calidad de la malla, el nivel de refinamiento, la eficiencia de la generación de la malla. A continuación se abordarán estos conceptos para entender mejor los pasos futuros.

#### 3.2.1 Definiciones básicas

**Malla** Dado un dominio cerrado  $\Omega$  en  $R^2$  o  $R^3$ , una malla  $T$  es un conjunto de elementos geométricos tales que

- La unión de todos los elementos  $\bigcup_{K \in T} K = \Omega$
- El interior de cualquier elemento  $K \in T$  es no vacío
- La intersección de dos elementos vecinos es una arista común o un vértice común.

En particular en el método de elementos finitos interesan las mallas conformes. Una malla  $T$  se dice que es conforme si

- Es una malla
- La intersección de dos elementos en  $T$  es vacía, un vértice, una arista o una cara (en 3D).

En la figura 3.4 se aprecia la diferencia entre una malla conforme y una no conforme.



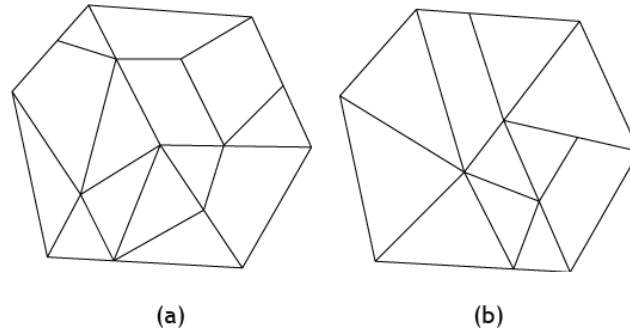


Figura 3.4: Malla conforme (a) y malla no conforme (b). Nótese que el vértice de un elemento cae sobre la arista de otro.

**Triangulación Delaunay** Una triangulación es un caso particular de una malla, donde todos los elementos son triángulos en 2D y tetraedros en 3D. Este caso particular es importante dado que en esta tesis se utilizarán triangulaciones para discretizar el continuo.

Una triangulación puede definirse a partir de un conjunto de puntos. Dado un conjunto de puntos, son posibles varias mallas de triángulos, una de éstas es la denominada triangulación Delaunay, de gran interés para el método de elementos finitos.

Sea  $S$  un conjunto de puntos y  $\Omega$  el conjunto convexo de  $S$ , es decir, un polígono que encierra a todos los puntos de  $S$ . Una triangulación  $T$  de  $\Omega$  es Delaunay si los círculos (o esferas en 3D) circunscritos a cada triángulo  $K \in T$  no contienen ningún vértice de  $S$ . La figura 3.5 muestra una triangulación Delaunay en 2D.

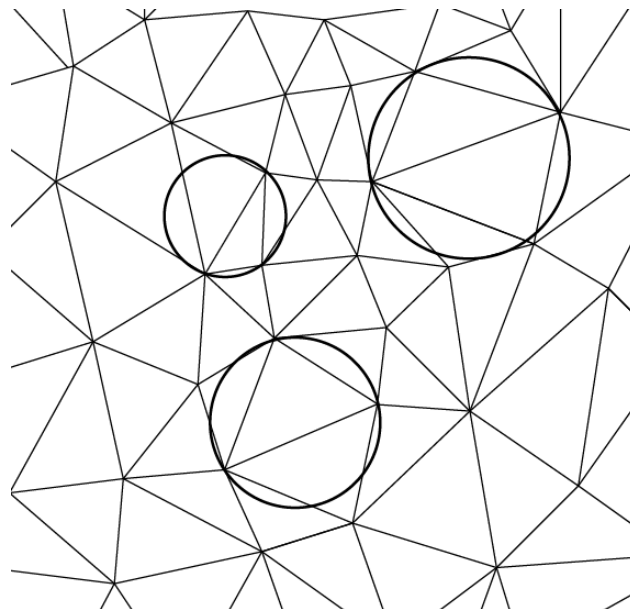


Figura 3.5: Triangulación Delaunay en 2D. El círculo circunscrito de cualquier elemento no encierra ningún punto de  $S$ .

**Malla de elementos finitos** Las mallas mencionadas hasta ahora son sólo entidades geométricas. Para efectos del método de elementos finitos, las mallas no sólo tienen propiedades geométricas sino que tienen atributos extra necesarios para que el método funcione. Éstas propiedades están asociadas a la implementación computacional del método.

- En una malla de elementos finitos existen elementos, que son los mismos elementos de la malla. Estos elementos están indexados correlativamente. Existen varios tipos de elementos, que varían según la cantidad de lados (o caras en 3D) que tiene, o la cantidad de nodos asociados.
- También existen nodos. Un nodo es un punto al cual se le asocian las incógnitas del modelo, o grados de libertad, que en el caso de la elasticidad corresponden a las componentes del desplazamiento. Los nodos también están indexados correlativamente. La cantidad de nodos está asociada al elemento elegido, y al orden de las funciones de interpolación (ver cap. 3.1.3). Algunos ejemplos de elementos y sus nodos asociados se ven en la figura 3.2.
- En el método de elementos finitos interesan ciertas características de la malla, como por ejemplo el tamaño de los elementos o nivel de refinamiento de la malla. El error de la solución obtenida está ligado al tamaño de los elementos. Por otro lado, al aumentar el número de grados de libertad (que está asociado a la cantidad de nodos, y ésta a la cantidad y por lo tanto al tamaño de los elementos) aumenta el tiempo de computación de la solución. Por eso las mallas utilizadas deben ser “óptimas” en el sentido que los elementos deben tener un tamaño sólo lo suficientemente pequeño para garantizar un nivel de error razonable, evitando grados de libertad innecesarios. El tamaño de los elementos puede variar drásticamente de un lugar a otro del continuo.
- También interesa lo que se denomina la calidad de los elementos, en particular en elementos triangulares. Un elemento triangular (o tetraedro) es de buena calidad mientras su forma sea más cercana a la de un triángulo (o tetraedro) equilátero. Existen variadas formas de medir la calidad, como midiendo sus ángulos agudos u obtusos. Para asegurar la convergencia del método de elementos finitos, es necesario utilizar elementos de buena calidad (Babuska & Aziz, 1976),(Zienkiewicz & Taylor, 2000).

### 3.2.2 Métodos de mallado

A continuación se describirán brevemente los tres métodos de mallado de geometrías más utilizados: el método Delaunay, el frente de avance y el método quadtree/octree.

#### **Método quadtree/octree**

En este método, antes de mallar la geometría, ésta se aproxima por una unión de celdas de tamaño variable, que se obtienen a partir del refinamiento recursivo de una celda mayor que encierra a la geometría. Luego a partir de estas celdas se generan los elementos triangulares. La aproximación por medio del quadtree (u octree en 3D) permite generar elementos de tamaño variable a lo largo de la geometría, obteniéndose elementos más finos en las zonas donde la geometría es más intrincada. Una vez obtenida la malla a partir

del quadtree/octree, se mejora su calidad utilizando un suavizamiento laplaciano. Éste consiste en reubicar los nodos en el centroide del polígono formado por los triángulos que lo comparten. La secuencia puede verse en la figura 3.6

### **Método del frente de avance**

El método del frente de avance consiste en generar una malla a partir del borde (o superficie) de la geometría, creando la malla secuencialmente, elemento por elemento, insertando puntos y conectándolos a los elementos ya creados. A medida que se van creando elementos, la malla creada va avanzando en las zonas de la geometría que no han sido malladas, hasta que esta zona desaparece. La gran ventaja de este método es que genera elementos de buena calidad pues al crearse los elementos uno por uno, puede controlarse la calidad del elemento desde su creación. Una desventaja es que pueden haber problemas de convergencia, sobre todo en 3D, pues hay ciertas geometrías que son más difíciles de mallar con este método. En la figura 3.7 se observa la secuencia de mallado utilizando este método. El mallado siempre parte por generar una discretización del borde.

### **Mallado Delaunay**

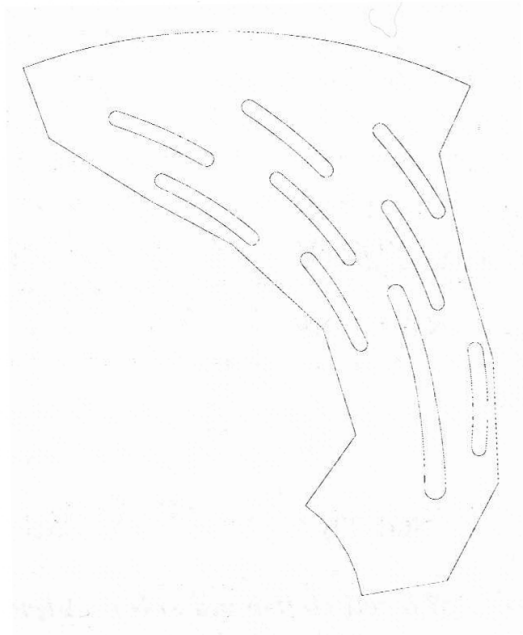
El mallado Delaunay se basa en el concepto de triangulación Delaunay para generar una malla. En términos generales, el algoritmo parte por encerrar la geometría a mallar dentro de un espacio convexo. Luego se insertan una serie de puntos en los bordes de la geometría, cuya densidad determinará el tamaño de los elementos posteriores. Una vez insertados estos elementos, se efectúa un post-proceso para asegurar el respeto de bordes, y luego se borran los triángulos que no están en el interior de la geometría. Con esto se obtiene una malla inicial Delaunay, la que es mejorada insertando incrementalmente puntos en el interior de la geometría. Los algoritmos Delaunay incrementales tienen la propiedad de que dada una triangulación Delaunay  $T_i$  con  $i$  puntos, se puede generar una triangulación Delaunay  $T_{i+1}$  insertando un punto en el interior de alguno de los elementos y verificando que los nuevos triángulos cumplen con la propiedad Delaunay. Los criterios para la inserción incremental de puntos en la malla son variados. En la figura 3.8 se observa el mallado de una geometría por medio de la metodología mencionada.

El software desarrollado por Calderón & Rivara (2004) utiliza una versión incremental del algoritmo de Delaunay, que inserta los nodos directamente en el borde, sin usar una geometría auxiliar. La mejora se realiza mediante un algoritmo Lepp-Delaunay (Rivara *et al.*, 2001).

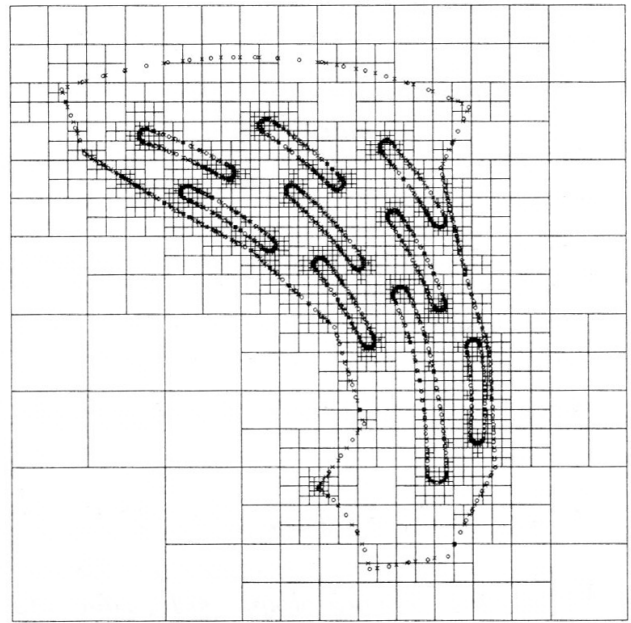
### **3.2.3 Refinamiento**

#### **Introducción**

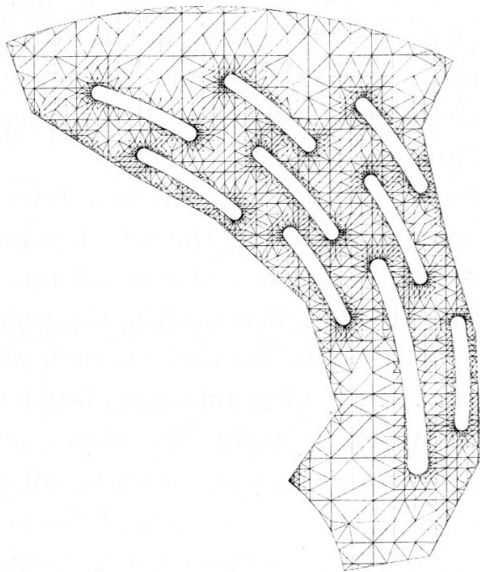
El error en el método de elementos finitos depende del tamaño local de los elementos. En problemas de elasticidad, cuando un sólido arbitrario es sujeto a una carga, zonas con muescas, cambios de sección o grietas generan gradientes de esfuerzos mayores que en una geometría más suave. Por lo tanto para encontrar una solución con un error adecuado es necesario utilizar en esas zonas una malla más fina que en el resto del cuerpo. Por esto, es de gran importancia tener herramientas para modificar una malla ya existente y



i)



ii)



iii)



iv)

Figura 3.6: Secuencia para el mallado basado en un quadtree (2D). (i) geometría original, (ii) generación del quadtree, (iii) malla obtenida a partir del quadtree y (iv) malla final mejorada.(figura obtenida de Frey & George (2000)).

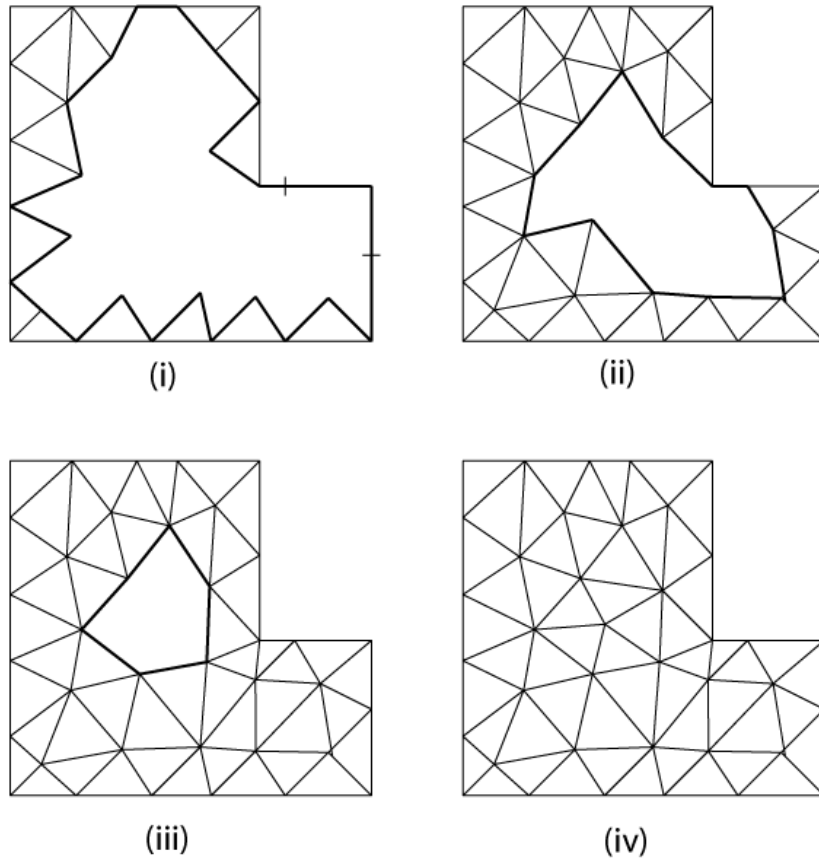


Figura 3.7: Secuencia de pasos para el mallado con la técnica del frente de avance.

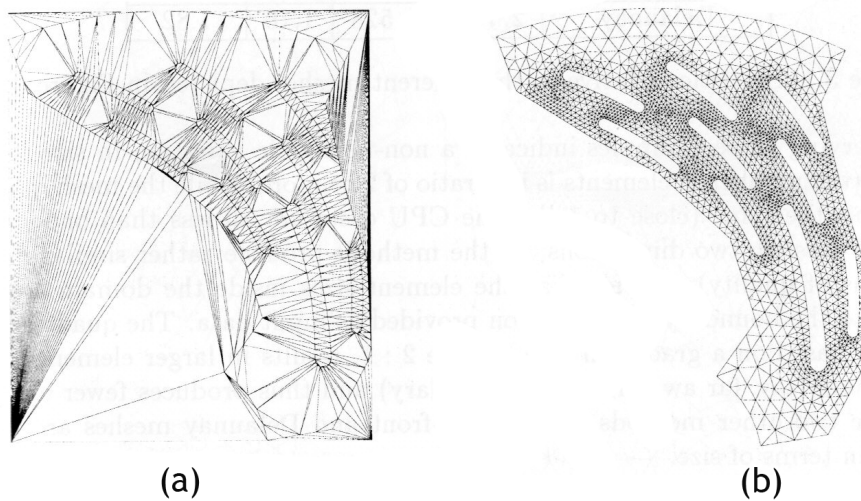


Figura 3.8: Mallado basado en una triangulación Delaunay de una geometría 2D. La geometría se encierra en un rectángulo (a) y se crea una triangulación Delaunay a partir de puntos en el borde de la geometría. Luego (b), se eliminan los triángulos externos y se mejora la malla. (figura obtenida de Frey & George (2000)).

encontrar un tamaño local óptimo de elementos, de tal forma de utilizar elementos grandes en zonas de poca variación de esfuerzos, y elementos más pequeños en zonas de altos gradientes: este es el objetivo del refinamiento localizado. En la figura 3.9 se observa un típico caso de refinamiento localizado cerca de una muesca, para optimizar el error. Las grietas, que es el tema de esta tesis, son un caso particular de la necesidad de refinamiento localizado.

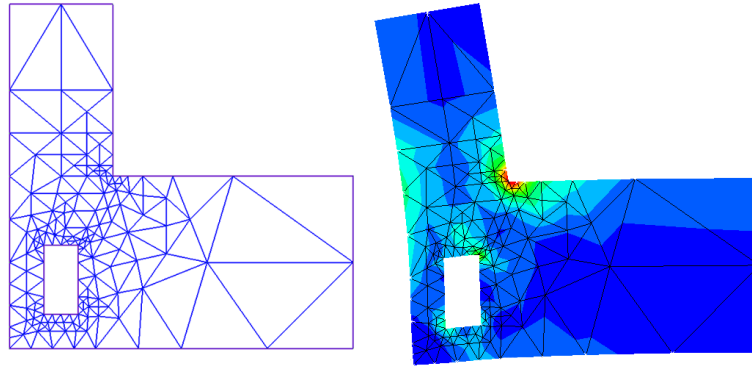


Figura 3.9: Refinamiento localizado cerca de una zona donde existe un alto gradiente de esfuerzos.

### Refinamiento basado en el LEPP

Existen variadas formas de refinar mallas de triángulos. Dentro de éstas existen los algoritmos de refinamiento/mejoramiento basados en el LEPP (*Longest Edge Propagation Path*), introducidos por Rivara & Palma (1997) y luego estudiados empíricamente (Rivara *et al.*, 2001). Al refinar un elemento, normalmente no sólo ese elemento es modificado sino también 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 3.10. Al refinar, sólo los elementos que están en el LEPP serán afectados. Estos algoritmos garantizan que siempre existirá una malla válida mientras se refina.

Estos algoritmos pueden ser Delaunay o no Delaunay. La diferencia radica en que si al insertarse un punto en la malla se hace o no una comprobación si la nueva malla es Delaunay. Existen básicamente 3 variaciones de este algoritmo:

**Lepp-Bisección** En este caso se bisectan por el lado más largo los triángulos que comparten la arista terminal, lo que garantiza que la calidad de los triángulos se mantiene a través del proceso (Rivara, 1984, 1997). Este algoritmo tiene la ventaja de generar un mallado más rápido de manera robusta, 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 algo-

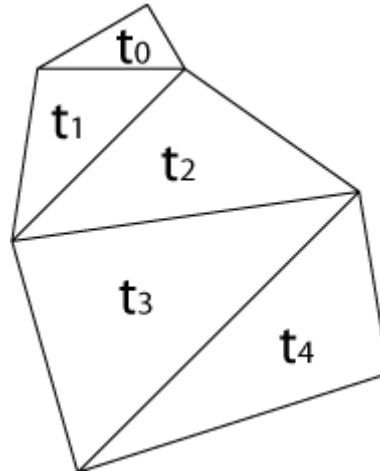


Figura 3.10: “Longest Edge Propagation Path (LEPP)”.  $t_0$  es el triángulo a refinar, y los triángulos incluidos en el LEPP son los que se verán afectados por el refinamiento, para mantener una malla conforme. La última arista más larga se denomina arista terminal.

ritmo de desrefinamiento sobre mallas generadas con este algoritmo. La figura 3.11 ilustra el refinamiento de un triángulo  $t_0$  utilizando el algoritmo Lepp-Bisección.

---

**Algorithm 1** Lepp-Biseccion( $t, T$ ),  $T$  una malla de triángulos,  $t \in T$  un triángulo

---

```

while t permanezca sin ser biseccionado do
  Encontrar LEPP(t)
   $t^* =$  último triángulo del LEPP(t)
  if  $t^*$  es borde then
    Biseccionar  $t^*$ 
  else { $t^*$  no es borde}
    Biseccionar el par de triángulos terminales de LEPP(t)
  end if
end while

```

---

**Lepp-Delaunay-AristaTerminal** Este realiza sucesivamente una 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 que depende de la triangulación inicial). En la figura 3.12 se observa la secuencia de refinamiento del algoritmo.

---

**Algorithm 2** Lepp-Delaunay-AristaTerminal( $t, T$ ),  $T$  una malla de triángulos,  $t \in T$  un triángulo

---

```

while t no sea modificado do
  Encontrar LEPP(t)
  Realizar una inserción Delaunay del punto p en el punto medio de la arista terminal de LEPP(t)
end while

```

---

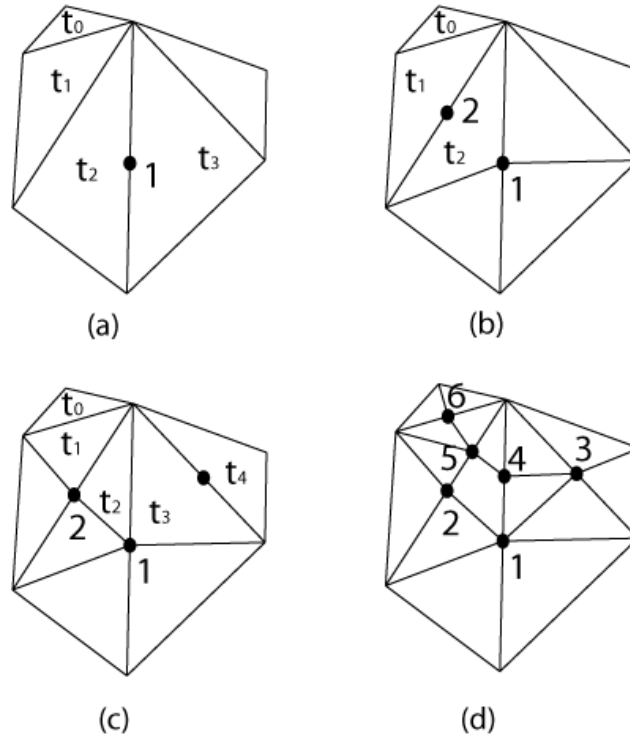


Figura 3.11: Refinamiento de un triángulo  $t_0$  utilizando el algoritmo Lepp-Bisección. (a) triangulación inicial; (b) primer paso del proceso; (c) segundo paso del proceso; (d) triangulación final.

**Lepp-Delaunay-CentroidesTT** 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 3.13 se ilustra el algoritmo.

---

**Algorithm 3** Lepp-Delaunay-CentroidesTT( $t, T$ ),  $T$  una malla de triángulos,  $t \in T$  un triángulo

---

**while**  $t$  no sea modificado **do**

    Encontrar LEPP( $t$ )

    Realizar una inserción Delaunay del punto  $p$  en el centroide del cuadrilátero formado por dos últimos triángulos del LEPP

**end while**

---

En esta tesis al referirse a un “algoritmo basado en el LEPP” se estará hablando de cualquiera de las variantes recién mencionadas. Al referirse a un “algoritmo Lepp-Delaunay” se estará hablando del algoritmo *Lepp-Delaunay-AristaTerminal* o del algoritmo *Lepp-Delaunay-CentroidesTT*. Estos algoritmos son usados para generar mallas iniciales, para mejorar mallas existentes o para refinar localizadamente una malla, dependiendo del contexto en que se usen, por lo que se omitirá el prefijo “refinamiento” o “mejoramiento”.



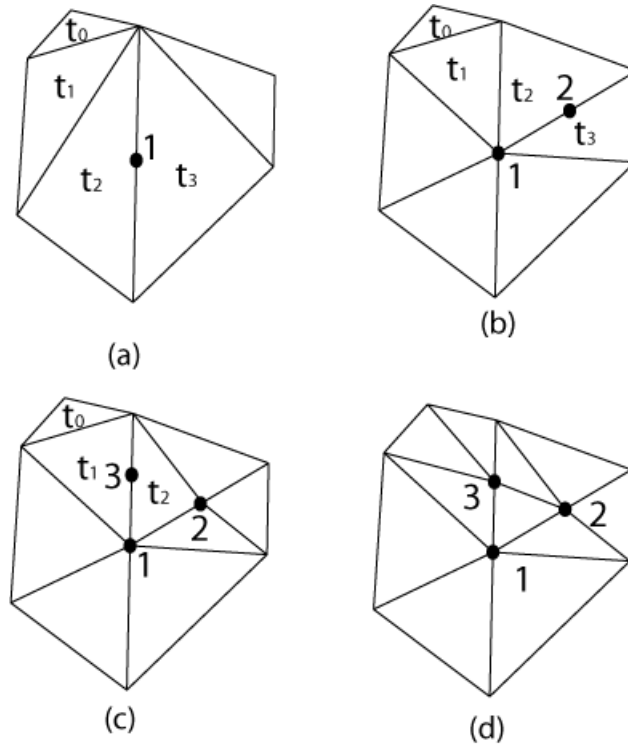


Figura 3.12: Refinamiento de un triángulo  $t_0$  utilizando el algoritmo Lepp-Delaunay. En el Algoritmo Lepp-Delaunay, después de insertar un punto se verifica que los triángulos generados sigan siendo Delaunay. Las inserciones continúan hasta que el triángulo objetivo es modificado.

### 3.2.4 Mejoramiento

Otro aspecto importante en el ámbito de las mallas geométricas es el mejoramiento de una malla existente. Ya se mencionó la importancia de tener elementos de buena calidad, y su relación con la convergencia del método de elementos finitos (Babuska & Aziz, 1976). Recordando el capítulo 3.2.2, se observa que los métodos de generación de mallas tienen un paso final donde la malla es mejorada. El mejoramiento consiste en convertir una malla con triángulos predominantemente de mala calidad en una malla con triángulos predominantemente de buena calidad.

#### Indicadores de calidad

Para cuantificar la calidad de un triángulo existen distintos indicadores asociados a sus variables geométricas. Se define el triángulo (o el tetraedro) equilátero como el elemento de mejor calidad posible, por lo que un indicador de calidad puede ser un número que vale 1 si es un triángulo equilátero, y es mayor o menor que 1 si no lo es. Por ejemplo, la calidad de un elemento  $t$  puede ser expresada como (web, n.d.)

$$q(t) = f \frac{A}{a^2 + b^2 + c^2} \quad (3.26)$$

donde  $A$  es el área del triángulo  $t$ ;  $a$ ,  $b$  y  $c$  son el largo de sus tres lados y  $f = 4\sqrt{3}$  es un coeficiente de normalización para que  $q = 1$  si el triángulo es equilátero. Análogamente, para un tetraedro

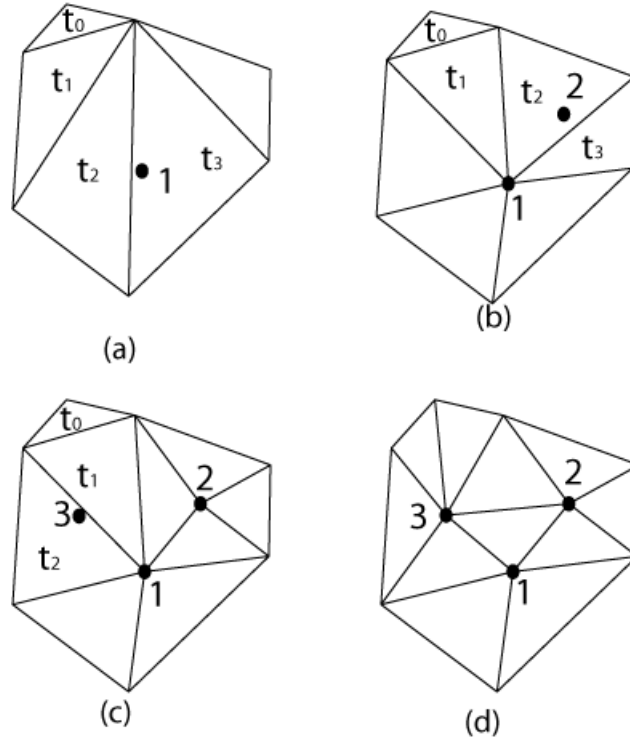


Figura 3.13: Refinamiento de un triángulo  $t_0$  utilizando el algoritmo Lepp-Centroide. La diferencia con el algoritmo Lepp-Delaunay consiste en que el punto no se inserta en el punto medio de la arista terminal, sino en el centroide del cuadrilátero formado por los dos triángulos terminales.

$$q(t) = f \frac{V^2}{(A+B+C+D)^3} \quad (3.27)$$

donde  $V$  es el volumen del tetraedro,  $A$ ,  $B$ ,  $C$  y  $D$  son el área de cada una de sus caras y  $f = 216\sqrt{3}$  es el coeficiente de normalización.

Otra forma de medir la calidad es a partir de los ángulos. Por ejemplo, para un triángulo

$$q = \frac{\alpha_{min}}{\alpha_{max}} \quad (3.28)$$

donde  $\alpha_{min}$  y  $\alpha_{max}$  son el ángulo mínimo y el máximo del triángulo, respectivamente. Análogamente, para un tetraedro

$$q = \frac{\delta_{min}}{\delta_{max}} \quad (3.29)$$

donde  $\delta_{min}$  y  $\delta_{max}$  son el ángulo diedro mínimo y el ángulo diedro máximo, respectivamente. Claramente  $q = 1$  para un triángulo (o tetraedro) equilátero.

Otros indicadores posibles se muestran a continuación. Estos no necesariamente valen 1 para un triángulo equilátero, pero cumplen con que mientras mayor sea el número, mejor es la calidad

$$q_1 = \frac{l_{min}}{R} \quad (3.30)$$

$$q_2 = \frac{r}{l_{max}} \quad (3.31)$$

$$q_3 = \frac{r}{R} \quad (3.32)$$

$$q_4 = \frac{V}{l^3_{max}} \quad (3.33)$$

Luego, un indicador de la calidad de la malla completa puede ser su media aritmética

$$Q = \frac{1}{N} \sum_{i=1}^N q_i \quad (3.34)$$

o su media armónica

$$Q = \frac{N}{\sum_{i=1}^N \frac{1}{q_i}} \quad (3.35)$$

Donde N es el número total de elementos en la malla. El indicador armónico tiene la ventaja de ser muy sensible a elementos de muy mala calidad.

Además de usar estos indicadores, es necesario conocer el valor del elemento de menor calidad, y la distribución de calidad de los elementos. En la figura 3.14 se observa una malla de triángulos con su respectivo histograma de calidad, utilizando el indicador dado por la ecuación 3.26. Se puede saber, cualitativamente, si una malla es de buena calidad, viendo la distribución de calidad de los elementos.

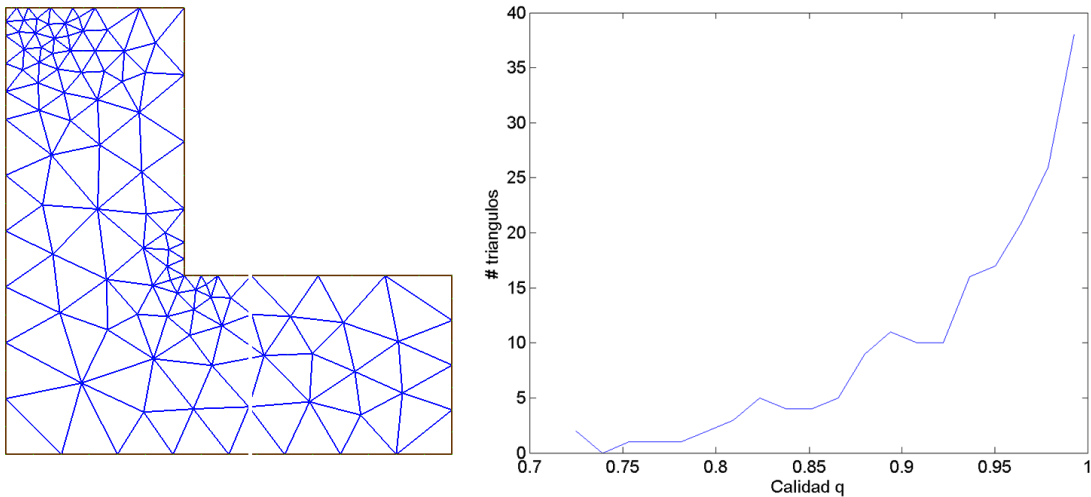


Figura 3.14: Malla geométrica (a) y su respectivo histograma (b) de la calidad de los triángulos. Se utiliza el indicador dado por la ecuación 3.26.

## Mejoramiento con algoritmos LEPP

Los algoritmos Lepp-Delaunay se utilizan no sólo para refinar sino también para mejorar una malla (Rivara, 1997; Rivara *et al.*, 2001). A medida que se van insertando puntos, los nuevos triángulos generados son de mejor calidad que los anteriores. Dado que mejora la malla creando nuevos elementos, no es buena idea aplicar este algoritmo sobre una malla de mala calidad pero que sus elementos son del tamaño deseado, porque se generarán elementos innecesariamente pequeños. La mejor forma de aplicarlo es sobre una malla inicial con nodos sólo en los bordes, como se muestra en la figura 3.15

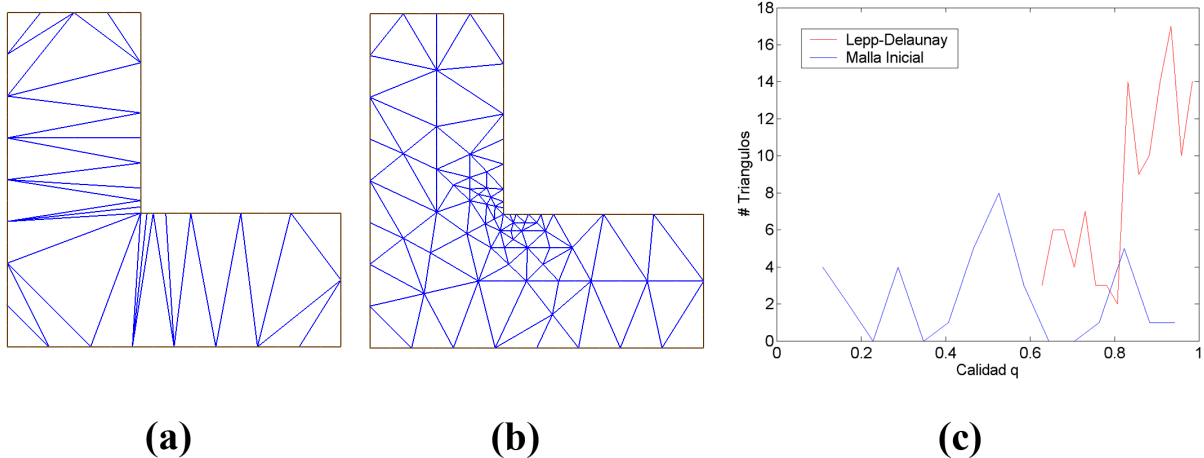


Figura 3.15: (a) malla inicial de un cuerpo, a partir de la inserción de nodos en el borde. (b) Malla final lograda sólo aplicando un algoritmo Lepp-Delaunay sucesivamente sobre los triángulos de mala calidad hasta que desaparezcan. (c) Histograma de calidad para la malla inicial y para la malla después de aplicar el algoritmo Lepp-Delaunay

En la figura 3.15(a) se muestra una geometría donde se han insertado nodos sólo en el borde de ésta, siguiendo un espaciado previamente definido. Cada una de estas inserciones es Delaunay, por lo que la malla inicial, aunque sea una malla de mala calidad, es una malla Delaunay. Se puede obtener una malla final de buena calidad utilizando el algoritmo Lepp-Delaunay (figura 3.15(b)). El algoritmo usado para obtener la malla es

---

**Algorithm 4** MalladoDelaunay( $T$ ),  $T$  una malla de triángulos,  $t \in T$  un triángulo

---

```
for all  $t \in T$  do  
    while  $t$  no cumpla criterio calidad do  
        Lepp-Delaunay( $t$ )  
    end while  
end for
```

---

Se puede demostrar que en cada paso que se aplica el algoritmo Lepp-Delaunay, los nuevos triángulos serán de igual o mejor calidad que los anteriores (Rivara & Hitschfeld, 1999). Un punto a destacar de este algoritmo es que no se generan mallas no-válidas, aspecto relevante a la hora de implementar computacionalmente estas metodologías.

## Suavizamiento Laplaciano

Una de las técnicas más usadas para mejorar la calidad de una malla es el suavizamiento laplaciano. A diferencia del mejoramiento con los algoritmos Lepp-Delaunay, esta técnica se aplica sobre una malla ya existente, pues no inserta nuevos nodos (y por lo tanto nuevos elementos) sino que reposiciona los ya existentes, lo que permite mantener siempre una malla válida.

La implementación más simple del suavizamiento laplaciano consiste en reposicionar cada nodo como el promedio de los nodos adyacentes, como muestra la figura 3.16

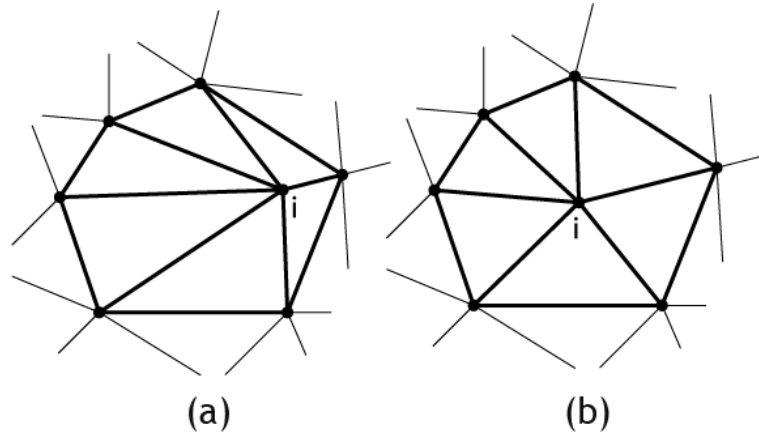


Figura 3.16: (a) Grupo de elementos antes del suavizamiento laplaciano. (b) El nodo  $i$  se reposiciona como el promedio de la posición de los nodos adyacentes, o lo que es lo mismo, el centroide del polígono formado por ellos.

Por lo tanto, para cada nodo  $i$ , la nueva posición está dada por

$$n_x^i = \frac{\sum_{j=1}^N n_x^j}{N} \quad (3.36)$$

$$n_y^i = \frac{\sum_{j=1}^N n_y^j}{N} \quad (3.37)$$

donde  $n_x^i$  y  $n_y^i$  son las coordenadas  $x$  e  $y$  del nodo  $i$ , respectivamente, y  $N$  es el número de nodos que rodea al nodo  $i$ .

Este desplazamiento se realiza sobre todos los nodos. En la figura 3.17 se muestra una malla antes y después del suavizamiento laplaciano.

### 3.3 Técnicas de cálculo de los factores de intensificación de esfuerzos

Para obtener el valor de  $K_I$ , se necesita conocer el campo de desplazamientos y/o los tensores de esfuerzos y deformaciones cerca de la punta de la grieta. Éstos normalmente se obtienen por medio del método de elementos finitos, que es lo que se hace en esta tesis. 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.

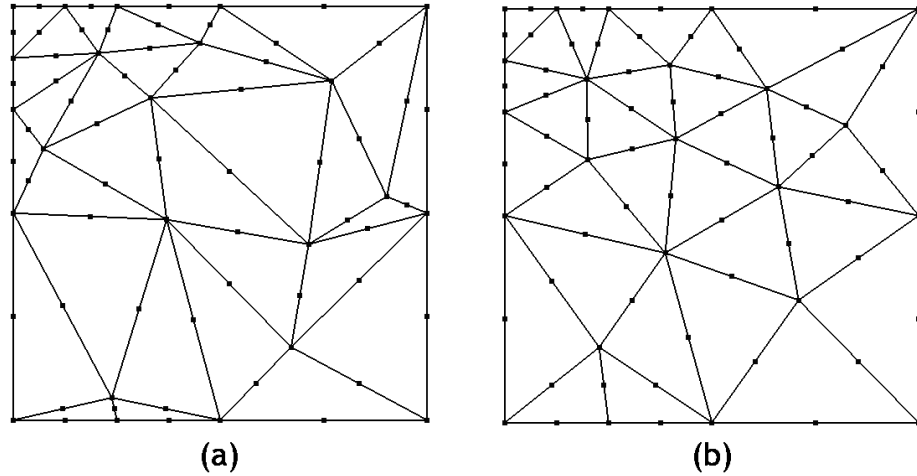


Figura 3.17: (a) Malla Inicial. (b) Malla después del suavizamiento laplaciano.

### 3.3.1 Elementos especiales con nodos desplazados

Como se mostró en el capítulo 2.2.1, en las ecuaciones 2.6, 2.7, 2.9 y 2.10, el tensor de esfuerzos cerca de la punta de la grieta es de la forma  $1/\sqrt{r}$ , y el campo de desplazamientos es de la forma  $\sqrt{r}$ , donde  $r$  es la distancia radial a la punta de la grieta. Dado que las funciones de interpolación normalmente usadas en el método de elementos finitos son polinomiales, se hace muy difícil para este tipo de elementos aproximar adecuadamente el campo cerca de la punta y es necesario un alto grado de refinamiento en esta zona para obtener un resultado razonable. Con el fin de evitar este problema, Barsoum (1976) y Henshell & Shaw (1975) introducen, casi al mismo tiempo, un tipo de elemento especial que permite mejorar esta aproximación. El esquema elemental de la solución propuesta por Barsoum se aprecia en la figura 3.18.

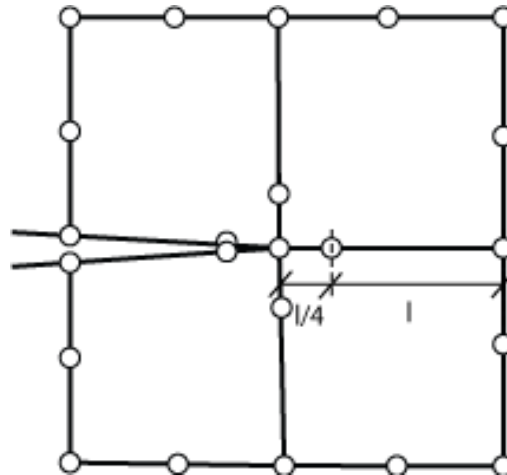


Figura 3.18: Elemento especial básico. El nodo medio de las aristas radiales a la punta de la grieta se desplaza a una posición de  $1/4$  del largo de la arista.

En la figura la malla en la punta de la grieta está formada por 4 elementos isoparamétricos de 8 nodos. En

este método, en cada elemento el nodo medio de las aristas radiales a la punta de la grieta se desplaza a una posición de  $1/4$  del largo de la arista. Al hacer esto y conservando la definición de las funciones de interpolación originales, se genera una aproximación que tiene la misma forma que el campo de desplazamientos en la punta de la grieta, es decir de  $\sqrt{r}$ . Luego, al derivar las funciones de interpolación para obtener la aproximación por elementos finitos de los tensores de esfuerzos y deformaciones, se obtienen aproximaciones de la forma  $1/\sqrt{r}$ , de la misma forma que los tensores dados por las expresiones analíticas. Este método se conoce en inglés como *Quarter Point Element Method* (QPEM). Una descripción más detallada de las funciones de interpolación y de por qué toman esta forma puede verse en Liu & Quek (2003, cap. 10.2), además de los artículos originales.

A partir de este desarrollo inicial se generan otras propuestas. Barsoum (1977) propone otra versión esta vez para elementos triangulares de 6 nodos, como el que se muestra en la figura 3.19 (a). En la literatura, al conjunto de elementos que tocan la punta de la grieta y que son afectados por el desplazamiento de los nodos se le denomina *roseta*. La experiencia muestra que se obtienen mejores resultados si en vez de usar una roseta irregular, se genera una roseta regular y simétrica, como las mostradas en la figura 3.19 (b) y (c). También es sabido que mientras más elementos tenga la punta de la grieta, mejor es la aproximación, por ejemplo ver Guinea *et al.* (2000).

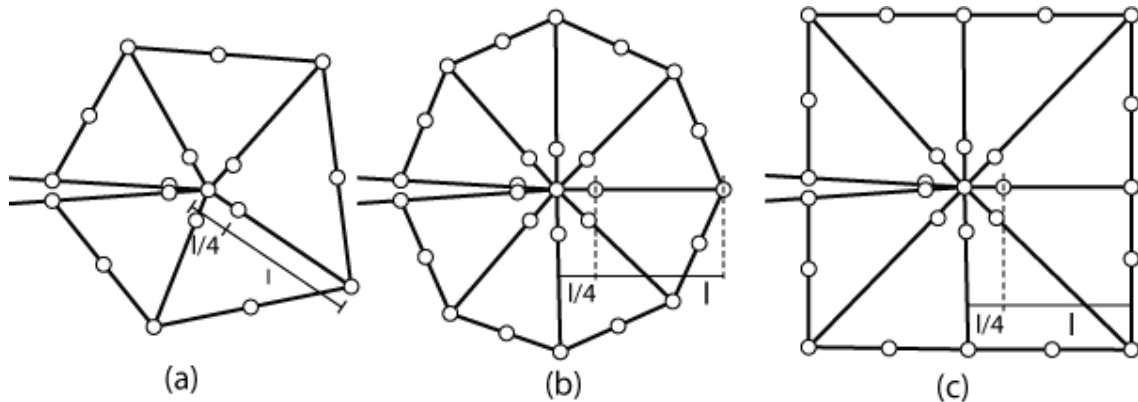


Figura 3.19: (a) Elemento especial con los nodos desplazados para el caso de los elementos triangulares. (b) y (c) El mismo tipo de elementos, pero utilizando rosetas simétricas para obtener mejores resultados.

Otro tipo de roseta es el que se muestra en la figura 3.20. En ésta, se utilizan elementos isoparamétricos de 8 nodos pero tres de sus nodos son colapsados en una misma posición geométrica (pero distintos topológicamente), generando elementos de forma triangular. Con esto, puede mallarse el modelo completo con elementos isoparamétricos, incluyendo la punta de la grieta, sin cambiar las funciones de interpolación y se obtiene una mejor aproximación al tener más aristas radiales. Además, este tipo de elementos es utilizado para modelar problemas elasto-plásticos, pues los tres nodos que están colapsados en la punta de la grieta pueden desplazarse independientemente, pudiendo modelar así la deformación plástica en la punta de la grieta que tiende a redondearla, que es lo que normalmente ocurre en estos problemas.

Este método tiene la función de mejorar la aproximación de elementos finitos del campo cerca de la punta de la grieta. Para calcular los factores de intensificación de esfuerzos se necesita hacer un post-proceso

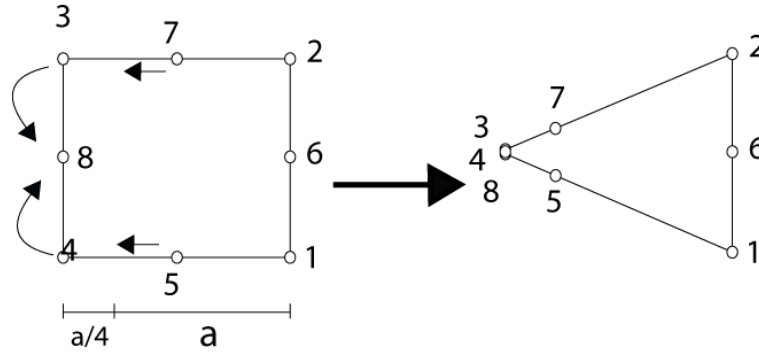


Figura 3.20: Elemento especial isoparamétrico de 8 nodos, donde 3 nodos son colapsados a la misma posición geométrica pero independientes topológicamente. Este tipo de elementos se utiliza para modelar el enromamiento que ocurre en la punta de la grieta en el caso de la fractura elasto-plástica.

sobre los resultados del método de elementos finitos. Por lo tanto este método se utiliza en complemento con los métodos de post-proceso para mejorar la precisión y evitar el refinamiento excesivo. Los métodos de post-proceso se muestran a continuación.

### 3.3.2 Extrapolación de desplazamientos

El método de extrapolación de desplazamientos consiste en calcular los SIF a partir del desplazamiento de los nodos cerca de la punta de la grieta. A partir de la MFEL (Kanninen & Popelar, 1985), se obtiene una expresión analítica que relaciona  $K_I$  y los desplazamientos, utilizando el sistema de referencia de la figura 3.21

$$\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \frac{K_I}{2\mu} \sqrt{\frac{r}{2\pi}} \begin{Bmatrix} \cos(\theta/2)[\kappa - 1 + 2\sin^2(\theta/2)] \\ \sin(\theta/2)[\kappa + 1 - 2\cos^2(\theta/2)] \end{Bmatrix} \quad (3.38)$$

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  $\theta$ . 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

$$\lim_{r \rightarrow 0} \frac{2\mu}{\kappa + 1} \sqrt{\frac{2\pi}{r}} u_2 \quad (3.39)$$

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 al extrapolar linealmente estos valores hasta  $r=0$ .

También existen expresiones que evalúan directamente sin extrapolar, tomando valores muy cercanos a la punta de la grieta. Estas expresiones varían dependiendo si se utilizan elementos especiales o elementos normales. En el caso de los elementos especiales (figura 3.21), la expresión es

$$K_I = \frac{2\mu}{\kappa + 1} \sqrt{\frac{2\pi}{l}} (4v_{mid} - v_{ext}) \quad (3.40)$$



donde  $v_{mid}$  es el desplazamiento según  $y$  en el nodo medio desplazado, y  $v_{ext}$  es el desplazamiento en el nodo opuesto a la punta según  $y$ , como muestra la figura 3.21. En forma equivalente, para  $K_{II}$

$$K_{II} = \frac{2\mu}{\kappa+1} \sqrt{\frac{2\pi}{l}} (4u_{mid} - u_{ext}) \quad (3.41)$$

donde  $u$  indica desplazamiento en la dirección  $x$ . En el caso de utilizar elementos normales, las expresiones son

$$K_I = \frac{2\mu}{\kappa+1} \sqrt{\frac{2\pi}{l}} v_{ext} \quad (3.42)$$

$$K_{II} = \frac{2\mu}{\kappa+1} \sqrt{\frac{2\pi}{l}} u_{ext} \quad (3.43)$$

Estas expresiones son más prácticas de implementar en un software automatizado, pues se evita la interpolación que requiere muchos valores nodales

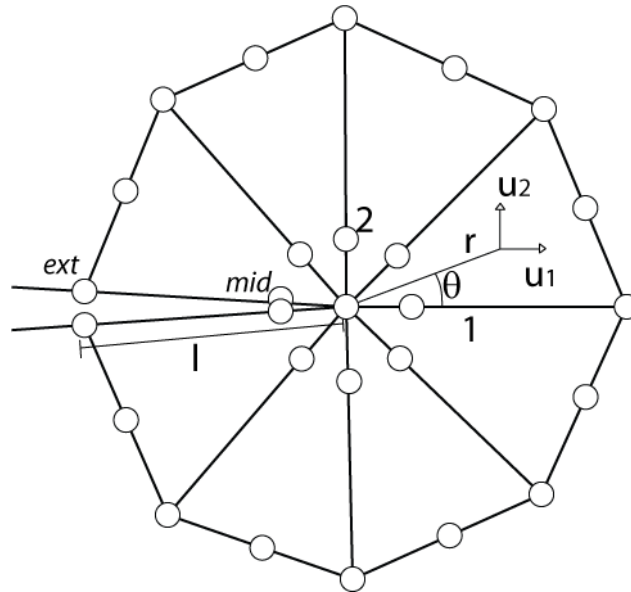


Figura 3.21: Elementos especiales utilizados para utilizar el método de extrapolación de desplazamientos. En los elementos especiales, el nodo medio (mid) es desplazado a una distancia  $1/4l$  de la punta de la grieta.

### 3.3.3 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

$$J = \int_{\Gamma} \left( U dy - t \frac{\partial d}{\partial x} ds \right) \quad (3.44)$$

donde  $\Gamma$  es un camino que parte en la superficie inferior de la grieta, y llega a la superior, encerrando a ésta, como muestra la figura 2.5;  $U$  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  $G = d\Pi/dA$ , donde A es el área de la superficie de la grieta (Kanninen & Popelar, 1985). Por otro lado, esta expresión está relacionada con los factores de intensificación de esfuerzos, como se mostró en el capítulo 2.2.2, por lo tanto

$$J = G = \frac{K^2}{E} \quad (3.45)$$

para esfuerzo plano, y

$$J = G = \frac{K^2}{E}(1 - \nu^2) \quad (3.46)$$

para deformación plana. Si la condición de carga genera modos mixtos, el valor de G se debe descomponer en los distintos modos (ecuación 2.25).

Por lo tanto, en el caso de la MFEL, calculando numéricamente dicha integral se puede obtener el factor de intensificación de esfuerzos. En el caso de un problema elasto-plástico la integral J sigue siendo válida, pero la relación J=G no es válida.

En teoría cualquier camino que se elija entregará el mismo resultado. Normalmente esto no se cumple, debido al error propio de la integración numérica. Por esto lo más razonable es realizar el cálculo para varios caminos y luego obtener el promedio.

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). Además, el método de extrapolación de desplazamientos obtiene directamente los valores separados de  $K_I$  y  $K_{II}$  en caso de existir modos de carga combinados, mientras que con el método de la integral J es necesario un paso posterior.

### Integral de Dominio Equivalente (EDI)

Como se mencionó la integral J es una integral de línea. Raju & Shivakumar (1990) observaron que la integración numérica de un camino alrededor de la punta de la grieta genera una importante pérdida de precisión en el cálculo. Por esto, propone un método alternativo, llamado *Integral de Dominio Equivalente* (EDI, Equivalent Domain Integral). Utilizando el teorema de la divergencia sobre la integral de línea<sup>1</sup>, se obtiene una expresión para una integral de área

$$G = - \int_{A_I} P_{1j} \frac{\partial q_1}{\partial x_j} dA = \int_{A_I} \left[ \sigma_{ij} \frac{\partial u_i}{\partial x_1} - W \delta_{1j} \right] \frac{\partial q_1}{\partial x_j} dA \quad (3.47)$$

<sup>1</sup>Se puede ver el detalle de esta deducción en el artículo Raju & Shivakumar (1990)

donde  $A_I$  es el área de integración 3.22,  $P_{ij}$  es el tensor Energía-Momentum de Eshelby (Eshelby, 1970), definido como

$$P_{ij} = W \delta_{ij} - \sigma_{ij} \frac{\partial u_i}{\partial x_j} \quad (3.48)$$

con  $i, j, l$  los ejes del sistema de coordenadas, y  $q_1(x, y)$  es una función definida de tal forma que vale 1 en el contorno  $\Gamma$ , 0 en el contorno  $C_0$ , y en el interior de  $A_I$  es una función suave y arbitraria de  $x$  e  $y$  con valores en el rango desde cero hasta uno.

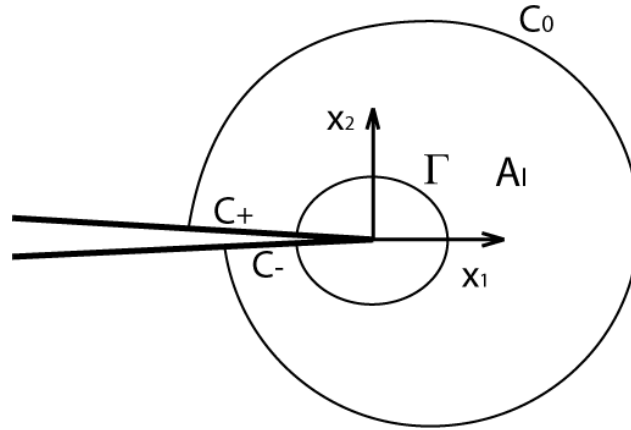


Figura 3.22: Contornos y área de integración definidos para utilizar el método de la integral de dominio equivalente.

Todo lo anterior puede extenderse a tres dimensiones, reemplazando la integral de área por una integral de volumen que encierra todo el frente de la grieta.

### Integral de Interacción

Recordando la ecuación 2.25, en caso de existir modos de carga combinados,  $J = G$  incluye el aporte de todos los modos, por lo tanto para obtener el valor de  $K_I$ ,  $K_{II}$  y  $K_{III}$  por separado, es necesario un cálculo extra para descomponer el valor de  $J$  en la contribución de cada modo. Uno de los métodos más utilizados es el llamado *Integral de Interacción* (Interaction Integral), propuesto por Shih & Asaro (1988). La tasa de liberación de energía  $G$  puede escribirse en forma matricial como

$$G = \frac{1}{8\pi} K_\alpha B_{\alpha\beta}^{-1} K_\beta \quad (3.49)$$

con  $\alpha, \beta \in \{1, 2, 3\}$ , corresponden a los modos  $\{I, II, III\}$  y  $B$  es una matriz que en el caso de un material isotrópico es diagonal, con componentes

$$B_{11} = B_{22} = \frac{\mu}{4\pi(1-\nu)}, B_{33} = \frac{\mu}{4\pi} \quad (3.50)$$

recuperándose la ecuación 2.25.

Por ejemplo, para calcular  $K_I$ , puede escribirse

$$G = \frac{1}{8\pi} [K_1 B_{11}^{-1} K_1 + 2K_1 B_{12}^{-1} K_2 + 2K_1 B_{13}^{-1} K_3 + \dots (*)] \quad (3.51)$$

donde (\*) son los otros términos que no incluyen a  $K_1$ . Si un campo auxiliar en modo I de intensidad  $k_1$  es superpuesto al campo existente, la tasa de liberación de energía total resultante es

$$G_{tot} = \frac{1}{8\pi} [(K_1 + k_1) B_{11}^{-1} (K_1 + k_1) + 2(K_1 + k_1) B_{12}^{-1} K_2 + 2(K_1 + k_1) B_{13}^{-1} K_3 + \dots (**)] \quad (3.52)$$

donde (\*\*) son los términos que no incluyen ni  $K_1$  ni  $k_1$ . La tasa de liberación de energía asociada al campo auxiliar es

$$G_{aux} = \frac{1}{8\pi} k_1 B_{11}^{-1} k_1 \quad (3.53)$$

Se define fuerza de extensión de interacción como

$$G_{int} = G_{tot} - G - G_{aux} = \frac{1}{8\pi} [2k_1 B_{11}^{-1} K_1 + 2k_1 B_{12}^{-1} K_2 + 2k_1 B_{13}^{-1} K_3] \quad (3.54)$$

$G_{tot}$ ,  $G$  y  $G_{aux}$  pueden ser calculadas utilizando el método de la integral de dominio equivalente. Con esto se obtiene un sistema de ecuaciones de la forma

$$G_{int}^{(\alpha)} = \frac{1}{4\pi} B_{\alpha s}^{-1} K_s \quad (3.55)$$

sin sumar sobre  $\alpha$ . Si se define para cada  $\alpha$ ,  $k_\alpha = 1$ , la solución para los factores de intensificación de esfuerzos es

$$K_\alpha = 4\pi B_{\alpha s} G_{int}^{(s)} \quad (3.56)$$

Así se obtiene cada factor de intensificación de esfuerzos por separado.

En este capítulo se han mostrado los aspectos relevantes para el desarrollo de esta tesis en lo que respecta a la metodología numérica aplicada a la mecánica de la fractura. En particular se muestran aspectos esenciales del método de elementos finitos en 2D; la generación de mallas geométricas para elementos finitos, con énfasis en el refinamiento y mejoramiento aplicado a la fractura; y finalmente técnicas de cálculo de los factores de intensificación de esfuerzos. Hasta este punto se ha mostrado todo el marco teórico necesario para desarrollar la tesis. A continuación se describirá el software desarrollado.

## Capítulo 4

# Software

### 4.1 Introducción

En este capítulo se describirá el software desarrollado y utilizado en esta tesis. El desarrollo se hizo a partir de un trabajo previo, realizado en el Departamento de Ciencias de la Computación de la Universidad de Chile (Calderón & Rivara, 2004), dentro de la línea de investigación en metodologías de generación y refinamiento de mallas. A partir de dicho trabajo, el autor de esta tesis creó un software cuyo objetivo final es simular la propagación de grietas en 2D utilizando el método de elementos finitos.

Para esto, primero el software permite generar modelos de elementos finitos en 2D, que son luego resueltos por medio del software comercial ABAQUS. Este software comercial tiene la capacidad de interactuar fácilmente con software externo, pues puede crear modelos de elementos finitos a partir de un archivo de entrada en formato texto plano, cuyas especificaciones están disponibles en su documentación. Por otro lado, ABAQUS posee una API<sup>1</sup> que permite crear código propio en los lenguajes de programación python o C++ y obtener información directamente de las estructuras de datos de ABAQUS. Esto facilita la creación de software de pre-proceso y post-proceso.

Utilizando estas funcionalidades se implementó un post-proceso el que a partir de los resultados del modelo de elementos finitos calcula los factores de intensificación de esfuerzos y el ángulo de propagación de la grieta. El software tiene implementado los algoritmos de generación, refinamiento y mejoramiento de mallas Lepp-Delaunay mencionados en el capítulo 3.2, que se utilizan para generar mallas iniciales y actualizar el modelo de elementos finitos entre un paso y otro.

### 4.2 Descripción general

Calderón & Rivara (2004) desarrollaron un software para la generación de mallas de triángulos 2D. Este software fue desarrollado en C++ para Windows y para la interfaz gráfica se utilizó la librería Qt. El software permite generar una malla de triángulos de una geometría poligonal con orificios poligonales. Se incluyen también funcionalidades para obtener estadísticas de la triangulación, como cantidad de puntos, calidad de

---

<sup>1</sup>API:Application Programming interface. Es una interfaz de código fuente que un sistema computacional provee para soportar requerimientos de servicios desde otro programa.

la malla, etc. La característica principal de este software es que tiene implementado 4 variaciones de algoritmos de refinamiento basados en el LEPP, que pueden ser usados en forma manual seleccionando triángulos objetivo, o refinando toda la malla según un criterio de calidad. El software además permite guardar la información de la malla en un archivo de texto, que puede ser cargado posteriormente en el mismo software. Mayores detalles sobre este trabajo se encuentran en la memoria de título de su autor (Calderón, 2004).

Para esta tesis se realizaron varias modificaciones a este software, con el fin de transformarlo en una herramienta orientada a la creación de modelos para el método de elementos finitos, y en particular a la resolución de problemas de la mecánica de la fractura. Lo primero fue modificar la estructura de datos de la malla para incluir información relevante para el método, como numeración de nodos y elementos e información del material a modelar. Además se implementaron funciones para agregar condiciones de desplazamiento y de carga, así como crear sendas estructuras de datos que permitan guardar esta información. En lo que respecta a los problemas de mecánica de la fractura, se implementó la posibilidad de transformar una o varias aristas internas en grietas, y en complemento con esto, se implementaron funciones específicas para refinar en la punta de una grieta, así como generar elementos especiales para éstas, y una serie de funciones para realizar el post-proceso de los resultados del software ABAQUS. Finalmente, se modificó la forma en la que se guarda la información en el archivo de salida, así como una función extra para generar la misma información según el formato requerido por ABAQUS. En la figura 4.1 se muestra una vista general del software

### 4.3 Implementación

El software fue implementado utilizando programación orientada a objetos. Para el desarrollo del código del software se utilizó el lenguaje C++, con la plataforma de desarrollo Visual Studio C++ para Windows. Para la interfaz gráfica, se utilizó la librería Qt 3.3.2. Y el software Qt Designer. Para el post-proceso se utilizó el lenguaje Python, utilizando la API de ABAQUS para acceder a sus estructuras de datos.

**Ventana Principal** La clase principal es la ventana principal (clase *miigwindow*), en donde se inicializan todos los objetos relevantes para la representación del modelo. Los objetos a iniciar son las distintas ventanas (Widgets) del programa, así como la zona de dibujo de OpenGL (QGIWidget). Dentro del objeto WinES, que corresponde a la ventana de inserción de puntos, existe el objeto EstructuraGeometrica, de la cual se hereda la malla de triángulos. Es decir, al crearse el objeto *miigwindow*, se crea, entre otros, el objeto WinES, y al crearse este objeto, se crea el objeto MallaTriangulos. Cada una de las distintas ventanas tiene sus funciones propias, y punteros a otras ventanas por si necesita acceder a sus funciones o modificar sus objetos. Por ejemplo la ventana de información (clase *WinIF*), que sirve para obtener distinta información de la malla (número de aristas, triángulos, calidad de los elementos, etc) tiene una lista de punteros a triángulos, una lista de punteros a aristas y una lista de punteros a puntos, que se utilizan para guardar temporalmente las estructuras seleccionadas durante la ejecución de la aplicación, por medio de las herramientas de selección, función que es fundamental para poner en marcha los algoritmos de refinamiento basados en criterios. Como este objeto puede modificar la malla de triángulos, tiene un puntero al objeto WinES, y a través de esta clase puede acceder a la malla.

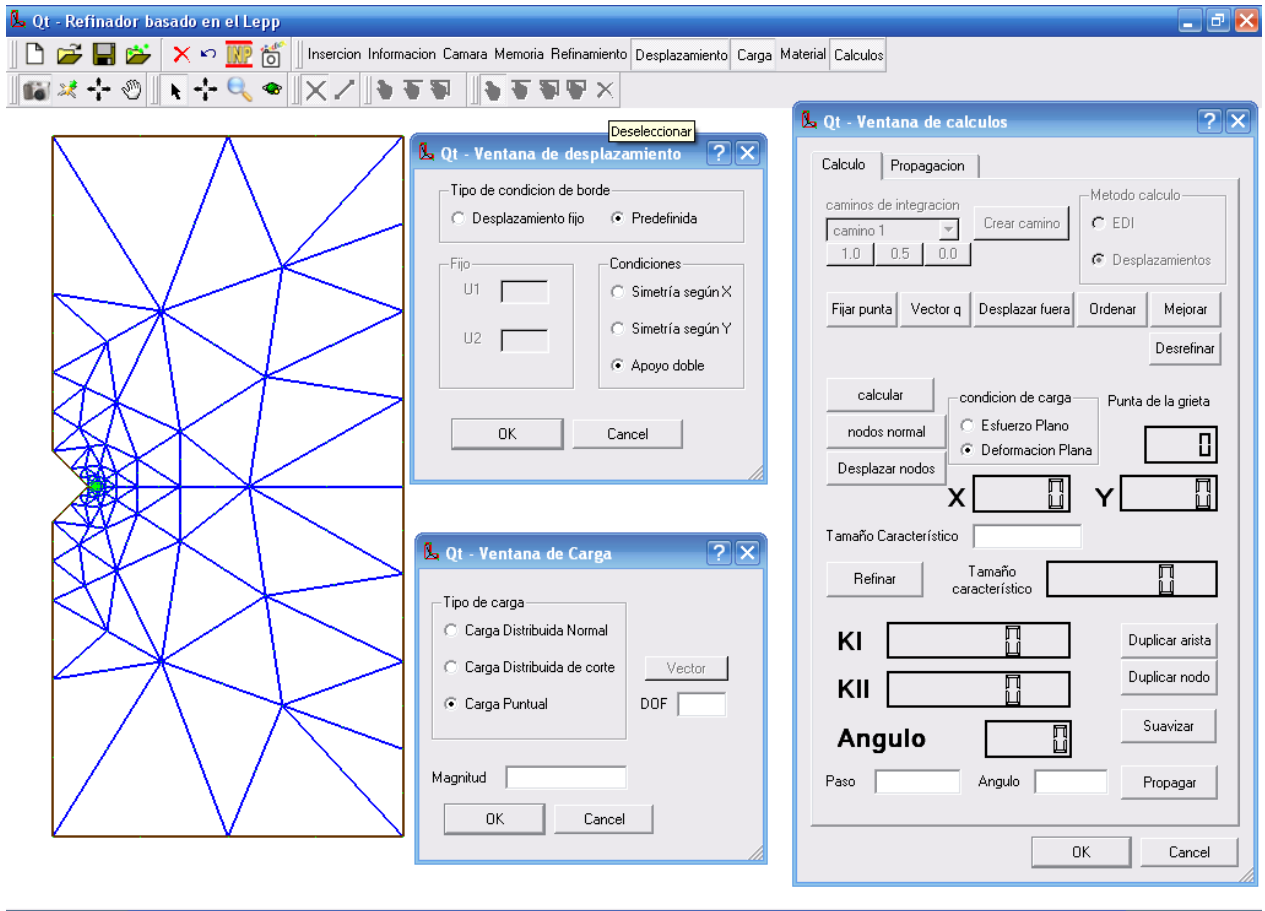


Figura 4.1: Vista general del software desarrollado.

**Malla de Triángulos** La malla de triángulos (objeto *MallaTriangulos*) hereda a la clase *EstructuraGeometrica*. Se representa por medio de un puntero a uno de los Triángulos, y por tres arreglos con los punteros a cada uno de los objetos (nodos, aristas y triángulos) de la malla. Estos arreglos son estructuras auxiliares que facilitan algunas funciones, como por ejemplo la lectura y escritura en archivos de texto. La estructura de datos utilizada para los elementos se observa en la figura 4.2. La estructura base es el triángulo. Éste guarda información de sus vértices y de sus aristas. Las aristas guardan información de sus triángulos vecinos y de sus tres nodos, los dos de los extremos y el nodo medio. Por lo tanto, los triángulos se conectan a sus vecinos a través de las aristas. Esta estructura fue propuesta por Calderón (2004) para optimizar el funcionamiento de los algoritmos de refinamiento. En la estructura original de Calderón el objeto *AristaMalla* no tenía un nodo medio. La modificación hecha por el autor al código original de la estructura de datos de la malla, a grandes rasgos, consistió en agregar índices correlativos a los nodos y los elementos, y modificar la estructura de datos de la arista para incluir el nodo medio, dado que se utilizan elementos finitos de 6 nodos. Los vértices están orientados en sentido contrario a las agujas del reloj, y los nodos están numerados como muestra la figura 3.2 (c).

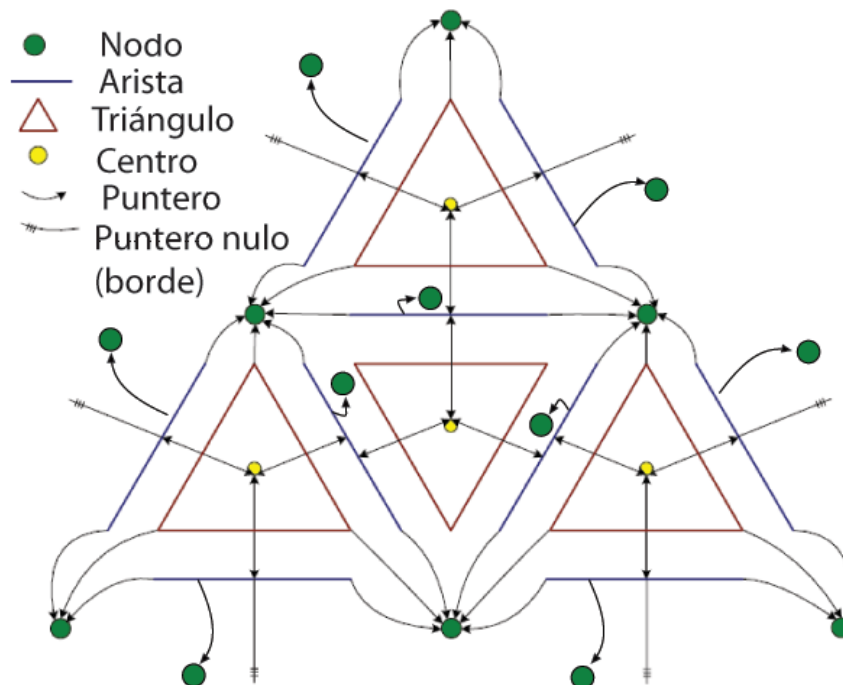


Figura 4.2: Estructura de datos utilizada para representar la malla de triángulos.

**Algoritmos de Refinamiento** La parte más importante del desarrollo de Calderón es la implementación de los algoritmos de refinamiento basados en el LEPP. Para refinar la malla, los algoritmos de refinamiento son objetos utilizados por otro objeto, que es el que desea refinar. Por ejemplo, el objeto *WinRF* (ventana de refinamiento) posee un objeto *AlgoritmoRefinamiento* y un objeto *AlgoritmoMejoramiento*. Entonces si una función de *WinRF* desea refinar un triángulo de la malla, ésta le entrega un puntero de este triángulo al objeto



*AlgoritmoRefinamiento* y este lo refina. no se hicieron modificaciones a estos algoritmos.

**Objetos para elementos finitos** Además de la modificación de la estructura de datos de la malla ya mencionada, se agregaron nuevos objetos ventana al programa, con sus respectivas funciones, y se crearon nuevas estructuras de datos para guardar información relevante para el modelo de elementos finitos. La información del modelo de elementos finitos se guarda en un objeto llamado *ModeloEF*. La clase *modeloEF* incluye tres estructuras de datos, una para guardar información del material (clase *Material*), una para guardar la información de las condiciones de desplazamiento (clase *condDesp*) y otra para guardar las condiciones de carga (clase *condCarga*). La clase *modeloEF* se inicializa al arrancar la aplicación y crearse la ventana principal.

La clase *material* permite guardar información de un sólo material para todo el modelo, y está limitado a materiales elásticos e isotrópicos, por lo que sólo guarda la información del nombre del material, su módulo de elasticidad y su Módulo de Poisson.

La clase *condDesp* se implementa como una lista de puntos que son los nodos en los que se fija la restricción, y un “string” con el tipo de restricción (empotrado, apoyo simple, etc) en caso de ser una restricción común, o como un par (u1, u2) para fijar el desplazamiento según x o y en forma más general. El objeto *ModeloEF* tiene un arreglo de objetos *condDesp*, para poder incluir más de una condición de desplazamiento en el modelo.

La clase *condCarga* permite guardar cargas homogéneas normales o de corte en una superficie, y cargas puntuales. El objeto *condDesp* posee un número *double* que guarda la magnitud de la carga y tres listas de índices de triángulos, s1 s2 y s3. Al fijar una carga sobre una superficie, se determinan los triángulos involucrados en dicha carga. Luego en el objeto *condDesp*, en la lista s1 se guardan los triángulos en los cuales la carga actúa sobre su arista 1 (entre el 1er y segundo nodo, según la notación de la figura 3.2); en la lista s2 se guardan los triángulos en los cuales la carga actúa sobre su arista 2 (entre el 2do y 3er nodo), y en la lista s3 los triángulos en los cuales la carga actúa sobre su arista 3 (entre el 3er y 1er elemento). Este método es el usado por el software ABAQUS en sus archivos de entrada y salida. Lo anterior es válido para las cargas normales y de corte. En el caso de una carga normal, un signo negativo en la magnitud de la carga indica una carga de tracción. En el caso de una carga de corte, además de la magnitud se guarda un objeto *segmento* para representar el vector con la dirección de la aplicación de la carga. Para la carga puntual se incluye una variable para guardar el grado de libertad en el que se aplica la carga. El objeto *ModeloEF* tiene un arreglo de objetos *condCarga* para incluir más de una condición de carga, igual que en el caso de los objetos *condDesp*.

**Duplicación de nodos y aristas** Otra cosa nueva que se implementó en esta versión del software fue un grupo de métodos que permiten modificar la estructura de datos para duplicar un nodo o una arista. Duplicar significa crear una copia del objeto, diferente del original topológicamente, pero con la misma posición geométrica.

Para duplicar una arista, es necesario identificar el par de triángulos que la comparten. Luego se crea una nueva arista que se asocia a uno de los triángulos, desconectando éste de la arista vieja. El otro triángulo entonces se queda con la arista original, que ahora pasa a ser orilla. La arista nueva también pasa a ser orilla. El proceso se muestra esquemáticamente en la figura 4.3

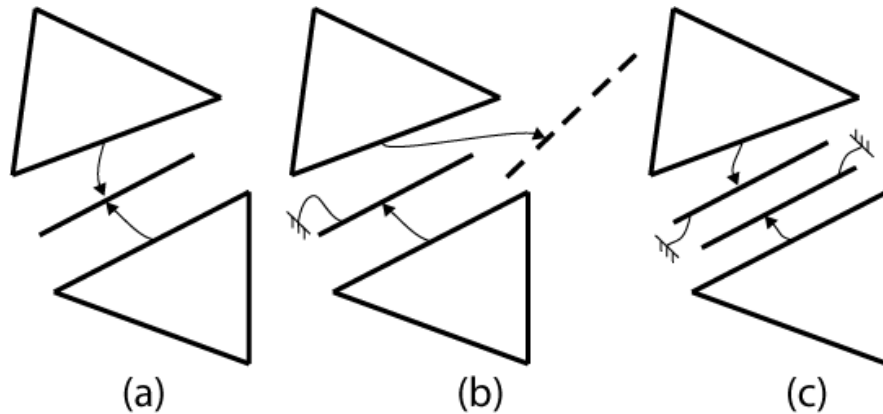


Figura 4.3: Esquema de la modificación de la estructura de datos de la malla para la duplicación de una arista. (a) Arista sin duplicar. (b) Creación de un nuevo objeto arista. (c) Ambas aristas se transforman en orilla y se hacen independientes.

En el caso de la duplicación de un nodo, se deben identificar todos los triángulos y las aristas que lo comparten. La duplicación de un nodo se hace normalmente cuando una arista toca la orilla o para crear una propagación. En estos casos, el nodo separa dos grupos de triángulos como se muestra en la figura 4.4(a). Para duplicar el nodo, se deben identificar estos dos grupos, elegir uno de ellos, crear un nuevo objeto nodo y asociar los punteros de uno de los grupos a este nodo, como se muestra en la figura 4.4(b).

**Estructura Geométrica y Formato IGES** La última modificación hecha al programa consistió en implementar métodos que permitiesen cargar un modelo geométrico 2D en formato IGES, que es uno de los formatos estándar para guardar información CAD más utilizados, así como el más sencillo de implementar. En conjunto con los métodos necesarios para la lectura de este tipo de archivos, se crearon nuevas estructuras de datos para guardar las entidades geométricas cargadas, que pueden ser líneas rectas, curvas, arcos, círculos, etc.<sup>2</sup>

Para guardar la información geométrica se creó la clase *IgesModel* y la interfaz *IgesObject*. La primera guarda el modelo geométrico completo, representado como una lista de objetos que corresponden a las entidades geométricas. Los objetos que representan a estas entidades son implementaciones de la interfaz *IgesObject*. Los métodos que leen o modifican entidades geométricas no pueden hacerlo directamente sino que deben hacerlo a través de los métodos de la clase *IgesModel*. Las implementaciones existentes de la interfaz *IgesObject* son *arc* y *line*, que corresponden a un arco de círculo y a una línea recta, respectivamente. Un círculo se representa como dos arcos conectados entre sí, y no tiene una implementación propia.

Cada entidad geométrica tiene asociada un ID único y correlativo, y una variable *double* correspondiente al “seed”, que consiste en el espaciamiento inicial entre los nodos que se insertarán en esta entidad al momento de crear la malla. Esta variable permite generar una malla inicial con tamaño característico variable según la zona. Por ejemplo, el espaciado inicial en un arco debe ser mayor que en una línea recta.

<sup>2</sup>La mayor parte de este trabajo fue realizado por Eduardo Graells, como parte del curso “Trabajo Dirigido”, a cargo de la profesora María Cecilia Rivara.

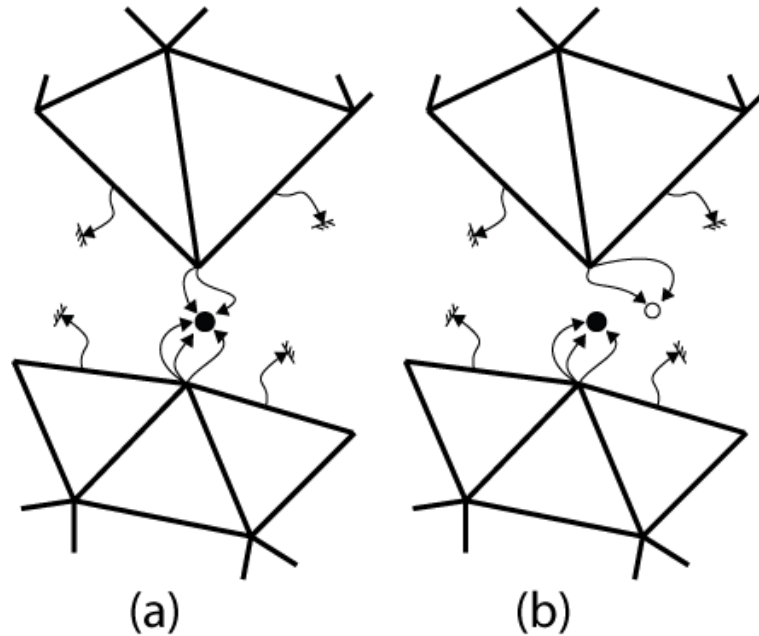


Figura 4.4: Duplicación de un nodo. (a) El nodo separa dos grupos de triángulos que apuntan a él, debido a que la duplicación ocurre después de la duplicación de una arista. (b) Se elige uno de los grupos y se asocia al nuevo nodo, quedando el nodo original con el otro grupo.

## 4.4 Funcionamiento

### 4.4.1 Detalle de las funciones

En la figura 4.5 pueden apreciarse las distintas barras de herramientas con sus respectivos botones<sup>3</sup>. A continuación se muestra un detalle de cada uno de los botones y Widgets relacionados con estas barras. Las funciones marcadas con un (\*) fueron implementadas en esta tesis.

- **Barra Principal** Los botones de esta barra permiten acceder a las funciones básicas para abrir y guardar archivos. Las funciones disponibles son
  1. **Nuevo** Crea una nueva malla geométrica, es decir, una pantalla en blanco.
  2. **Abrir** Abre un archivo para crear un nuevo modelo. Los tipos de archivo posibles son ‘mtr’ (malla de triángulos), ‘igs’(\*) (geometría formato IGES) y ‘pts’ (lista con coordenadas de puntos).
  3. **Guardar** Guarda la malla geométrica actual. Ésta se guarda en el formato ‘mtr’
  4. **Generar Malla** Genera una malla a partir de la lista de puntos (formato ‘pts’) cargada anteriormente con el botón **Abrir**
  5. **Borrar** Borra las estructuras geométricas (*no implementado*)

<sup>3</sup>Algunas de las opciones mostradas no están implementadas desde el desarrollo del software original de Calderón & Rivara (2004), y tampoco fueron implementadas en esta tesis porque no eran necesarias para su desarrollo.

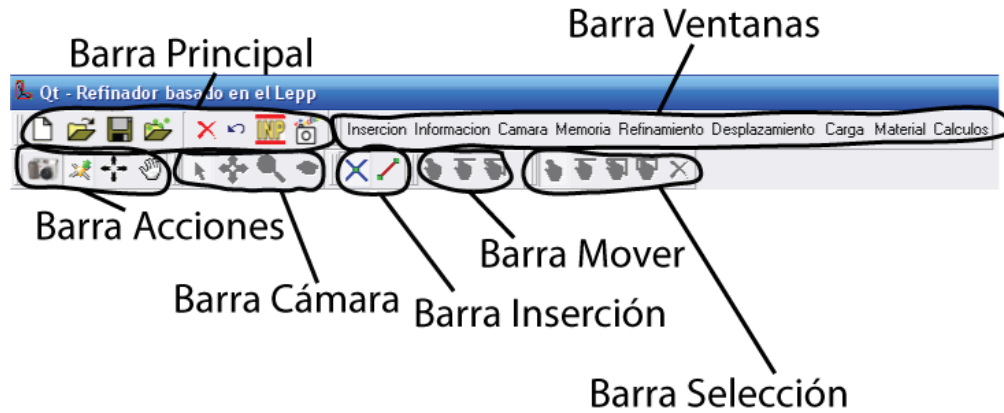


Figura 4.5: Detalle de las distintas barras de herramientas del software.

6. **Deshacer** Deshace la última acción (*no implementado*)
  7. **Generar INP(\*)** A partir de la malla geométrica, la información del material y las condiciones de borde actuales en el software, genera un archivo en formato 'inp' que puede ser cargado en el software ABAQUS para ser resuelto.
  8. **Guardar Imagen(\*)** Realiza una captura de lo que aparece actualmente en la pantalla y lo guarda en un archivo en formato 'jpg'.
- **Barra Ventanas** Los botones de esta barra de herramientas activan y desactivan los distintos Widgets con las funciones para operar sobre las estructuras de datos. Éstas son
    1. **Inserción** Esta ventana se utiliza para insertar manualmente puntos y aristas restringidas sobre una malla ya existente o creando una que está dada por el conjunto convexo de los puntos insertados. Puede elegirse entre una inserción del tipo Delaunay (que conserva la propiedad de los círculos circunscritos) o no Delaunay.
    2. **Información** Esta ventana posee bastantes funciones. Está dividida en cuatro 'tabs', dependiendo si las funciones operan sobre aristas, triángulos, mallas o sobre la geometría, que se describen a continuación
      - (a) **Arista** A la izquierda se muestra la cantidad de aristas seleccionadas y los puntos inicial y final. A la derecha existen cuatro funciones que operan cuando una sola arista está seleccionada, con la acción **Selección** y la sub-acción **Arista** seleccionadas. El display de la derecha muestra el largo de la arista seleccionada.
      - (b) **Triángulo** Muestra estadísticas generales del triángulo seleccionado.
      - (c) **Malla** A la izquierda se muestran estadísticas generales de la malla completa. A la derecha se muestran varios criterios de selección de triángulos.  
Si la acción **Selección** y la sub-acción **Triángulo** están seleccionadas, definiendo uno de los criterios mostrados y luego apretando 'seleccionar', todos los triángulos que cumplan con el criterio se mostrarán seleccionados en pantalla. Hasta ahora sólo se encuentra implementado

el criterio de ángulo mínimo. Si la acción **Selección** y la sub-acción **Arista** están seleccionadas, y marcando el “checkbox” de lado menor, todas las aristas cuyo largo sea mayor que el valor seleccionado serán marcadas. Este criterio se utiliza en la metodología de eliminación de triángulos malos que se muestra en el capítulo 5.2.3.

Además hay un botón ‘histograma(\*)’ que genera dos archivos de texto, uno llamado ‘histograma.txt’ y otro llamado ‘listahistograma.txt’, y que son una lista con el indicador de calidad para cada uno de los triángulos de la malla. El indicador usado es el dado por la ecuación 3.26. Este archivo puede ser luego procesado, por ejemplo con el software MATLAB, para generar un histograma de calidad como el mostrado por ejemplo en la figura 3.14.

- (d) **Geometría(\*)** Estas funciones operan sobre las entidades geométricas cargadas desde un modelo IGES.

El primer cuadro de texto tiene la función de seleccionar una de las entidades geométricas, entregando como parámetro el ID de ésta. Una vez seleccionada la entidad (el ID de la entidad actualmente seleccionada se muestra en el display de la derecha), el segundo cuadro de texto permite modificar la variable “seed” de cada entidad geométrica. Al apretar ‘pre-visualizar’, se muestra en pantalla los lugares donde se insertarían los nodos al generar la malla.

El botón ‘ordenar índices’ regenera los índices de los nodos y los elementos, en caso de que durante la utilización del programa se hayan borrado algunos elementos, perdiendo la correlatividad de los índices. Esta función es útil cuando se eliminan los triángulos externos en la malla de una geometría cóncava.

El botón ‘generar malla inicial’ inserta nodos en todas las posiciones dadas por los distintos ‘seeds’ de las entidades geométricas, generándose una malla inicial dada por el conjunto convexo de estos nodos. En caso de que la geometría a mallar sea cóncava, deben borrarse los elementos que quedan fuera de la geometría, para eso se utiliza la sub-acción **Deseleccionar/Borrar** disponible en la barra **Selección**.

3. **Cámara** Permite seleccionar una vista 2D ó 3D, lo que no tiene utilidad hasta el momento pues no se han implementado modelos 3D.
4. **Memoria** Muestra la memoria pedida y devuelta (*no implementado*)
5. **Refinamiento** Permite refinar uno o un grupo de triángulos, y seleccionar la variante del algoritmo de refinamiento basado en el LEPP. Existen 4 implementaciones en el software.

Esta ventana permite efectuar el refinamiento de tres formas: la primera, mostrando paso a paso el proceso del refinamiento, similar a lo mostrado en la figura 3.12; la segunda, refinando un triángulo seleccionado (si hay más de uno, por cada vez que se apriete el botón ‘refinar’ se refinará uno); y la tercera, refinando toda la malla. Este último caso funciona sólo cuando se ha cargado una geometría IGES, y no existía en el software original. Lo que hace esta función es refinar sucesivamente todos los triángulos que no cumplan con el criterio de calidad definido en la ventana **Información**, en el tab **Malla**, y además los triángulos cuyo lado mayor sea mayor que el mayor de los valores de ‘seed’ de las entidades geométricas del modelo. El valor del ‘seed’

se fija en la ventana **Información**, en el tab **Geometría**.

6. **Desplazamiento(\*)** Esta ventana permite fijar las condiciones de desplazamiento sobre un borde de la geometría. Para funcionar debe estar activada la acción **Selección** y la sub-acción **Arista** o la sub-acción **Punto**, pues las condiciones de desplazamiento se fijan sobre un conjunto de aristas, o sobre un conjunto de puntos. Una vez seleccionado el conjunto de aristas o puntos, se puede fijar el desplazamiento en los cuadros de texto de la izquierda, o sino fijar una condición de desplazamiento predefinida en la derecha, como simetría o apoyo doble.
7. **Carga(\*)** Esta ventana permite fijar las condiciones de carga sobre un borde de la geometría. Las condiciones de carga normal o de corte se fijan sobre un conjunto de aristas. Una carga puntual se fija sobre un sólo punto. Para el caso de la carga tangente se debe definir un vector de dirección. Para eso, debe estar activada la acción **Selección** y la sub-acción **Punto**, se seleccionan dos puntos y se aprieta el botón 'vector'. El vector tiene su origen en el primer punto seleccionado, y la punta en el segundo punto seleccionado. En el caso de fijar una carga puntual, en la ventana 'DOF' se fija el grado de libertad sobre el que actuará la carga puntual. Finalmente en la ventana 'magnitud' se define la magnitud de la carga, cualquiera sea el tipo.
8. **Material(\*)** En esta ventana se definen las propiedades del material a modelar. Como sólo pueden modelarse materiales elásticos isotrópicos, los parámetros son el módulo de Young, el módulo de Poisson, y el nombre del material.
9. **Cálculos(\*)** En esta ventana se prepara y realiza la interacción con el software ABAQUS. La ventana está hecha para calcular con el método de la integral J y la extrapolación de desplazamientos, pero el método de la integral J no está implementado. Los botones están distribuidos de manera de realizar paso a paso el proceso de preparación del modelo para resolverlo. Esta ventana posee dos 'tabs', uno para el cálculo de un problema simple (un paso), y otro para simular iterativamente la propagación
  - (a) **Tab Cálculos** Primero, con la acción **Selección** y la sub-acción **Punto** activadas, se selecciona el nodo que corresponde a la punta de la grieta y se aprieta el botón 'fijar punta', para identificarla. En los displays de la derecha de la ventana aparece el índice del nodo fijado, así como sus coordenadas.

En el campo de texto donde dice 'tamaño característico', se fija el tamaño que tendrán los elementos en la punta de la grieta. Se fija un valor, y al apretar 'refinar' los elementos de la punta de la grieta se refinan sucesivamente hasta el tamaño característico fijado. Luego de refinar se debe fijar de nuevo la punta de la grieta.

Luego, para fijar el vector de dirección de la propagación (que debe ser un vector que apunta en la misma dirección que la grieta), seleccionando sólo dos puntos, se aprieta el botón 'vector q'. Para hacer esto, necesariamente se debe haber fijado ya la punta de la grieta.

Con el botón 'Desplazar fuera' el software mueve los nodos que están dentro del radio que tendrá la roseta (definido por el largo de la arista que toca la cara de la grieta), para el posterior ordenamiento de ésta.

Con el botón ‘ordenar’ se genera una roseta simétrica donde el valor de  $l$  corresponde al largo de la arista que toca la grieta, desplazando o insertando nodos, según sea necesario.

Posterior a la generación de la roseta, se obtienen elementos colindantes de mala calidad. Para volver a tener una malla de buena calidad se debe apretar el botón ‘mejorar’, que aplica el algoritmo Lepp-Delaunay sobre todos los elementos de mala calidad, según el criterio definido en la ventana **Refinamiento**.

Luego pueden eliminarse triángulos malos con el botón “desrefinar”, que utiliza la metodología mostrada en el capítulo 5.2.3.

Posterior al mejoramiento, el botón ‘suavizar’ realiza un suavizamiento laplaciano sobre toda la malla, obteniendo la malla definitiva para los cálculos.

Luego se debe elegir si el problema es de deformación plana o esfuerzo plano, y si se desea utilizar elementos especiales con los nodos desplazados o elementos normales.

Una vez fijado todo esto, al apretar el botón ‘calcular’ internamente el software genera un archivo ‘inp’ con el modelo, lo resuelve con el software ABAQUS, y calcula los factores de intensificación de esfuerzos y el ángulo de propagación, que son mostrados en los displays de más abajo.

Una vez obtenidos los resultados del post-proceso, se puede realizar un paso de la propagación. Para esto, se define un ‘paso’, que será el largo del nuevo segmento de grieta y se fija manualmente el ángulo de propagación en la ventana ‘angulo’. Luego al apretar el botón ‘propagar’, se genera un nuevo segmento de grieta del largo fijado, y en el ángulo fijado. Repitiendo iterativamente este proceso se simula la propagación de una grieta. Lo normal es que el paso de propagación sea del mismo orden de magnitud que el tamaño característico de la grieta, que se muestra en el display que dice ‘tamaño característico’.

También existe un botón para duplicar una arista y uno para duplicar un nodo, necesarios para crear la grieta inicial, o realizar el proceso en forma manual.

#### (b) **Tab Propagación**

En el cuadro de texto ‘nombre’ se escribe el nombre que tendrá el problema. Eso es para que cuando comience la propagación se generen archivos *nombre\_0.mtr*, *nombre\_0.inp*, *nombre\_1.mtr*, *nombre\_1.inp*, etc. Luego, con la acción **Selección** y la sub-acción **Punto Activadas**, se selecciona el punto que corresponde a la punta de la grieta y se aprieta el botón ‘fijar punta’. Si el checkbox ‘paso a paso’ está marcado, al apretar el botón ‘iniciar’ se resolverá sólo un paso de la propagación. Si se desactiva, en el cuadro de texto ‘cantidad iteraciones’ se fija la cantidad de iteraciones que se desea resolver. Luego al apretar ‘iniciar’ el programa comenzará a resolver iterativamente los pasos de la propagación, y al llegar al último mostrará la malla final.

- **Barra Acciones** En esta barra están los botones que determinan qué tipo de acción se realizará sobre el modelo. Las acciones posibles son

1. **Cámara** Activa las funciones (sub-acciones) para controlar el movimiento de la cámara (barra

**Cámara).**

2. **Inserción** Activa las funciones (sub-acciones) que permiten insertar puntos o aristas restringidas.
3. **Mover** Activa las funciones (sub-acciones) que permiten mover componentes topológicos de la malla (barra **Mover**, *no está implementado*)
4. **Seleccionar** Activa las funciones (sub-acciones) que permiten seleccionar los componentes topológicos (barra **Selección**). Estas funciones se utilizan en conjunto con las funciones de las ventanas, por ejemplo, para refinar un triángulo, primero se selecciona con una de estas funciones (quedando el o los triángulos seleccionados marcados con un color amarillo) y luego se aplica la función.

- **Barra Cámara** Se muestran las sub-acciones específicas para la acción de la cámara. Éstas son

1. **Global** Permite un control global de la cámara. Con la rueda (wheel) del “mouse” se controla el zoom de cámara, y manteniendo apretado el botón derecho del “mouse” y moviéndolo se puede mover la vista de lo que se muestra en pantalla.
2. **Mover** Permite mover la vista de lo que se muestra en pantalla.
3. **Zoom** Permite controlar el zoom de lo que se muestra en pantalla.
4. **Rotar** Permite rotar la vista de lo que se muestra en pantalla (disponible sólo si en la ventana **Cámara** se ha seleccionado ‘vista 3D’).

- **Barra Inserción** Sub-acciones específicas para la inserción. Éstas son 2, insertar un punto, o insertar una arista restringida.

- **Barra Mover** Sub-acciones específicas para mover la estructura (*no está implementado*).

- **Barra Selección** Indica qué entidad topológica será seleccionada al estar activada la acción Seleccionar. Las sub-acciones posibles son

1. **Punto** Permite seleccionar puntos de la malla. Para hacerlo se debe acercar el mouse a uno de los vértices de uno de los triángulos y apretar el botón izquierdo. Al estar seleccionado el punto cambiará a color amarillo. Pueden seguir seleccionándose más puntos de la misma manera. Para deseleccionar un punto se debe apretar nuevamente el botón izquierdo del mouse encima de él.
2. **Arista** Permite seleccionar aristas de la malla. Funciona de la misma manera que la selección de puntos.
3. **Triángulo** Permite seleccionar triángulos de la malla. Funciona de la misma manera que la selección de puntos.
4. **Geometría** Permite seleccionar una entidad geométrica obtenida de un modelo IGES. Funciona de la misma manera que la selección de puntos, pero la entidad geométrica seleccionada no queda marcada en color amarillo, y sólo puede seleccionarse una entidad a la vez.



5. **Deseleccionar/Borrar** Si está seleccionada la sub-acción **Punto** o **Arista**, se deseleccionan todos los elementos seleccionados. si está seleccionada la sub-acción **Triángulo**, se borran los triángulos seleccionados. Esta última función se utiliza para generar mallas de geometrías convexas, donde es necesario eliminar los triángulos externos.

#### 4.4.2 Pasos para crear y resolver un modelo

**Partida** El programa parte con una ventana gráfica en blanco. Para empezar existen cuatro opciones: 1.- crear una nueva malla insertando secuencialmente aristas y/o puntos (ventana **Inserción**, barra **Ventanas**); 2.- abrir un archivo en formato “mtr”, que es un archivo de texto con los datos de una malla geométrica (botón **Abrir**, barra **Principal**); 3.- abrir un archivo en formato “pts”, que es un archivo de texto con una lista de puntos a insertar (botón **Abrir**, barra **Principal**); y 4.- abrir un archivo en formato “igs”, que es un archivo de texto en formato IGES, con información geométrica, sin malla (botón **Abrir**, barra **Principal**).

En el caso 1 la inserción secuencial generará una triangulación del conjunto convexo de los nodos y aristas insertados. En el caso 2 la malla ya está lista. En el caso 3, al abrir el archivo ‘pts’ se pueden visualizar los nodos pero no la triangulación. Para generar la triangulación correspondiente, se debe apretar el botón **Generar Malla** de la barra **Principal**.

**Borrar triángulos** En los tres casos, luego de generada una malla, se pueden borrar triángulos para obtener la geometría deseada, como muestra la figura 4.6. Para esto se seleccionan los triángulos que se desean borrar y utiliza el botón **Deshacer/seleccionar** de la barra **Selección**. Después de borrar los triángulos, los índices correspondientes a estos desaparecerán: con el botón ‘reordenar índices’ de la ventana **Información**, tab **Geometría**, se regeneran los índices para que sean correlativos.

**Generar malla inicial a partir de seed (modelos IGES)** En el caso 4, el proceso es distinto. Una vez cargada la geometría en el software, se debe generar la malla inicial. Para esto se utiliza la ventana **Información**, tab **Geometría**.

El primer paso es fijar el ‘seed’ de cada una de las entidades geométricas del modelo. Para seleccionar una entidad, existen dos formas, una es a través de la acción **Seleccionar**, sub-acción **Geometría**, y la otra es por medio del campo de texto superior donde se ingresa el ID de una entidad, o el número 0 para seleccionarlas todas. Una vez seleccionada una o todas las entidades, se fija el ‘seed’ deseado en el campo de texto y se aprieta el botón ‘previsualizar’. El nuevo seed se mostrará en pantalla como muestra la figura 4.7.

Una vez que se fija el seed, con el botón ‘generar malla inicial’ se genera una malla insertando nodos en las posiciones determinadas por el seed. Con esto se forma una malla del conjunto convexo de los nodos, como muestra la figura 4.7.

Si la geometría a mallar es cóncava, entonces varios triángulos quedarán afuera de ésta por lo que deben borrarse y reordenar los índices de la misma forma que en los casos 1, 2 y 3, figura 4.6.

**Malla mejorada** Una vez creada una malla inicial , puede mejorarse o refinarse, usando la ventana **Refinamiento**. En la figura 4.8, a partir de la malla inicial obtenida desde un modelo IGES, se genera una malla de mejor calidad.

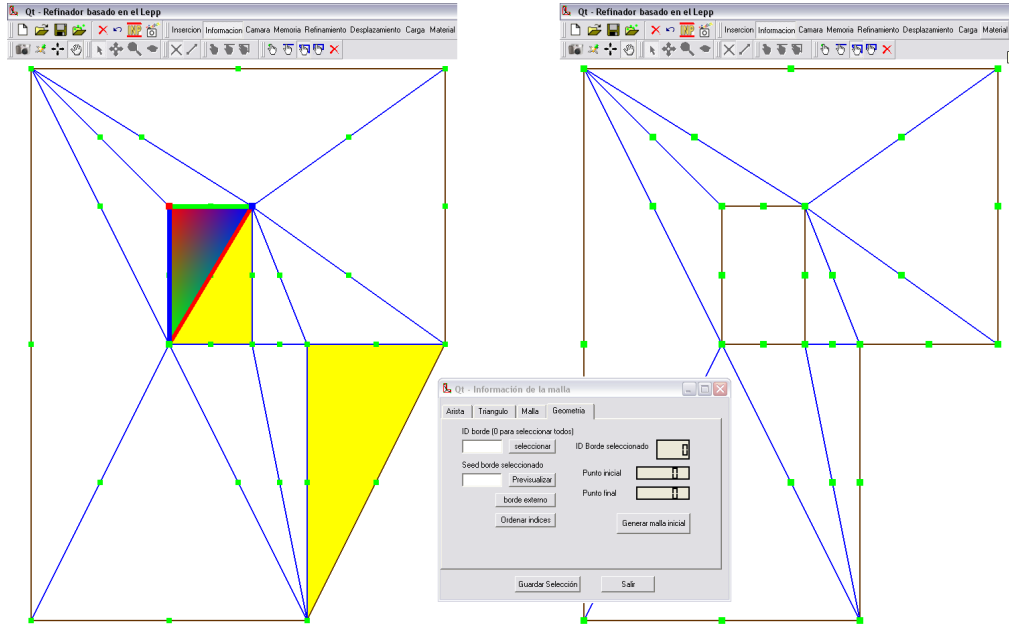


Figura 4.6: Una vez generada una malla, pueden borrarse triángulos para generar orificios o figuras convexas. Posterior a eso deben reordenarse los índices para que sean correlativos.

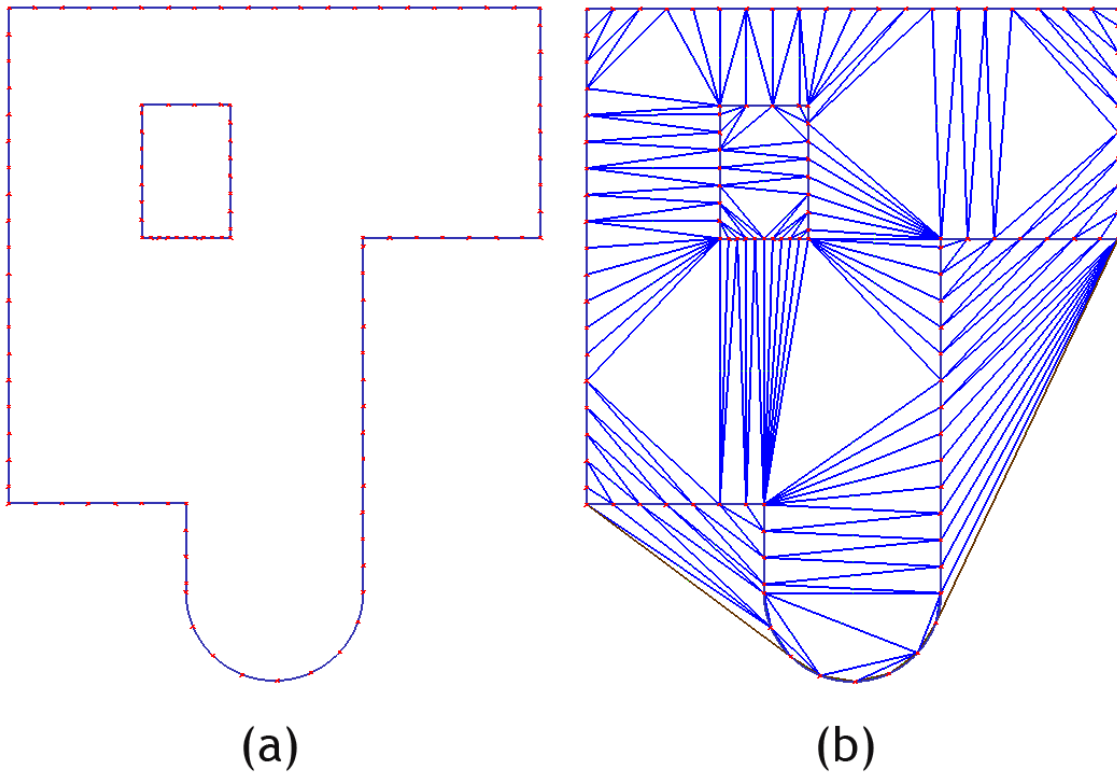


Figura 4.7: (a) Geometría obtenida desde un archivo IGES, con el seed definido para sus bordes. (b) malla inicial definida por pos puntos dados por el seed.

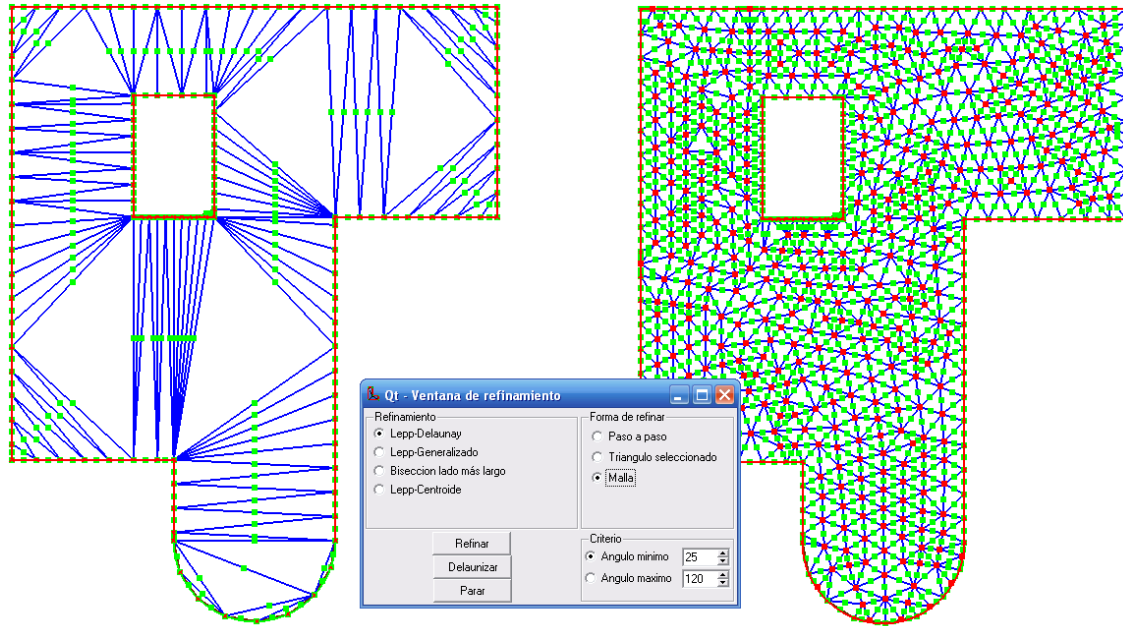


Figura 4.8: Generación de malla inicial. Primero se genera una malla Delaunay insertando puntos en el borde de la geometría. Luego se obtiene la malla final aplicando sucesivamente el algoritmo Lepp-Delaunay hasta que ningún triángulo tenga su lado mayor más largo que el seed mayor de la orilla.

**Insertar grieta** Después de definir la malla, se inserta una arista que correspondería a la grieta inicial. Esto se logra al insertar una arista restringida, un par de nodos inicial y final, o utilizando nodos existentes. Estas funciones están en la ventana **Inserción**.

La arista recién insertada debe seleccionarse para transformarse en una grieta, lo que significa que la arista es transformada en dos superficies libres. Para transformar una arista en grieta, se debe seleccionar la arista utilizando la herramienta de selección, y luego en la ventana de **Cálculos** se aprieta el botón ‘duplicar arista’. Si la arista a refinar llega hasta la orilla, debe duplicarse el nodo de intersección entre la arista y el borde, seleccionándolo y apretando el botón ‘duplicar nodo’.

**Roseta** Una vez lista la grieta, se debe refinar en la punta y crear la roseta simétrica. Todo este proceso se realiza con las funciones de la ventana **Cálculos**, y está descrito en el capítulo 4.4.1. Con esto la malla queda completamente lista para resolver el modelo, ahora deben fijarse los parámetros físicos.

**Fijar material** En la ventana de **Material**, barra **Ventanas**, se fijan las propiedades del material. El software permite generar modelos con materiales isotrópicos elásticos lineales, por lo tanto los parámetros a ingresar son el Módulo de Poisson y el módulo de elasticidad. La ventana de material se muestra en la figura 4.9

**Condiciones de borde** Finalmente se fijan las condiciones de desplazamiento y la carga con las ventanas de desplazamiento y carga respectivamente, como se muestra en la figura 4.10. El software permite fijar condiciones de desplazamiento de simetría y de restricción de grados de libertad. Para fijar las condiciones

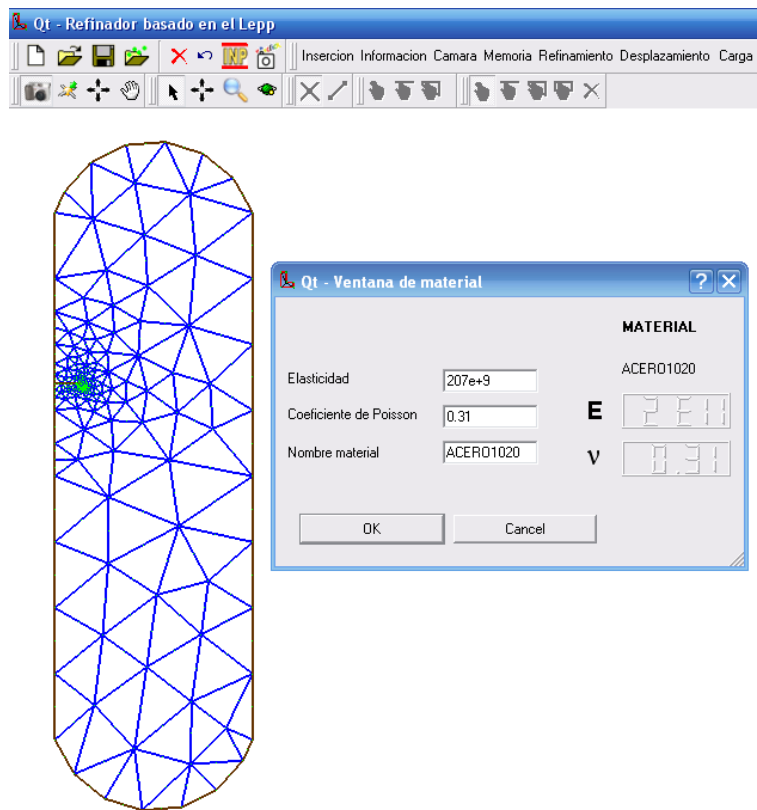


Figura 4.9: En la ventana de material, se fijan las propiedades del material a modelar.

de borde sobre una cara, se deben seleccionar todas las aristas que incluyen esta cara, lo mismo para la carga. El software permite fijar cargas distribuidas homogéneas normales o tangenciales, y cargas puntuales. El detalle de esta parte se puede ver en el capítulo 4.4.1

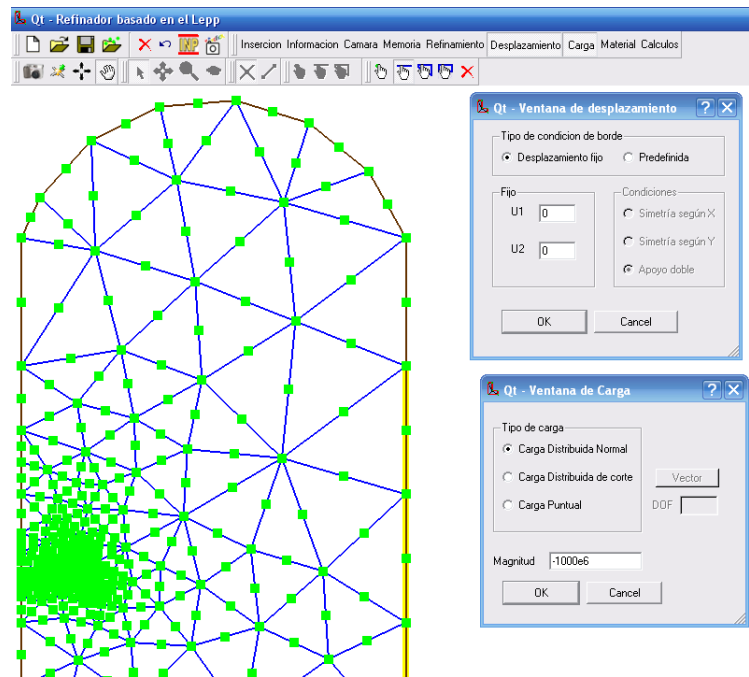


Figura 4.10: Fijación de las condiciones de desplazamiento y de carga distribuida.

**Cálculo** Una vez definido el modelo, la ventana de cálculo posee todas las funciones necesarias para calcular los factores de intensificación de esfuerzos o simular una propagación de grieta.

La ventana de cálculos se puede ver en la figura 4.11. En ésta se puede seleccionar si el modelo será de esfuerzo plano o deformación plana, y si se utilizarán elementos especiales con el nodo desplazado o elementos normales.

Cuando todo esté listo, se aprieta el botón “calcular”. Al hacer esto, el programa genera un archivo de texto con el formato de entrada de ABAQUS (extensión .inp) con la información definida en el programa. Además, con esta misma información, genera un script en el lenguaje de programación Python (que es un archivo de texto con extensión .py) y luego lo ejecuta. El script posee un encabezado con las librerías necesarias para acceder a las estructuras de datos de Abaqus. El script llama al solver para resolver el modelo definido en el archivo .inp y espera que se resuelva. Una vez resuelto, accede a la información de salida del modelo y obtiene los datos necesarios para el post-proceso, y finalmente realiza los cálculos que son escritos en un archivo de texto. Cuando termina de ejecutarse el script, el software lee el archivo de texto y muestra los resultados en la misma ventana de cálculos. Los resultados son los valores de  $K_I$  y  $K_{II}$ , además del ángulo en el que se propagará la grieta. Esto puede apreciarse en la figura 4.11.

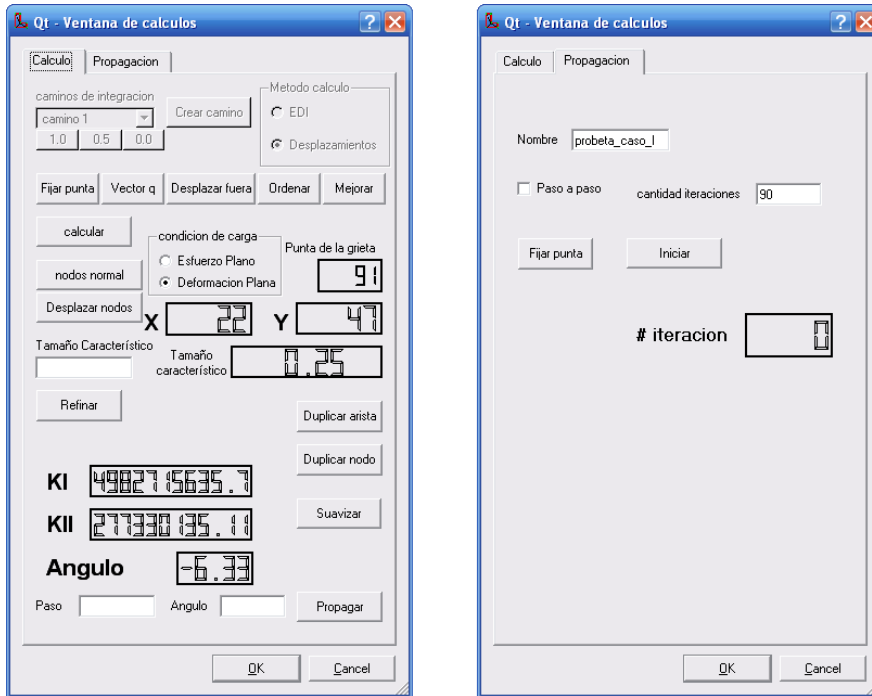


Figura 4.11: Ventana de cálculos del software.

**Paso de propagación** Una vez obtenidos los valores de  $K_I$ ,  $K_{II}$  y el ángulo, puede hacerse un paso de propagación de la grieta. Para esto, primero se reubican en su posición normal los nodos desplazados, se fija un “paso” (tamaño de crecimiento de la grieta) y luego se presiona el botón ‘propagar’. Con esto se inserta un nuevo punto dado por el paso y el ángulo, o se desplaza un nodo si es que este nuevo punto está muy cerca de otro ya existente.

En el caso de querer simularse la propagación en forma automática, en el tab **Propagación**, una vez definida la roseta inicial, se fija la punta, se define la cantidad de iteraciones y se aprieta el botón ‘iniciar’. El programa automáticamente genera la nueva malla a partir de la anterior en cada uno de los pasos.

#### 4.5 Archivos de entrada y salida

El software guarda la información del modelo en un archivo en formato “mtr”. Este archivo está estructurado de la siguiente manera

—Primera parte—

```

ARCHIVO_MALLA_TRIANGULOS INSERCIÓN_EXTERIOR_DESHABILITADA
PUNTOS: 380
1 0 0 0
2 100 0 0
3 100 100 0
4 22.6025390625 17.32208251953125 0

```

5 63.06640625 44.43359375 0  
6 10.3125 61.875 0  
7 0 100 0  
. . .  
378 20.632080435752869 19.388374090194702 0  
379 20.092083811759949 18.749626874923706 0  
380 20.645806193351746 18.662852048873901 0

—segunda parte—

TRIANGULOS: 170

1 99 35 95 102 17 36  
2 129 125 19 9 128 130  
3 235 67 22 236 70 8  
4 60 31 57 62 18 14  
5 67 42 64 16 24 6  
6 224 25 60 227 61 225  
. . .  
168 373 325 313 375 262 376  
169 377 373 369 372 316 379  
170 377 369 317 379 371 380

—tercera parte—

ARISTAS: 275

5 16 N 16  
1 39 N 17  
141 111 N 4

. . .

—cuarta parte—

PUNTAS: 2

10  
11

—quinta parte—

MATERIAL\_ISOTROPICO

Acero

200000000000 , 0.299999999999999999

—sexta parte—

cond\_desp 0

221 25 223 346 347 343 348 1 345 2 26 .

0 , 0

—séptima parte—

cond\_carga 0

S1

-1

S2

7 8 54

S3

-1

carga

-500000000

El archivo consta de varias partes cuyo orden no puede variar, y se describen a continuación

- La primera parte enumera los nodos, con el formato *índice coord\_x coord\_y coord\_z*.
- La segunda parte enumera los triángulos, con el formato *índice nodo\_1 nodo\_2 nodo\_3 nodo\_4 nodo\_5 nodo\_6*, donde *nodo\_i* corresponde a la numeración de la figura 3.2.
- La tercera parte enumera las aristas, con el formato *vertice\_1 vertice\_2 [Restringida/No restringida/-1(borde)] nodo\_medio(de la arista)*.
- En la cuarta parte se enumeran los nodos que son puntas de grieta, que pueden ser varios si hay varias grietas en el modelo.
- La quinta parte muestra las propiedades del material.
- La sexta parte enumera las condiciones de desplazamiento, agrupadas como *cond\_desp\_i*, luego el conjunto de nodos que incluye la condición de desplazamiento, y luego el tipo de condición de desplazamiento (xsymm= simetría según x; ysymm=simetría según y; a,b= grado de libertad 1 fijado en a, y grado de libertad 2 fijado en b, respectivamente).



- La séptima parte incluye la información de las condiciones de carga. Éstas se agrupan como *cond\_carga\_i*, luego los grupos S1, S2 y S3 que indican qué cara del elemento es está cargando y cuáles son cargados en esta cara, y luego la carga en sí. Este formato es similar al usado en ABAQUS. Por ejemplo, en el caso mostrado la condición de carga 0 involucra tres elementos (se seleccionas aristas para fijar la condición de carga, pero se almacenan según los elementos). De esos tres elementos, ninguno (por eso el -1) está cargado por la cara S1 (que corresponde a la arista entre los nodos 1 y 2), tres están cargados por la cara S2 (arista entre los nodos 2 y 3), y ninguna está cargado en la cara S3 (nodos 1 y 3). Luego la carga es de -500000000, el signo negativo significa que la carga apunta hacia el exterior del elemento.

El software puede generar con esta misma información un archivo ‘inp’, que es el formato utilizado por ABAQUS para ingresar modelos de elementos finitos. Para generar el archivo ‘inp’, el software lee el archivo ‘mtr’, por lo tanto ambos formatos son guardados a la vez. Los detalles del formato del archivo ‘inp’ están disponibles en la documentación de ABAQUS, disponible una vez instalado en una estación de trabajo.

#### 4.6 Post-proceso e interacción con ABAQUS

Una de las principales características del software desarrollado es su integración con el software ABAQUS para la resolución de los modelos creados y el post-proceso para calcular los factores de intensificación de esfuerzos.

El software posee una función que escribe la información del modelo en un archivo ‘.inp’ que es el formato leído por ABAQUS, como ya se ha mencionado antes. Este archivo se genera al presionar el botón correspondiente en la barra de herramientas del programa. Luego este archivo puede ser resuelto con el solver de Abaqus a través de la línea de comandos o ser importado al módulo CAE de Abaqus para ser modificado o resuelto desde él.

En el caso de querer obtener los factores de intensificación de esfuerzos, que es el fin último del software, el modelo puede ser resuelto en forma automática desde el software.

Cuando se presiona el botón “calcular” en la ventana de cálculos, si todo está bien definido, a partir de la información existente, se generarán dos archivos de texto. El primero es un archivo en formato .inp, que guarda la información del modelo de elementos finitos. Esto implica definición de nodos y elementos (coordenadas, índices, tipo, etc.), definición del material, definición de las condiciones de borde, definición de la información que se pide al solver (desplazamientos, energía, etc.). Este archivo está escrito en un formato legible por el programa ABAQUS. El segundo archivo es un script escrito en el lenguaje *Python*. Dependiendo si se desea realizar el post-proceso con el método de extrapolación de desplazamientos o con el método de la integral J se generarán scripts distintos. A continuación se muestra el caso del script generado para el caso del método de extrapolación de desplazamientos, que es el que está implementado:

```
"""
script para calcular extrapolación de desplazamientos
```

```

"""

from abaqus import *
from abaqusConstants import *
import visualization

#####
#Codigo
#####

#####
#variables
#####
theta=0.0
despl = 1
e= 2.07e+011
nu=0.3
theta=67.5# en grados
Pi = 3.14159265359
pa =155
pb =162
pm1 =156
pm2 =164
va=[]
vb=[]
l =0.00239177145
ptip=[0.0595670858,0.0855969883]
#####
#acceso a base de datos  resolución del modelo
#####
f = open('calculosdesp.txt', 'w')
basemodelos=Mdb()
trabajo=basemodelos.JobFromInputFile(name='Malla', inputFileNames='Malla.inp')
trabajo.setValues(nodalOutputPrecision=FULL)
trabajo.submit()
trabajo.waitForCompletion()
myOdb = visualization.openOdb(path='Malla.odb')
firstStep = myOdb.steps['Step-1']

```

```

parte = basemodelos.Model('modelo').PartFromOdb(name='PART-1', odb=myOdb)
frame1 = firstStep.frames[-1]
displacement = frame1.fieldOutputs['U']

va=[displacement.values[pa-1].dataDouble[0],
    displacement.values[pa-1].dataDouble[1]]
vb=[displacement.values[pb-1].dataDouble[0],
    displacement.values[pb-1].dataDouble[1]]
vpm1=[displacement.values[pm1-1].dataDouble[0],
    displacement.values[pm1-1].dataDouble[1]]
vpm2=[displacement.values[pm2-1].dataDouble[0],
    displacement.values[pm2-1].dataDouble[1]]
#####
# calculos
#####
thetar=Pi*theta/180.0
vap=[va[0]*cos(thetar)+va[1]*sin(thetar),
     -va[0]*sin(thetar)+va[1]*cos(thetar)]
vbp=[vb[0]*cos(thetar)+vb[1]*sin(thetar),
     -vb[0]*sin(thetar)+vb[1]*cos(thetar)]
vpm1p=[vpm1[0]*cos(thetar)+vpm1[1]*sin(thetar),
       -vpm1[0]*sin(thetar)+vpm1[1]*cos(thetar)]
vpm2p=[vpm2[0]*cos(thetar)+vpm2[1]*sin(thetar),
       -vpm2[0]*sin(thetar)+vpm2[1]*cos(thetar)]

if(despl==0):
ki=e/4/(1-nu*nu)*sqrt(2*Pi/l)*(vap[1]-vbp[1])/2
kii=e/4/(1-nu*nu)*sqrt(2*Pi/l)*(vap[0]-vbp[0])/2
else:
ki=e/4/(1-nu*nu)*sqrt(2*Pi/l)*(4*(vpm1[1]-vpm2[1])/2-(vap[1]-vbp[1])/2)
kii=e/4/(1-nu*nu)*sqrt(2*Pi/l)*(4*(vpm1[0]-vpm2[0])/2-(vap[0]-vbp[0])/2)
#####
#escritura de los resultados
#####
if(ki>=0):
f.write(repr(ki)),f.write('\n')
else:
f.write(repr(-ki)),f.write('\n')
if(kii>=0):

```

```

f.write(repr(kii)),f.write('\n')
else:
f.write(repr(-kii)),f.write('\n')

angulo=180.0*acos((3*kii*kii+sqrt(ki*ki*ki*ki+8*ki*ki*kii*kii))
/(ki*ki+9*kii*kii))/Pi
f.write(repr(angulo)),f.write('\n')
f.close()

```

Como se mencionó, el script está escrito en el lenguaje *Python*. Lo primero que hace el script es importar las funciones necesarias para poder interactuar con ABAQUS. Luego, a partir de los datos del modelo, se definen las variables necesarias para los cálculos. las variables *pa*, *pb*, *pm1* y *pm2* son los índices de los nodos de los cuales se obtendrán los desplazamientos para calcular los factores de intensificación de desplazamientos según las ecuaciones 3.40 y 3.41. Luego se crea un modelo de ABAQUS a partir del archivo *inp* (*JobFromInputFile()*), y se resuelve (*submit()*). Una vez resuelto, se abre la base de datos de salida (Output Data Base) desde donde se obtienen los resultados. Desde este objeto pueden obtenerse el campo de deformación o los tensores de esfuerzo y deformación para un nodo o elemento, dando como parámetro su índice. Con estos datos finalmente se realizan los cálculos y se escriben en un archivo de texto.

Una vez que se genera el script se ejecuta. Cuando termina de ejecutarse, el programa lee el archivo de texto generado por el script y muestra los resultados en la ventana de cálculos.

En este capítulo se mostró el trabajo realizado para desarrollar el software utilizado en esta tesis. Esto involucra los aspectos técnicos de la implementación, así como una descripción detallada de su funcionamiento, incluyendo los distintos datos de entrada y salida. Este software se usa entonces para validar la metodología propuesta en esta tesis, probando una serie de casos con soluciones numéricas y/o experimentales conocidas. Los resultados de estas pruebas se muestran en el siguiente capítulo.

## Capítulo 5

### Resultados

#### 5.1 Introducción

En este capítulo se muestran los resultados del trabajo realizado en esta tesis. Se probaron los algoritmos de refinamiento Lepp-Delaunay para refinar localmente en la punta de una grieta. Esto permitió calcular los factores de intensificación de esfuerzos para configuraciones con solución conocida, con el objeto de analizar y discutir las ventajas y desventajas de la metodología propuesta, así como probar la potencialidad del software desarrollado. También se muestra una metodología para simular el crecimiento de una grieta, que utiliza el refinamiento localizado basado en los algoritmos ya mencionados además de modificar la malla geométrica para simular explícitamente el crecimiento.

#### 5.2 Mallado y refinamiento localizado

La principal ventaja de los algoritmos basados en el LEPP es la posibilidad de poder refinar una malla ya existente sin rehacerla. También permiten generar mallas iniciales a partir de una discretización de los bordes del objeto geométrico, lo que vendría siendo una implementación particular del mallado Delaunay. Se utilizaron los algoritmos Lepp-Delaunay para generar mallas en variadas geometrías y analizar su comportamiento. Con respecto al refinamiento, se proponen estrategias de refinamiento localizado para zonas críticas, particularmente ángulos agudos y grietas, que es el objetivo de esta tesis. El detalle de estas metodologías se describe a continuación. Además se implementó un algoritmo de desrefinamiento que elimina pares de triángulos que comparten una arista muy pequeña.

##### 5.2.1 Generación de mallas iniciales

En el capítulo 3.2.2 se mencionaron los distintos métodos para generar una malla inicial de una geometría arbitraria. En el software desarrollado se incluye una implementación particular del método Delaunay. En dicho capítulo se menciona que para generar una malla inicial, la geometría se encierra en un espacio convexo y luego sobre él se insertan los puntos del borde de la geometría. En el caso de este software no es necesaria la geometría convexa, sino que los puntos se insertan directamente en el borde de la geometría a mallar. El método se muestra en la figura 5.1

El primer paso es cargar una geometría. Para esto se implementó un intérprete del formato IGES, como se mencionó en el capítulo 4. Una vez cargada la geometría, se define un tamaño característico de la malla.

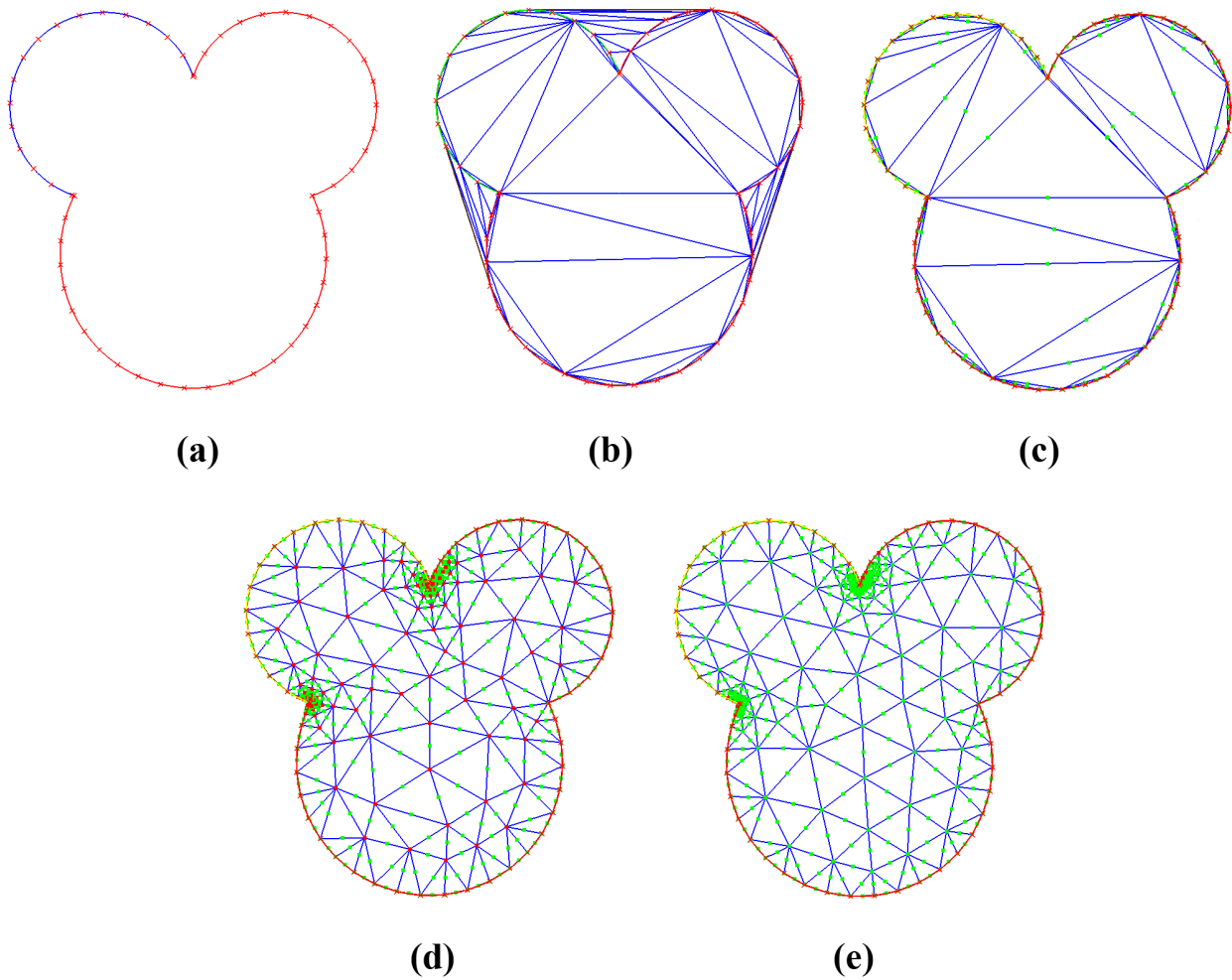


Figura 5.1: Metodología de mallado Delaunay implementada. (a) Seed de los bordes. (b) Inserción de puntos en el borde, generándose una malla inicial convexa. (c) Eliminación de los elementos fuera del borde. (d) Malla de mejor calidad generada aplicando secuencialmente un algoritmo Lepp-Delaunay. (e) Suavizamiento laplaciano de la malla.

Este se puede definir fijo para toda la malla, o puede definirse en forma independiente para cada una de las curvas que forman la geometría poligonal. El tamaño característico en la práctica significa fijar el espaciado de los nodos que se insertarán en el borde de la geometría, como muestra la figura 5.1(a).

Una vez fijado el tamaño característico, se insertan secuencialmente nodos en el modelo, siguiendo la posición determinada por el “seed” fijado para cada curva de la geometría. Con esto se genera una malla del conjunto convexo de los nodos insertados, como se muestra en la figura 5.1(b).

En el caso que la geometría no sea convexa, algunos de estos triángulos quedarán fuera de la geometría, por lo que en un siguiente paso deben ser eliminados del modelo, como muestra la figura 5.1(c), obteniéndose una malla inicial que respeta la geometría, pero con pocos elementos y de mala calidad.

El siguiente paso es generar la malla de buena calidad a partir de la anterior. Para eso se refinan secuencialmente todos los triángulos de la malla hasta que su tamaño característico (que se define como cualquiera de sus lados) sea menor que el mayor de los “seeds” fijados en los bordes, y que cumplan con el criterio de calidad fijado en la ventana de información. Con esto se obtiene la malla final mostrada en 5.1(d). Posterior a eso se deben reparar los índices de los nodos y los elementos de tal forma que sean correlativos.

Posterior a la malla inicial pueden hacerse dos cosas, una es generar un refinamiento localizado y la otra es un suavizamiento laplaciano (figura 5.1(e)). En la figura 5.2 se muestra la secuencia del refinamiento localizado en torno a una esquina. Dado que los algoritmos Lepp-Delaunay insertan nodos en una malla existente, no es necesario rehacer la malla para obtener una nueva refinada. Del mismo modo, la inserción de nuevos nodos respeta la correlatividad de los índices.

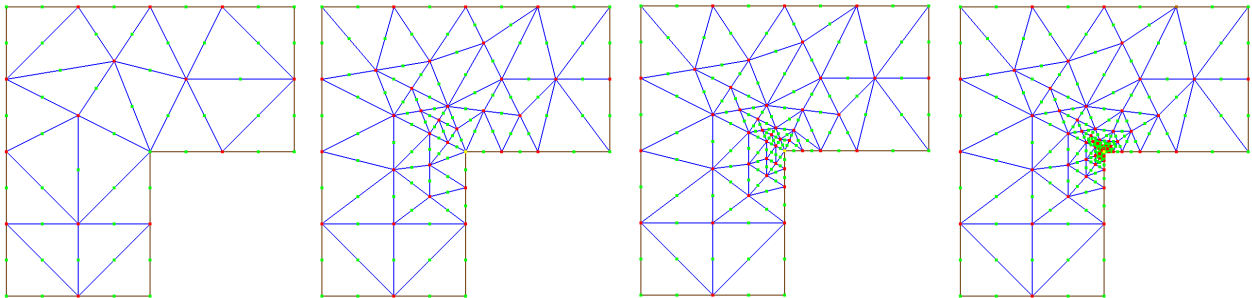


Figura 5.2: Secuencia de refinamiento usando el algoritmo Lepp-Delaunay sobre una esquina.

**Pruebas de la metodología** Para probar la calidad de las mallas generadas por los algoritmos Lepp-Delaunay, se probaron tres casos representativos a juicio del autor. Lo que se hizo fue generar una malla inicial de una geometría utilizando la metodología Delaunay mencionada anteriormente, y luego sobre dicha malla aplicar un suavizamiento laplaciano. Se compara entonces la calidad de la malla generada sólo con el algoritmo Lepp-Delaunay, con la calidad de la malla posterior al suavizamiento laplaciano.

En las figuras 5.3 (Caso 1) 5.5 (Caso 2) y 5.7 (Caso 3) se muestran tres geometrías cuyas mallas se generaron con el software desarrollado, antes y después del suavizamiento laplaciano. La primera geometría corresponde a una placa con una grieta en el borde. El segundo caso corresponde a una geometría convexa cualquiera. El tercer caso corresponde a una geometría cóncava con forma de “L”. En las figuras 5.4, 5.6 y 5.8

se muestra respectivamente la distribución de calidad de los elementos de los tres casos mencionados, para el caso de la malla generada con el algoritmo LEPP-Delaunay y para el posterior suavizamiento laplaciano, utilizando la ecuación 3.26. Se puede observar que existe una mejora de calidad debida al suavizamiento laplaciano, aunque no es indispensable, puesto que los algoritmos Lepp-Delaunay generan mallas suficientemente buenas. Un indicador  $q = 0,7$  corresponde aproximadamente a un triángulo con su ángulo mínimo igual a  $30^\circ$ .

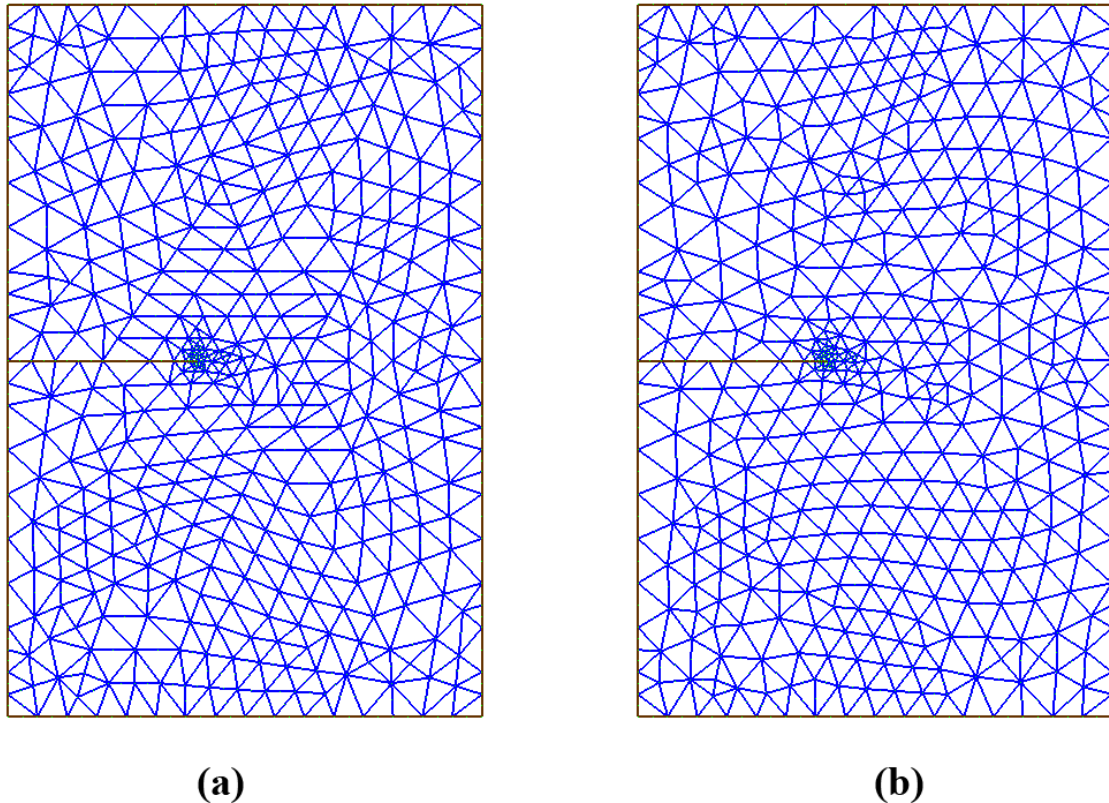


Figura 5.3: Malla del caso 1. (a) Malla generada utilizando el algoritmo Lepp-Delaunay. (b) Suavizamiento laplaciano de la misma malla.

## 5.2.2 Refinamiento de la punta de la grieta

### Introducción

Una grieta es una falla lineal (1D) en el caso de un sólido 2D. Dado que la punta de la grieta es un concentrador de esfuerzos, obtener adecuadamente los tensores de esfuerzos y deformaciones en esta zona por medio de un método numérico no es un problema trivial.

La aproximación más intuitiva para resolver el problema es usar el método de elementos finitos y modelar la grieta explícitamente, es decir incluir las superficies libres que conforman la grieta. Al utilizar el método de elementos finitos los altos gradientes de esfuerzos cerca de la punta de la grieta hacen tener especial



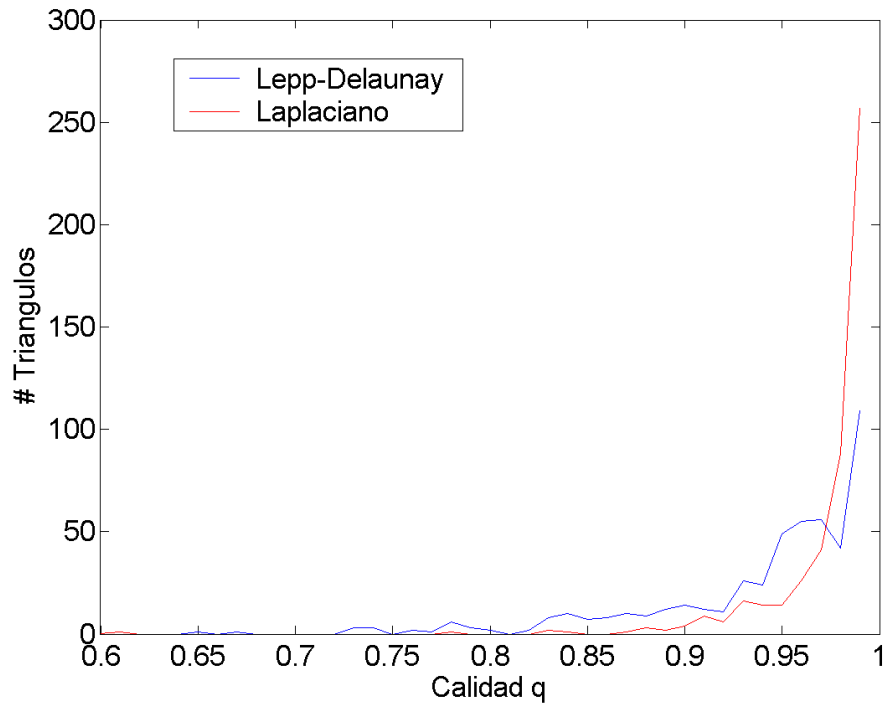


Figura 5.4: Distribución de la calidad de los elementos para el caso 1. En rojo la distribución de calidad obtenida con el algoritmo LEPP-Delaunay. En azul, la distribución de calidad obtenida con el suavizamiento laplaciano de la misma malla.

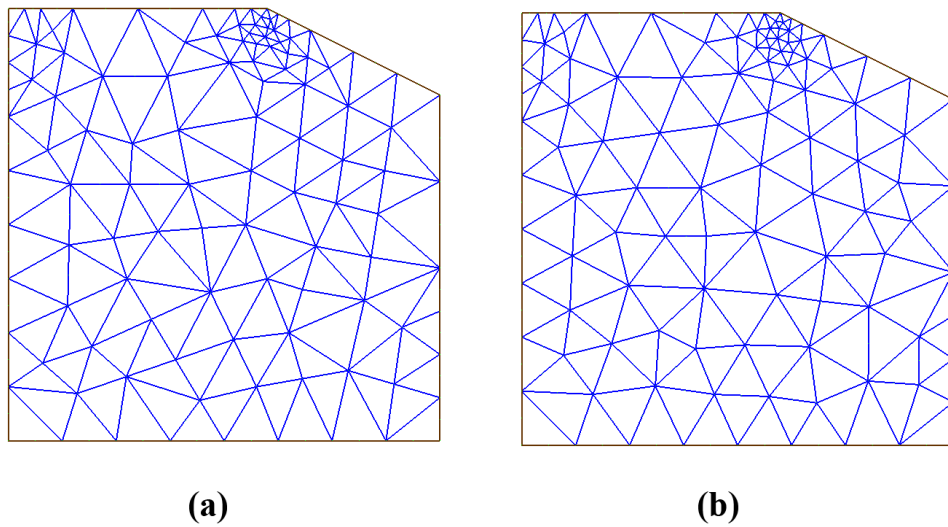


Figura 5.5: Malla del caso 2. (a) Malla generada utilizando el algoritmo Lepp-Delaunay. (b) Suavizamiento laplaciano de la misma malla.

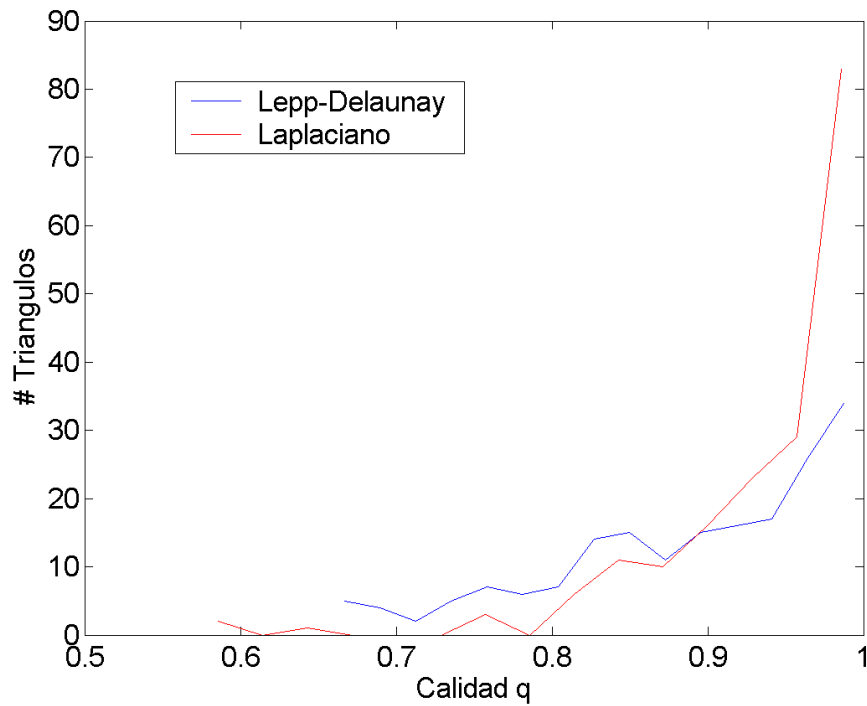


Figura 5.6: Distribución de la calidad de los elementos para el caso 2. En rojo la distribución de calidad obtenida con el algoritmo LEPP-Delaunay. En azul, la distribución de calidad obtenida con el suavizamiento laplaciano de la misma malla.

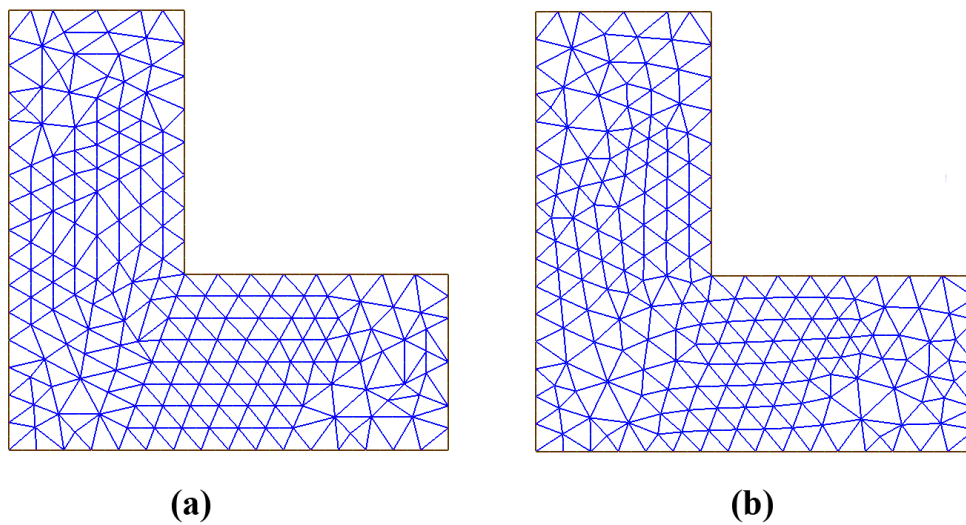


Figura 5.7: Malla del caso 3. (a) Malla generada utilizando el algoritmo Lepp-Delaunay. (b) Suavizamiento laplaciano de la misma malla.

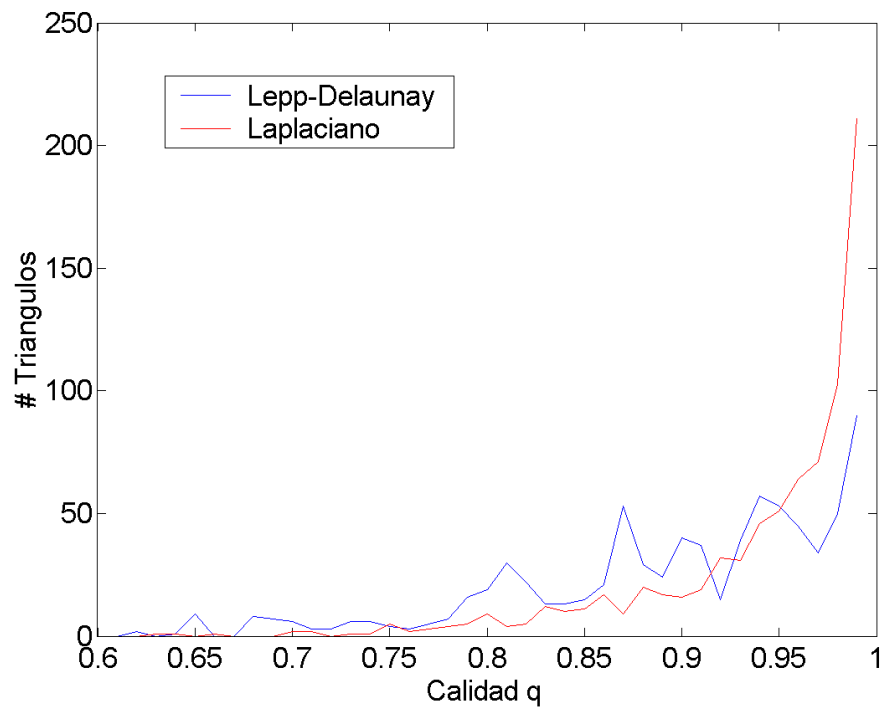


Figura 5.8: Distribución de la calidad de los elementos para el caso 3. En rojo la distribución de calidad obtenida con el algoritmo LEPP-Delaunay. En azul, la distribución de calidad obtenida con el suavizamiento laplaciano de la misma malla.

dedicación en esta zona, en particular en lo que respecta al refinamiento de la malla.

Existen otros métodos para representar una grieta en un modelo de elementos finitos. Por ejemplo, el método 'extended finite element method' (X-fem), desarrollado originalmente por Moës *et al.* (1999), utiliza los conceptos del método de la partición de la unidad (Babuska & Melenk, 1997) y consiste, a grandes rasgos, en enriquecer las funciones de forma en la zona de la punta de la grieta con nuevas funciones asintóticas que simulan el campo existente en esta zona, lo que evita la modelación explícita y por lo tanto la modificación de la malla. Otro método es el llamado s-version FEM (Fish, 1992; Okada *et al.*, 2005a), donde se superpone una malla más fina en la zona donde se encuentra la grieta, y el sólido global se modela con una malla gruesa como si no tuviese una grieta. Una técnica similar al s-fem ha sido aplicada en 3D (Schöllmann *et al.*, 2003).

Aunque los métodos mencionados arriba entregan muy buenos resultados, en los últimos años los avances en técnicas de generación y refinamiento de mallas han sido considerables, por lo que es posible abordar el problema de modelar una grieta desde la solución más intuitiva, es decir la modelación explícita sin mayores problemas. El verdadero desafío en el mediano y largo plazo es obtener mejores formas de abordar la mecánica de la fractura numérica en el caso de sólidos y grietas arbitrarias en 3D, campo que aún tiene mucho terreno por explorar.

### **Implementación del modelamiento de la grieta**

Se implementó en el software una metodología para modelar la grieta explícitamente. Esta idea consiste básicamente en hacer coincidir la grieta con aristas de la malla y luego duplicar las aristas y nodos que caen sobre ella para formar dos superficies libres, como muestra la figura 5.9. Esto, en conjunto con la utilización de elementos especiales en la punta de la grieta, permite obtener una buena aproximación de los campos y tensores en esta zona.

Los pasos para crear un modelo con una grieta se muestran en la figura 5.10. Primero se crea la geometría que se desea modelar, sin grieta. Luego se inserta un nodo al inicio de la grieta y otro al final de la grieta. Con esto se crea una arista de la malla que coincide geoméricamente con la posición de la grieta, como muestra la figura 5.10(b). Una vez creada la arista, ésta es duplicada, creando dos superficies libres. Si la grieta llega hasta la superficie, se debe duplicar el nodo que se ubica en la intersección entre la superficie y la grieta (figura 5.10(c)). Finalmente, se refina en la punta de la grieta (figura 5.10(d)). Si la grieta a modelar es curva, entonces se insertan varios nodos con tal de aproximar la curva por una serie de aristas. El algoritmo para refinar en la punta de la grieta es el siguiente

El refinamiento Lepp-Delaunay entrega elementos del tamaño deseado, pero son de forma irregular, particularmente en la punta de la grieta. Como se verá más adelante, los elementos irregulares entregan malos resultados numéricos, así que es necesario crear una roseta de 8 elementos simétricos. Para realizar esto, se implementó la metodología que se describe en la figura 5.11. La metodología consta de tres pasos. Primero, se refinan los elementos de la punta de la grieta hasta que tengan el tamaño deseado. El radio de la roseta está dado por el largo de la arista que coincide con la cara de la grieta, como muestra la figura 5.11(a). Luego de definido este radio, se definen las posiciones de los nodos que conforman la roseta (figura 5.11(b)) y se insertan nuevos nodos en esas posiciones, o se desplazan a esa ubicación algunos nodos existentes si es que están cerca (figura 5.11(c)).

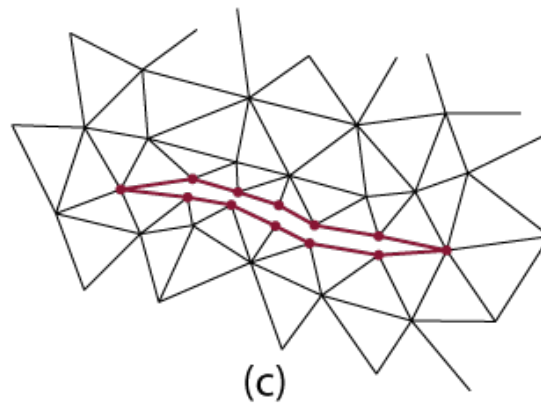
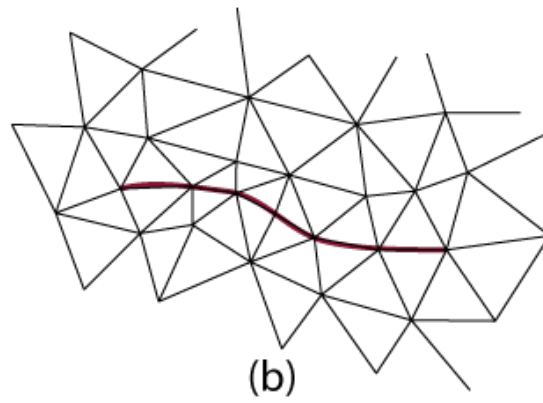
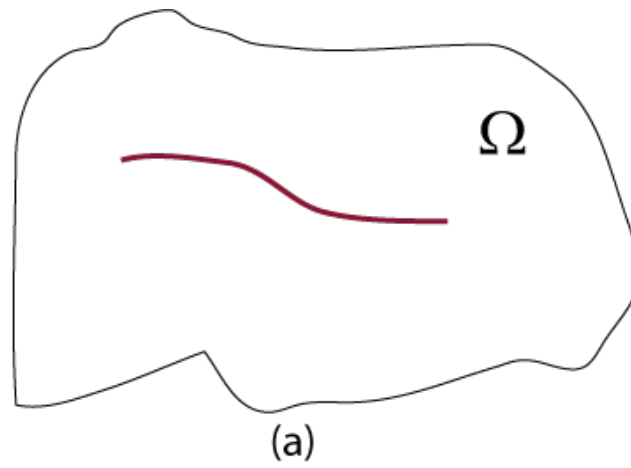


Figura 5.9: Modelación explícita de la grieta. Se hace coincidir la grieta (1D) con un conjunto de aristas, las que luego son duplicadas con sus respectivos nodos conectores. Esta metodología también se usa para simular la propagación.

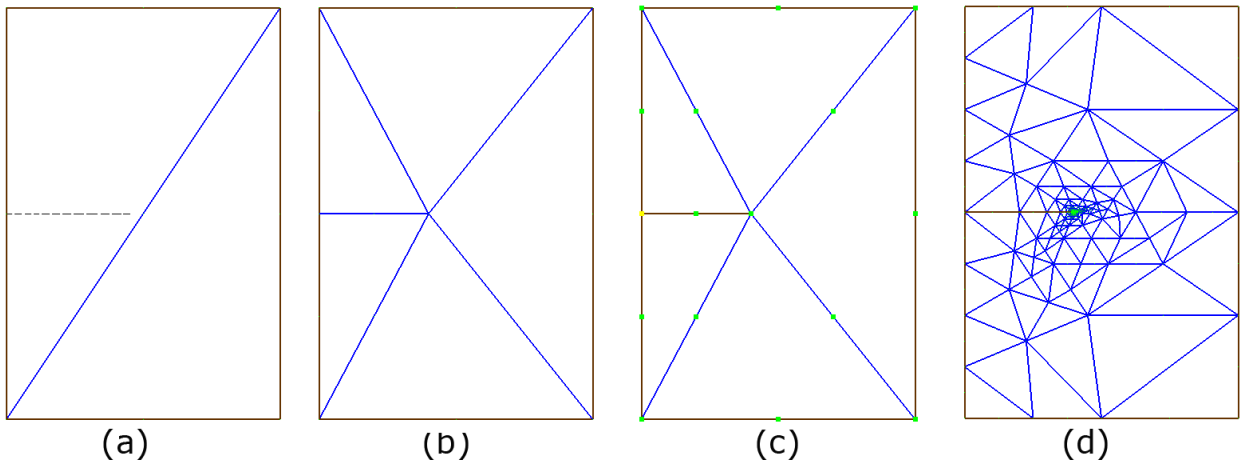


Figura 5.10: Pasos para incluir una grieta en el modelo. (a) se crea la geometría original, sin grieta. (b) Se insertan dos nodos en los extremos de la grieta, para que una arista coincida geoméricamente con ella. (c) Se duplica la arista y el nodo de la orilla, si es necesario. (d) Se refina la punta de la grieta .

---

**Algorithm 5** Refinar\_Punta( $p, l$ ),  $p$  punta a refinar,  $l$  largo característico final,  $T$  conjunto de elementos que tocan la punta

---

```

while tamaño característico de  $t \in T$  sea mayor que  $l$  do
  for all  $t \in T$  do
    Refinar-LEPP( $t$ )
    encontrar el nuevo conjunto  $T$ 
  end for
end while

```

---

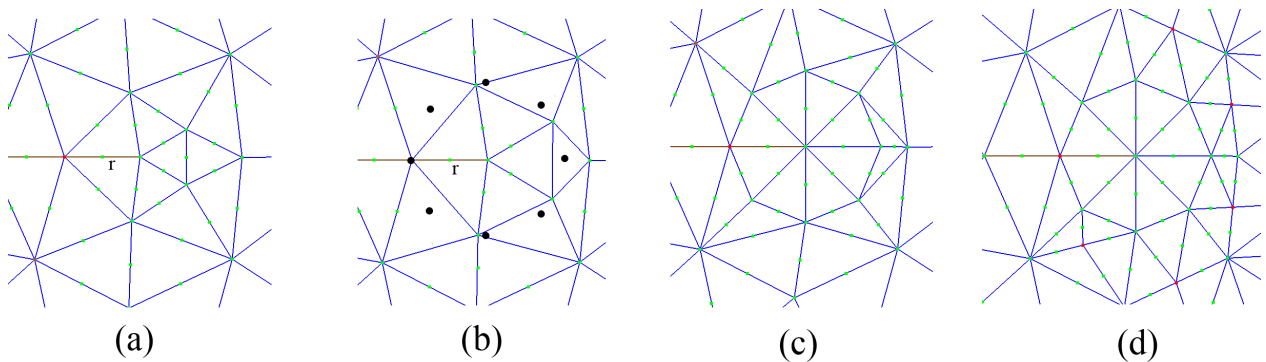


Figura 5.11: Metodología para generar una roseta simétrica. (a) Elementos en la punta de la grieta después de refinar con un algoritmo Lepp-Delaunay. La cantidad de elementos que tocan la punta de la grieta es variable, así como su geometría. Se define un radio de la roseta, que corresponde al largo de la arista que toca la cara de la grieta. (b) Se desplazan los nodos que están dentro del radio definido anteriormente. (c) Se define la posición de los nodos de la roseta, y secuencialmente se insertan nodos en estas posiciones o, si uno de los nodos existentes está cerca, se desplaza a esta nueva posición. (d) Al generar la roseta, se generan elementos de mala calidad que son mejorados con el mismo algoritmo Lepp-Delaunay.

Debido a los desplazamientos o inserciones, los triángulos que están conectados a la roseta generalmente son de mala calidad. Por lo tanto, posterior a esto es necesario un mejoramiento de la malla, como muestra la figura 5.11(d). Este mejoramiento se hace con uno de los algoritmos Lepp-Delaunay, pudiendo ser el algoritmo *LEPP-Delaunay-AristaTerminal* o el algoritmo *LEPP-Delaunay-CentroidesTT* (el algoritmo LEPP-Bisección no mejora la malla). Más adelante, en los resultados, se discutirá la ventaja de utilizar los algoritmos LEPP para este paso. Posterior al mejoramiento, puede hacerse un suavizamiento laplaciano, con lo que se obtiene la malla final.

### 5.2.3 Eliminación de triángulos malos

Se implementó en esta tesis un algoritmo de triángulos de mala calidad. Este algoritmo opera sobre pares de triángulos que comparten una arista muy pequeña, como se muestra en la figura 5.12.

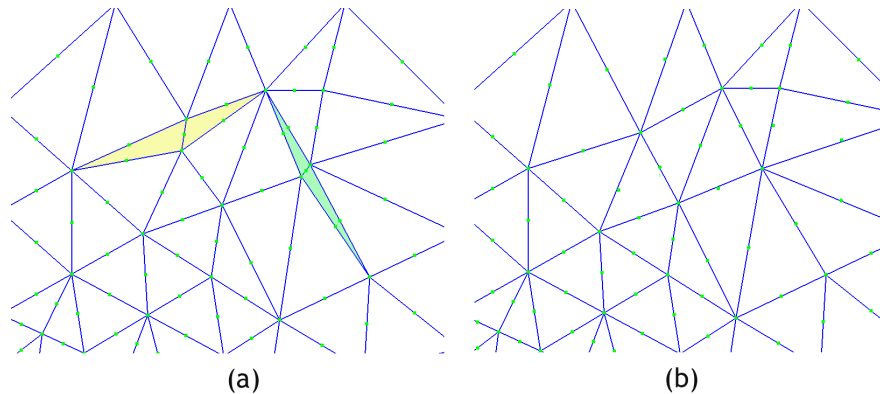


Figura 5.12: (a) Malla con pares de triángulos que comparten una arista muy pequeña. (b) Eliminación de dichos triángulos.

Si un triángulo tiene uno de sus lados mucho menor que los otros dos, entonces el ángulo opuesto a este lado será muy agudo, por lo que el elemento es de mala calidad, así que interesa eliminar este tipo de triángulos. Por otro lado, también puede ser necesario eliminar estos elementos por ser demasiado pequeños, que es lo que ocurre en esta tesis.

El procedimiento de eliminación de estos triángulos se muestra en la figura 5.13. Primero, se buscan en toda la malla las aristas que son más pequeñas que algún valor preestablecido, o si la relación entre el lado más pequeño y el más grande es menor que algún criterio. Una vez encontrada la arista, se seleccionan los dos triángulos que la comparten y se borran, quedando un agujero en el medio. Luego, se borra uno de los nodos que están frente a frente, y se asocia al otro a todos los triángulos (figura 5.13(c)), que se desplaza al punto medio entre los dos nodos. Finalmente, se unen los triángulos que quedan frente a frente, haciéndolos vecinos.

### 5.3 Cálculo del factor de intensificación de esfuerzos

Para probar la efectividad del método implementado, se calcularon los factores de intensificación de esfuerzos para varios problemas con soluciones conocidas. Como se mencionó en el capítulo 4, el software

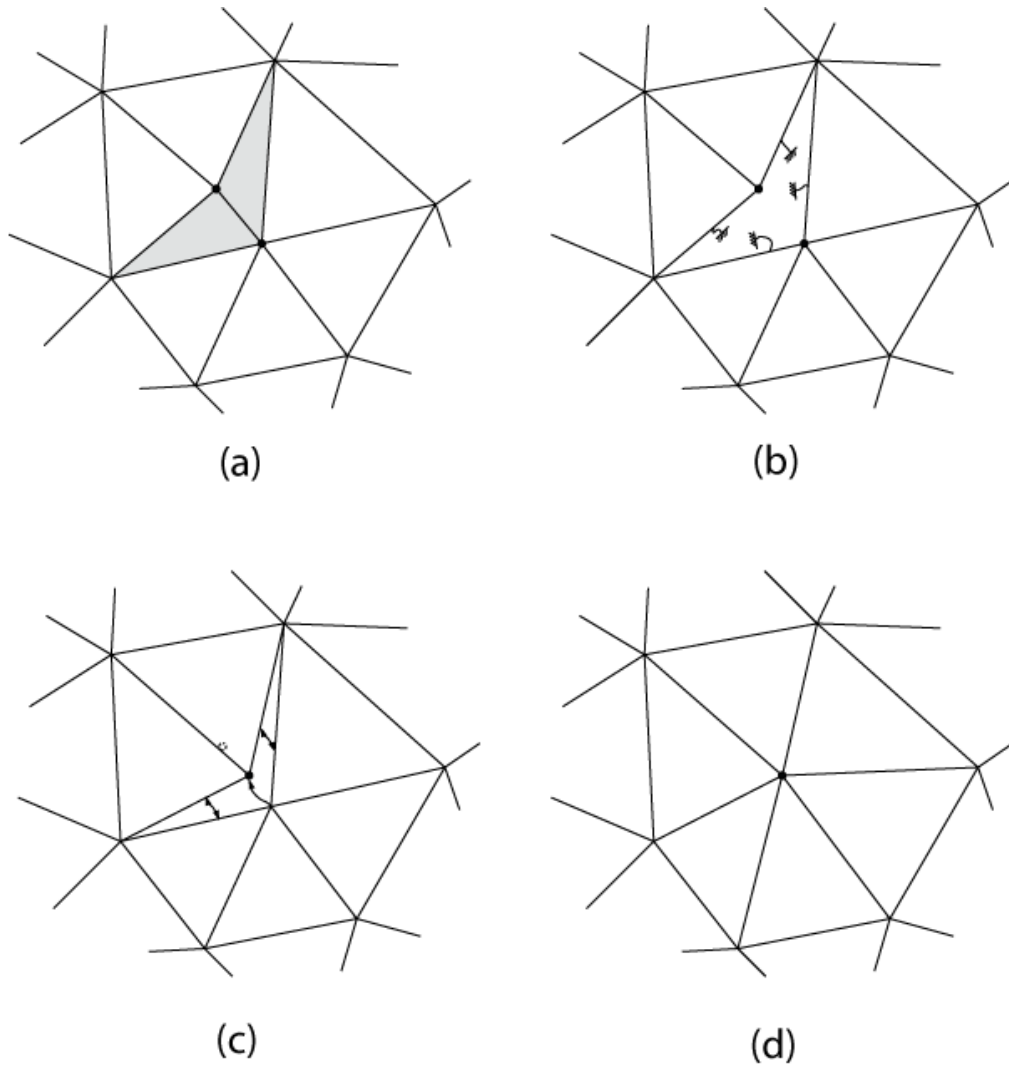


Figura 5.13: Procedimiento para eliminar pares de triángulos. (a) Se selecciona el par a eliminar. (b) Se borra el par de triángulos, quedando un agujero. (c) Se elimina uno de los nodos y se asocian sus triángulos al nodo del frente, y los triángulos que están frente a frente se hacen vecinos. El nodo que se conserva se desplaza al punto medio entre los dos nodos originales. (d) Se obtiene la malla final.



desarrollado permite generar un modelo de elementos finitos, es decir una malla geométrica además de las condiciones de borde y propiedades del sólido. Los modelos generados fueron resueltos utilizando el software comercial ABAQUS. A partir de los resultados entregados por ABAQUS se realiza un post-proceso para calcular los factores de intensificación de esfuerzos, cuyo resultado se muestra a continuación.

### 5.3.1 Placa con grieta en el borde

El primer caso a resolver es el de una placa con una grieta en la orilla, como muestra la figura 5.14<sup>1</sup>. Los valores de este problema son  $\sigma = 200[MPa]$ ,  $a = 0,025[m]$ ,  $W = 0,05[m]$ ,  $E = 210[GPa]$  y  $\nu = 0,3$  (material elástico lineal e isotrópico), y un estado de deformación plana.

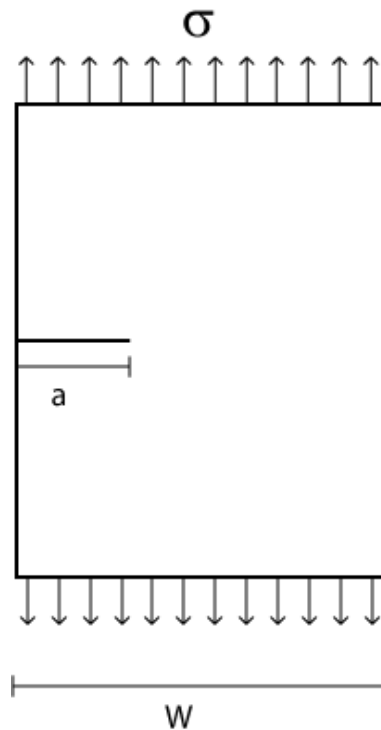


Figura 5.14: Configuración para una grieta de largo  $2a$  en la orilla de un sólido de ancho  $W$ .

Los resultados obtenidos se comparan con una solución obtenida por el método de la derivada de rigidez reportada por Banks-Sills & Sherman (1986), y cuyo resultado para el caso de deformación plana es  $K_I/K_0 = 2,818$ , donde  $K_0 = \sigma\sqrt{\pi a}$ .

En este caso se quiso observar la influencia del tamaño y forma de la malla en la punta de la grieta en los resultados numéricos. Primero se prueba una malla donde los elementos en la punta de la grieta tienen una configuración regular de 8 triángulos, como muestra la figura 5.15. Con esta malla se prueba el caso con elementos normales, y el caso con nodos desplazados un cuarto de su posición original. Luego se prueba con mallas irregulares. Se prueban cuatro casos, dos en que la punta de la grieta tiene seis triángulos (un caso

<sup>1</sup>Caso conocido en inglés como “single edge cracked plate in tension (SECT)”

con elementos normales y otro con nodos desplazados) y dos casos con 8 elementos en la punta (nuevamente uno con elementos normales y otro con nodos desplazados), como se muestra en la figura 5.15. Para calcular el factor de intensificación de esfuerzos se utiliza el método de extrapolación de desplazamientos.

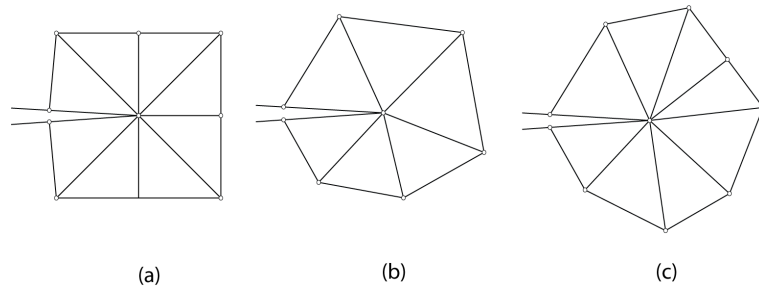


Figura 5.15: Configuraciones de elementos probados en el caso de la grieta en el borde. (a): 8 triángulos distribuidos en forma regular. (b): 6 triángulos irregulares. (c): 8 triángulos irregulares

El tamaño característico  $L$  de los elementos se elige de acuerdo a la relación propuesta por Raju (1987) ( $L/a = 1/16$ ). Los resultados se muestran en la tabla 5.1

CASO	nodos	elementos	$l$ [m]	$K_I/K_0$	Error( %)
Malla (a), elementos normales	259	112	0,0015625	2.663	5.5
Malla (a), elementos especiales	259	112	0,0015625	2.809	0.32
Malla (b), elementos normales	285	128	0,0015625	2.64	6.32
Malla (b), elementos especiales	285	128	0,0015625	2.63	6.67
Malla (c), elementos normales	293	132	0,0015625	2.67	5.25
Malla (c), elementos especiales	293	132	0,0015625	2.76	2.1
Solución de referencia				2.818	0

Tabla 5.1: Resultados para el cálculo de  $K_I$  para un sólido con una grieta en la orilla. Las mallas (a), (b) y (c) corresponden a las mostradas en la figura 5.15. Para todos los casos,  $l/a=1/16$ .

Se puede observar la importancia de los elementos en la punta de la grieta. El mejor resultado se obtiene con 8 elementos regulares, y con los nodos desplazados, obteniéndose un error de menos del 1 %. Al usar elementos normales, el error se mantiene del orden de 5-6 %, el cual podría disminuirse al refinar el tamaño de los elementos. Por otro lado se observa que al usar elementos irregulares, se obtiene una importante diferencia en el error al pasar de 6 a 8 elementos. Por lo tanto se debe preferir el uso de elementos regulares. Guinea *et al.* (2000) estudió en detalle la influencia de los elementos en el cálculo de los factores de intensificación de esfuerzos, y también observó que se obtienen mejores resultados aumentando la cantidad de triángulos alrededor de la punta de la grieta.

Como segundo paso, con el mismo problema, se prueba refinar los elementos de la punta de la grieta para observar la convergencia utilizando elementos normales. Los resultados se muestran en la tabla 5.2

Los refinamientos que conservan el tamaño característico se deben a que se refinó en otras zonas. Se puede observar que con el método de extrapolación de desplazamientos y con elementos normales, al refinar

CASO	nodos	elementos	l [m]	l/a	$K_I/K_0$	Error(%)
Caso base	285	128	0,0015625	16	2.64	6.32
Primer refinamiento	476	219	0,0015625	16	2.637	6.42
Segundo refinamiento	552	255	0,000390625	64	2.63	6.67
Tercer refinamiento	1085	518	0,000390625	64	2.683	4.79
Cuarto refinamiento	2403	1168	9.765675e-5	256	2.692	4.47
Quinto refinamiento	2760	1333	4.88292e-5	512	2.693	4.44
Solución de referencia					2.818	0

Tabla 5.2: Resultados para el cálculo de  $K_I$  para un sólido con una grieta en la orilla, probando varios niveles de refinamiento y utilizando elementos normales en la punta de la grieta.

sucesivamente se obtienen errores del orden del 4 %, sin bajar de ese valor. Si se desean errores del orden de 1 % se deben utilizar elementos especiales y mallas regulares en la punta de la grieta, pues esto asegura una rápida convergencia.

### 5.3.2 Grieta centrada

La segunda configuración que se probó es la mostrada en la figura 5.16, el caso de una placa con una grieta en el centro<sup>2</sup>. Dada la simetría, se resolverá 1/4 de la geometría, para cuatro niveles de refinamiento. También se resolverá la configuración completa, para observar la capacidad del software de refinar localizadamente. Este problema tiene solución analítica, y está dada por (Anderson, 1995)

$$K_I = \sigma \sqrt{\pi a} \left[ \sec \left( \frac{\pi a}{2W} \right) \right]^{0.5} \left[ 1 - 0,0025 \left( \frac{a}{W} \right)^2 + 0,06 \left( \frac{a}{W} \right)^4 \right] \quad (5.1)$$

Los valores de este problema son  $a = 1[m]$ ,  $W = 2[m]$ ,  $E = 2,07 \cdot 10^5 [MPa]$ ,  $\nu = 0,3$  y  $\sigma = 100 [MPa]$ , y un estado de deformación plana. Para estos valores, utilizando la ecuación 5.3.2 se obtiene  $K_I = 210,255 [MPa\sqrt{m}]$ .

La figura 5.17 muestra la malla utilizada para el caso de mayor refinamiento, y la malla para la grieta explícita. En el caso sin simetría se utiliza una roseta de 8 elementos triangulares simétricos como los mostrados en la figura 5.17(c). El refinamiento se realiza con el algoritmo LEPP-Delaunay. El cálculo de  $K_I$  se hará con el método de extrapolación de desplazamientos.

Los resultados de la simulación, errores y parámetros se resumen en la tabla 5.3. La longitud  $l$  mencionada en la tabla es el largo de la arista que toca la cara de la grieta, como se muestra en la figura 3.21.

Se observa en la tabla 5.3 que el método converge al refinar los elementos. El cálculo sin simetría obtiene menos error que los casos con simetría, pero no parece haber un motivo claro de ello.

### 5.3.3 Grieta en una muesca con forma de V

El tercer caso es el de una grieta ubicada en la punta de una muesca con forma de V, como se ve en la figura 5.18. Este es un caso de prueba propuesto por la “National Agency for Finite Element Methods and Stan-

<sup>2</sup>Conocido en inglés como “Center cracked plate in tension(CCT)”

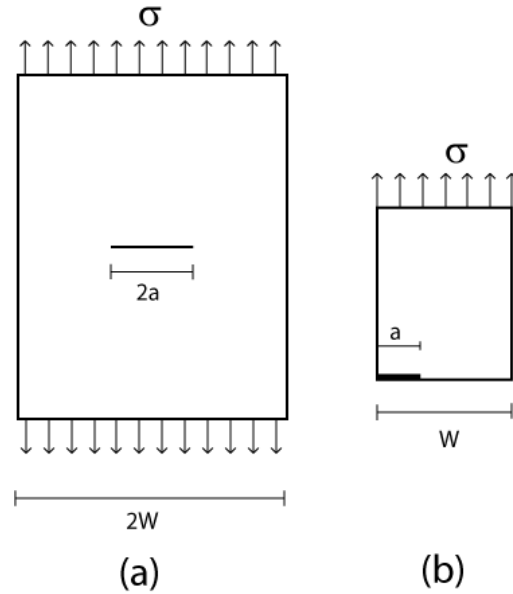


Figura 5.16: (a) El problema a resolver será el de una placa de ancho  $2W$  con una grieta en el centro de tamaño  $2a$ . (b) Dada la simetría, es posible resolver sólo  $1/4$  de la geometría.

CASO	Nodos	Elementos	$l$ [m]	$K[MPa\sqrt{m}]$	Error (%)
Caso 1	126	53	0,125	218,521	3,78
Caso 2	162	69	0,03125	215,076	2,24
Caso 3	492	225	0,03125	215,743	2,544
Caso 4	180	77	0,015625	214,57	2,01
Grieta explícita	732	344	0,0625	212,862	1,22
Solución analítica				210,255	0

Tabla 5.3: Resultados para el cálculo de  $K_I$  para la grieta centrada. Caso 1:  $l/a = 1/8$ , caso 2:  $l/a = 1/32$ , caso 3:  $l/a = 1/32$  (más refinado en otras zonas), caso 4:  $l/a = 1/64$ , grieta explícita:  $l/a = 1/16$

dards” (NAFEMS) del Reino Unido y es utilizado como referencia en el software ABAQUS<sup>3</sup>. Los valores utilizados son  $a = 0,005[m]$ ,  $\sigma = 100[MPa]$ ,  $E = 207[GPa]$  y  $\nu = 0,3$ , en un estado de deformación plana. Las dimensiones y las condiciones de borde aparecen en la figura 5.18. Para este caso, la solución es

$$K_I/K_0 = 2,74 \quad (5.2)$$

Donde  $K_0 = \sigma\sqrt{\pi a}$ .

La malla utilizada para los cálculos se muestra en la figura 5.19. Se probaron tres niveles de refinamiento, con mallas generadas con el software desarrollado utilizando el algoritmo LEPP-Delaunay, para el caso con simetría y sin simetría. Los resultados se comparan además con el resultado entregado por ABAQUS,

<sup>3</sup>Estos casos de prueba están disponibles en la documentación de ABAQUS, una vez instalado el software.

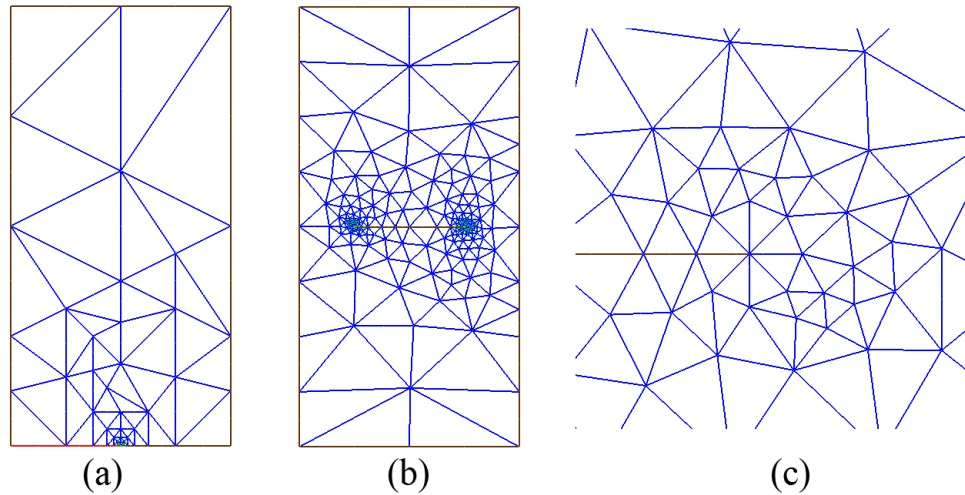


Figura 5.17: (a) Mallado utilizando 1/4 de la geometría, caso mayor refinamiento. (b) Malla utilizada para modelar la grieta sin simetría. (c) Detalle de la malla en la punta de la grieta para el caso sin simetría.

utilizando el método de la integral J y utilizando una malla de elementos isoparamétricos de 8 nodos con elementos especiales en la punta, como muestra la figura 5.20. Los resultados se muestran en la tabla 5.4

CASO	nodos	elementos	l [m]	$K_I/K_0$	Error( %)
Caso 1, s/simetría	455	208	0,0025	2,814	2,64
Caso 2, s/simetría	617	286	0,0003125	2,813	2,59
Caso 3, s/simetría	825	388	0,000078125	2,816	2,71
Caso 1, c/simetría	226	99	0,0025	2,827	3,08
Caso 2, c/simetría	325	146	0,0003125	2.807	2,39
Caso 3, c/simetría	355	158	0,000078125	2.811	2,53
Malla ABAQUS, c/simetría	777	238	0.0005556	2.795	1,94
Solución NAFEMS				2,74	0

Tabla 5.4: Resultados para el cálculo de  $K_I$  para la grieta en la muesca con forma de V. Caso 1:  $l/a = 1/2$ , caso 2:  $l/a = 1/16$ , caso 3:  $l/a = 1/64$ .

Como se observa en la tabla 5.4, Se obtienen errores del orden del 2 %. En el segundo refinamiento la dimensión característica del elemento de la punta de la grieta es  $0,0003125[mm]$  ( $l/a=1/16$ ), del mismo orden de magnitud del tamaño del elemento usado por la malla de ABAQUS, cuyo tamaño es  $0,0005556[mm]$ . Sin embargo la malla de triángulos tiene 325 nodos y 146 elementos, mientras que la malla de ABAQUS tiene 777 nodos y 238 elementos. El ahorro de nodos de la malla triangular se debe a que los algoritmos Lepp-Delaunay permiten realizar un refinamiento muy localizado, por lo que la transición de elementos pequeños a elementos grandes es más rápida. Esto es complicado de hacer con generadores de mallas de cuadriláteros.

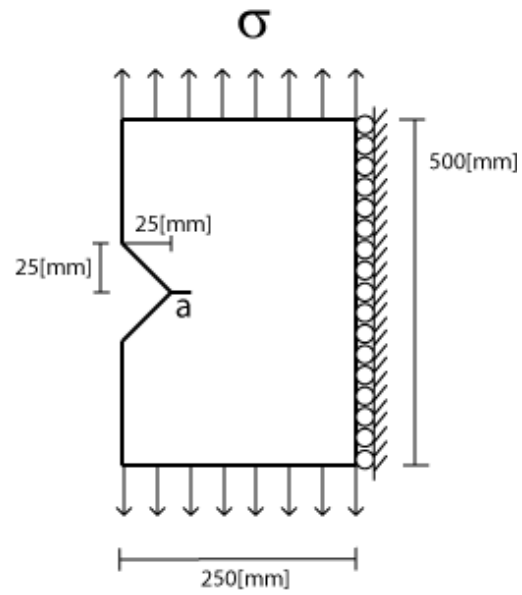


Figura 5.18: Configuración de una grieta ubicada en la punta de una muesca con forma de v, en un sólido en tensión.

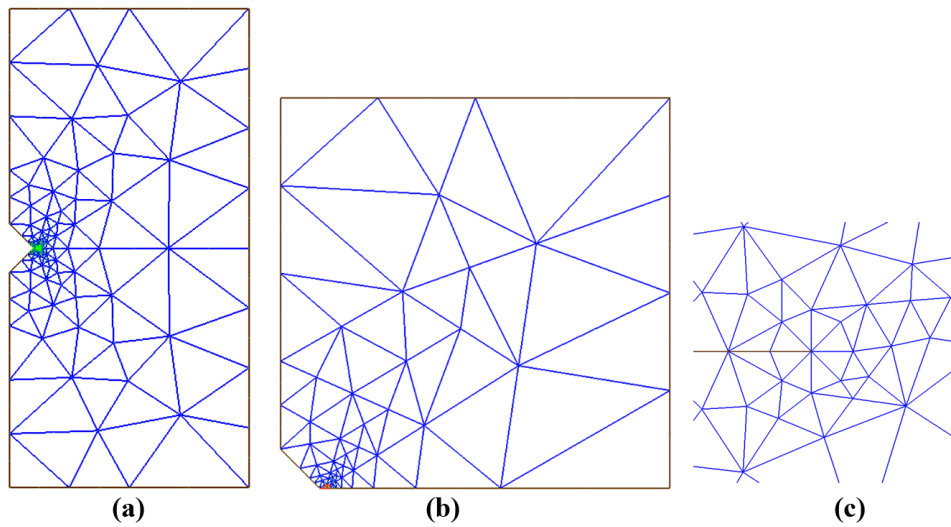


Figura 5.19: (a): Malla utilizada para los cálculos. (b): Malla para el caso con simetría. (c): detalle de la malla en la punta de la grieta, para el caso sin simetría.

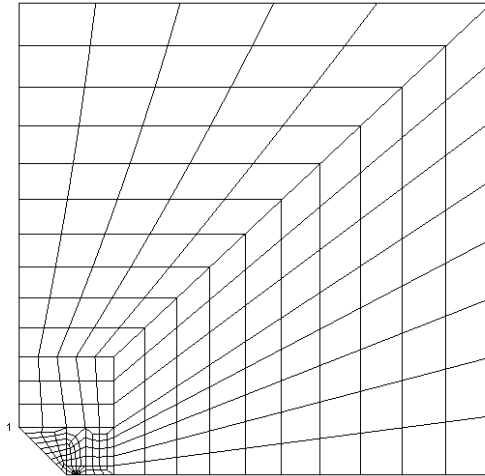


Figura 5.20: Malla isoparamétrica de 8 nodos generada por el software ABAQUS, que se utiliza para calcular  $K_I$  con el método de la integral J.

#### 5.3.4 Grieta en ángulo

El cuarto caso a probar es el de una grieta en ángulo en una placa, configuración que se muestra en la figura 5.21. Este caso también es propuesto por la NAFEMS. Los valores utilizados son  $a = 0,025$ ,  $\sigma = 100[MPa]$ ,  $E = 207[GPa]$  y  $\nu = 0,3$ . Las dimensiones aparecen en la figura 5.21. Para este caso, la solución es

$$\begin{aligned} K_I/K_0 &= 0,190 \\ K_{II}/K_0 &= 0,405 \end{aligned} \tag{5.3}$$

Donde  $K_0 = \sigma\sqrt{\pi a}$ .

Se probaron cuatro niveles de refinamiento, con mallas generadas con el software desarrollado. Una de las mallas utilizadas se muestra en la figura 5.22. Nuevamente se utiliza una roseta simétrica de 8 elementos, como se muestra en la figura 5.22(b). Los resultados se comparan además con el resultado entregado por ABAQUS al utilizar una malla de elementos isoparamétricos de 8 nodos con elementos especiales en la punta, utilizando el método de la integral J. La malla utilizada por ABAQUS se muestra en la figura 5.23. Los resultados se muestran en la tabla 5.5.

En este caso la diferencia de los resultados con el valor propuesto por la NAFEMS es del orden del 5%.

#### 5.4 Simulación del crecimiento de la grieta

El crecimiento cuasiestático de una grieta o la propagación por fatiga se modelan normalmente calculando en forma discreta su propagación, resolviendo incrementalmente un modelo de elementos finitos con una configuración de grieta dada, y encontrando la nueva configuración a partir de estos resultados. Por lo tanto en cada paso se tiene un nuevo modelo de elementos finitos a resolver. En la figura 5.24 se muestra un esquema de cómo está estructurada una metodología para simular la propagación de grietas. Esta metodología es

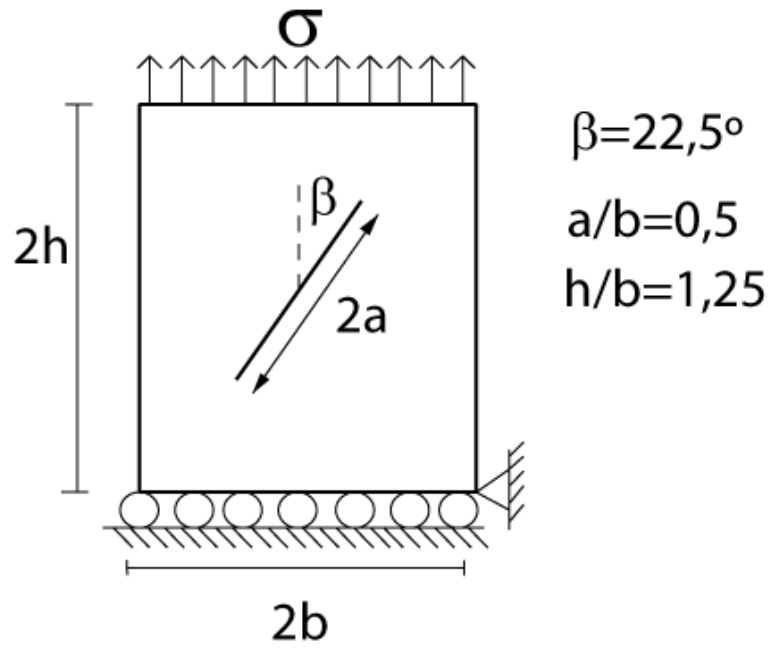


Figura 5.21: Configuración para la grieta en ángulo.

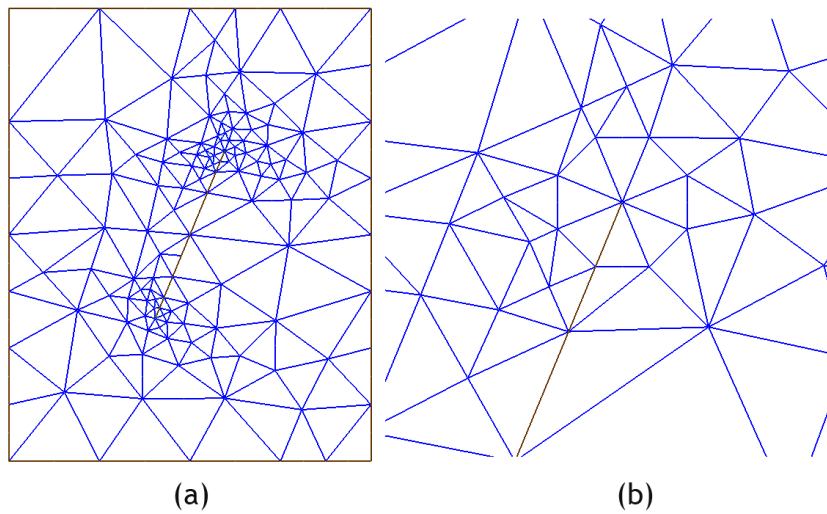


Figura 5.22: (a) malla generada por el software para el cálculo de los factores de intensificación de esfuerzos de una grieta en ángulo. (b) Detalle de la malla en la punta de la grieta.



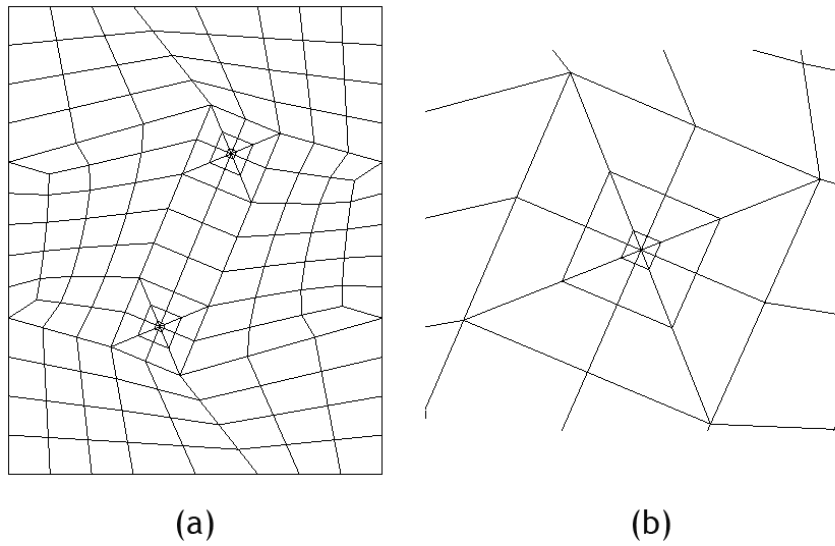


Figura 5.23: (a) malla generada por el software ABAQUS para calcular los factores de intensificación de esfuerzos en una grieta en ángulo. (b) Detalle de la malla en la punta de la grieta.

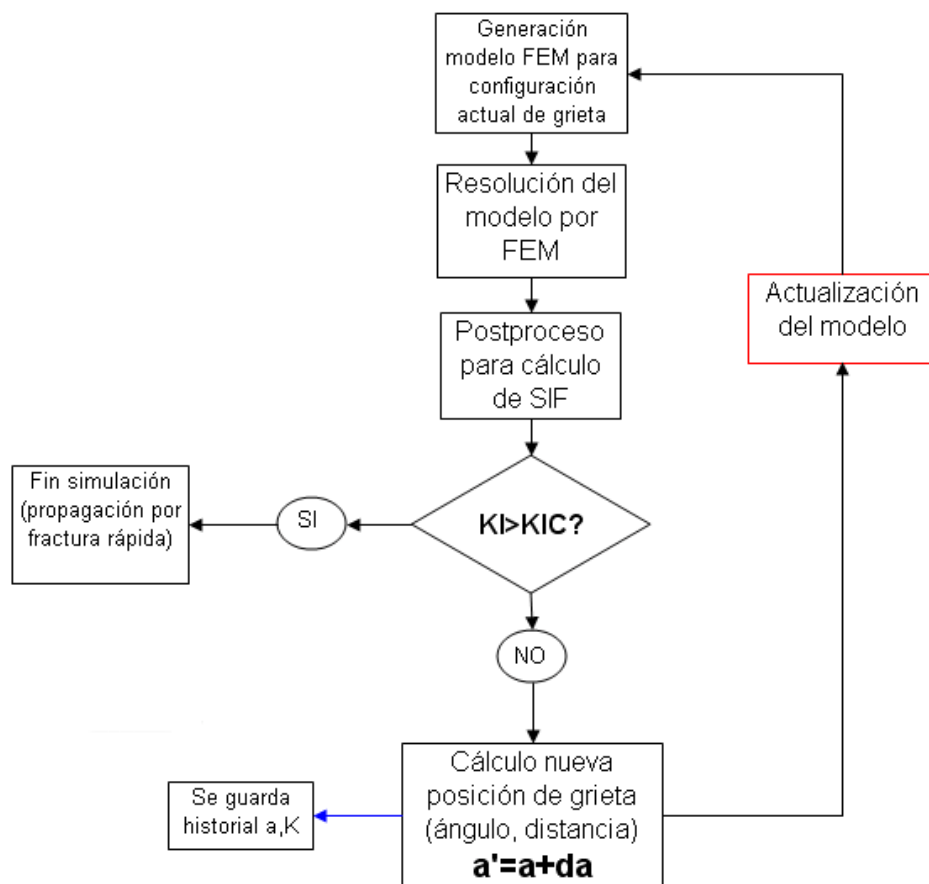


Figura 5.24: Esquema de una metodología para simular la propagación de una grieta, en forma secuencial.

CASO	nodos	elementos	l/a	$K_I/K_0(\text{error})$	$K_{II}/K_0(\text{error})$
Caso 1	339	153	1/4	0,196(-2,97)	0,402(0,67)
Caso 2	514	236	1/8	0,194(-2,28)	0,4(1,33)
Caso 3	975	459	1/75	0,183(3,57)	0,385(4,84)
Caso 4	960	452	1/128	0,183(3,5)	0,384(5,06)
Malla ABAQUS	586	170	22,5	0.185(-2,9)	0.405(0,1)
Solución NAFEMS				0,190	0,405

Tabla 5.5: Resultados para el cálculo de  $K_I$  y  $K_{II}$  para la grieta en ángulo.

válida para una propagación cuasiestática, o una propagación por fatiga. Para simular fractura dinámica existen metodologías más complejas que están fuera del alcance de esta tesis.

Este esquema general tiene variantes. Las diferencias de implementación radican principalmente en el paso  $da$  de crecimiento entre un paso y otro, y la forma en la que se actualiza el modelo entre un paso y otro (recuadro rojo en la figura 5.24).

En el caso de la simulación de la fatiga, el paso  $da$  está asociado a la ley de Paris (ecuación 2.44). Una primera aproximación consiste en fijar arbitrariamente el valor de  $da$  y mantenerlo fijo durante toda la simulación, o variarlo a criterio personal, y luego a partir de la ley de Paris deducir la cantidad de ciclos necesarios para obtener dicho avance  $da$  en cada paso. Esta metodología aunque es simple entrega buenos resultados con respecto a la vida a la fatiga en cargas cíclicas constantes, pero no es adecuada por ejemplo para simular una carga de amplitud variable, donde existen efectos más complejos sobre la vida a la fatiga debido a la variación de la carga (Castro *et al.*, 2003). Por otro lado, el valor de  $da$  podría calcularse automáticamente dependiendo de la curvatura que siga el avance de grieta, que está relacionado con el radio  $K_I/K_{II}$  en cada paso.

Con respecto a la actualización del modelo entre un paso y otro, la opción más elemental es rehacer completa la malla en cada paso, que es el método usado por ejemplo por Phongthanapanich & Dechaumphai (2004). Es una solución fácil de implementar y que entrega buenos resultados, pero tiene la desventaja de tener un alto costo computacional en el paso de generación de mallas, lo que se vuelve crítico en modelos grandes. En un caso real la zona afectada por la grieta es dimensionalmente pequeña en comparación con la geometría total, por lo que intuitivamente se puede concluir que deben buscarse metodologías que modifiquen sólo localmente la malla. Bittencourt *et al.* (1996) proponen un método donde la malla se rehace sólo en la zona cercana a la punta de la grieta, borrándola y haciéndola de nuevo, conservando el resto de la malla. Esto tiene la ventaja de reducir el costo de mallado, pero se generan pasos intermedios donde no existe una malla válida. En este trabajo se propone un método donde toda la malla es conservada, y sólo se agregan puntos si es necesario, a medida que la punta de la grieta avanza, y si se generan elementos demasiado pequeños éstos se eliminan con un proceso de desrefinamiento. Esto es posible gracias a los algoritmos basados en el LEPP, que permiten modificar una malla ya existente. Por otro lado, el refinamiento basado en el LEPP siempre mantiene una malla válida, lo que hace el método más robusto.

**Metodología propuesta** El proceso de simulación de un paso del crecimiento de la grieta, a grandes rasgos, se muestra en la figura 5.25. Primero se inserta una arista en la malla de triángulos, cuyo nodo inicial es la punta actual, y el nodo final es la nueva posición de la punta de la grieta, que está fijada por los cálculos de los factores de intensificación de esfuerzos. Esta arista corresponde al nuevo segmento de la grieta. La inserción es Delaunay. Luego se transforma la arista en una grieta, duplicándola y duplicando el nodo correspondiente a la punta anterior. Finalmente se genera la roseta simétrica con la metodología mostrada en la figura 5.11.

Como se mostró en el paso final de la metodología de la figura 5.25, la generación de la roseta por medio de la inserción Delaunay o desplazamiento de puntos genera elementos de mala calidad, por eso en forma posterior se debe incluir un paso de mejoramiento de la malla. Dado que a medida que la grieta avanza se va acercando a zonas donde el tamaño de los elementos es mucho mayor que la roseta, al utilizar los algoritmos Lepp-Delaunay además de mejorar los elementos, éstos se refinan, generándose en todo momento una transición suave desde los elementos de la roseta hacia los elementos más grandes, siempre con triángulos de buena calidad.

Se probaron varios casos para estudiar el funcionamiento de la metodología propuesta. Los resultados se muestran a continuación.

#### 5.4.1 Placa con grieta en la orilla

Para probar la metodología de propagación de grietas se resuelve el problema mostrado en la figura 5.26. Consiste en una placa con una grieta en un borde, fija en el borde de abajo y con una carga en el borde de arriba. Esta configuración tiene una solución analítica, que relaciona el factor de intensificación de esfuerzos con el largo de la grieta (Tada *et al.*, 1985), para un crecimiento cuasiestático

$$K_I(a) = \frac{\sqrt{2w \tan(\beta)}}{\cos(\beta)} (0,752 + 2,02\alpha + 0,37(1 - \sin(\beta))^3) \quad (5.4)$$

donde  $\alpha = a/w$  y  $\beta = \alpha\pi/2$ ,  $a$  y  $w$  el largo de la grieta y el ancho de la placa, respectivamente.

El problema fue resuelto utilizando la metodología explicada anteriormente. En el gráfico 5.27 se muestra la solución analítica, la solución utilizando la metodología mostrada en la figura 5.25, y una solución usando la misma metodología pero sin generar la roseta simétrica, sino con elementos irregulares como los de la figura 5.15(b) o (c).

Como se observa en el gráfico 5.27, al usar la roseta simétrica en la punta se obtienen mucho mejores resultados que sin la roseta. El paso de crecimiento y el tamaño característico de la roseta es de 0,11. Sin usar la roseta simétrica, es decir refinando y dejando elementos irregulares, se obtienen errores variables que llegan hasta 10%. En algunos pasos el error era menor. El error dependía de cuántos elementos se generaban en la punta de la grieta, mientras más fuesen, menor era el error. Por lo tanto utilizar rosetas irregulares no es una metodología muy confiable. Además se observa al final de la curva que la solución diverge, esto se debe a que al tener los cálculos previos mucho error, se genera un ángulo de propagación errado que hace diverger la solución. Ya se mostró en el capítulo 5.3.1 que incluso al usar un tamaño característico menor, el error no disminuye a niveles aceptables.

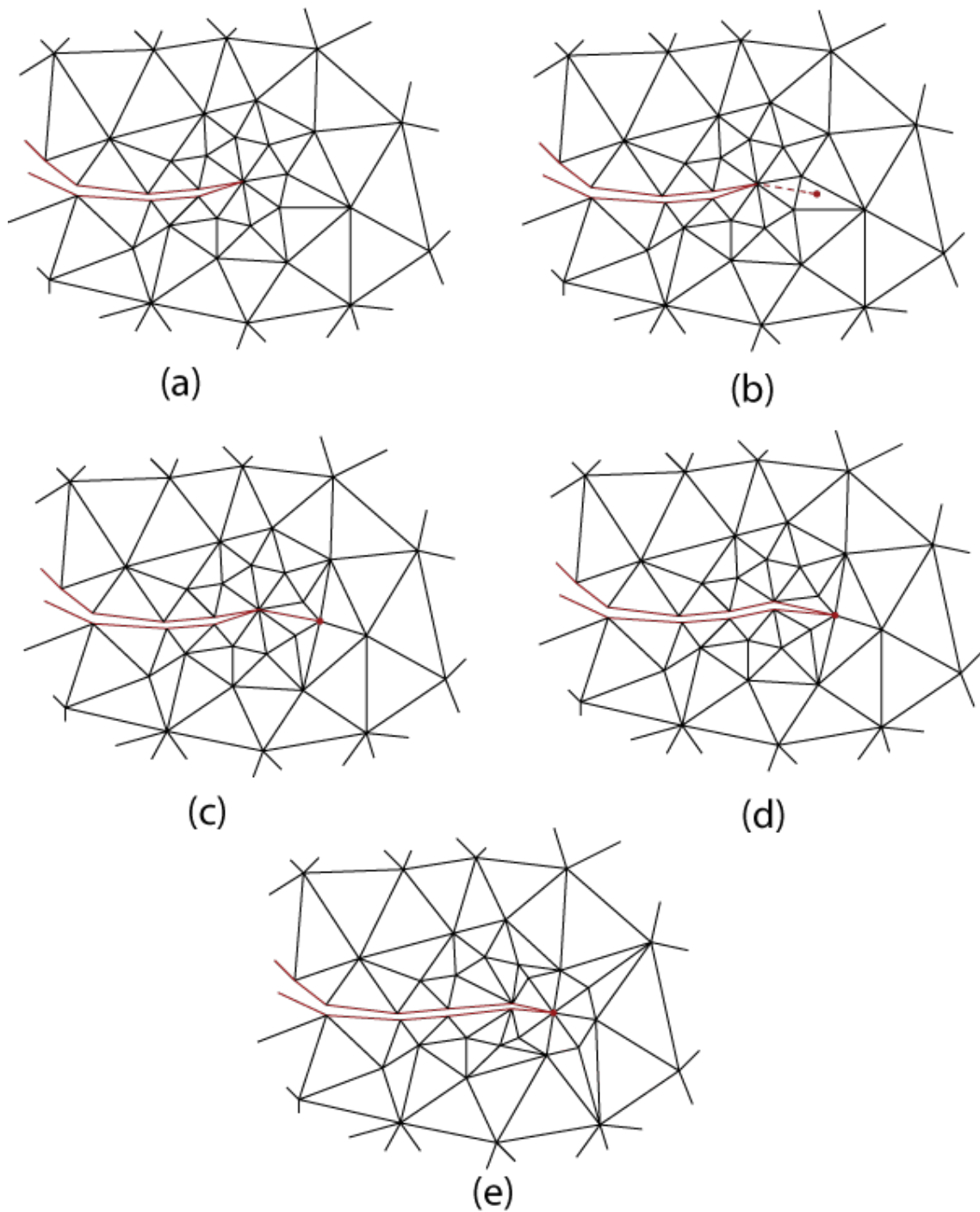


Figura 5.25: Procedimiento para simular el crecimiento de la grieta. (a) Grieta inicial. (b) Se fija la posición de la nueva punta de la grieta, según los cálculos de los factores de intensificación de esfuerzos. (c) Se inserta una arista, que corresponde al nuevo segmento de grieta. (d) Se duplica la arista y el nodo correspondiente a la antigua punta de la grieta. (e) Se genera la roseta simétrica.

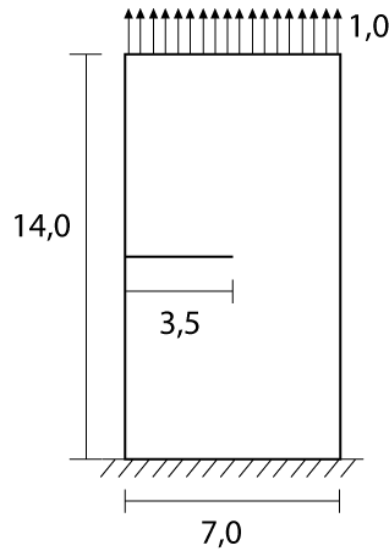


Figura 5.26: Problema a resolver para simular el crecimiento de la grieta. Este problema tiene solución analítica.

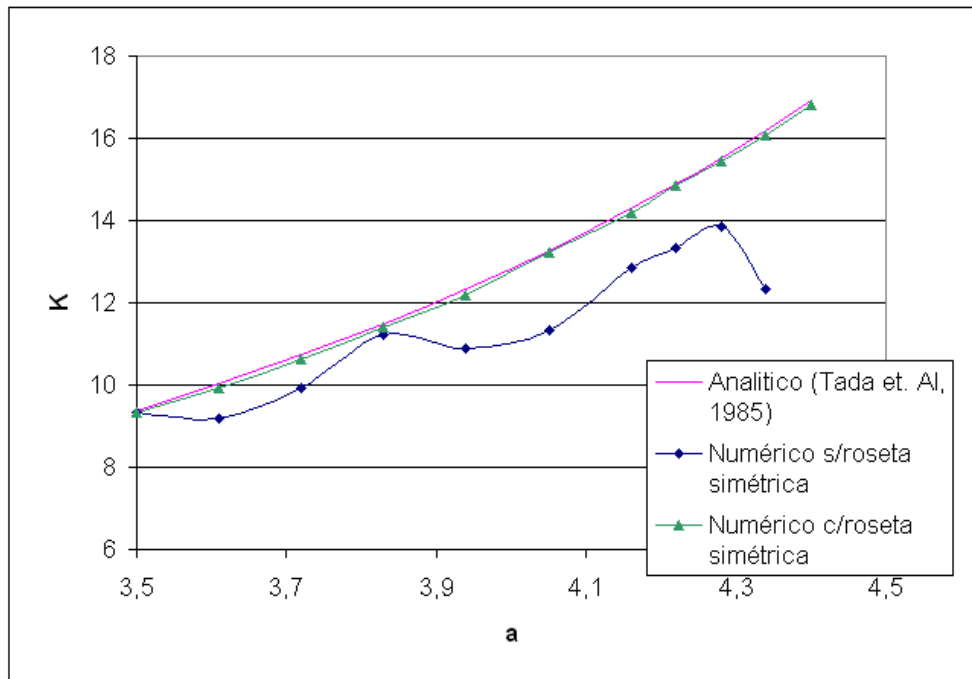


Figura 5.27: Resultados de la simulación de la propagación de una grieta en una placa con grieta en el borde. Los resultados se comparan con una solución analítica.

Por otro lado, al usar una roseta simétrica se obtienen muy buenos resultados, obteniendo un error por debajo del 1% en todos los pasos. Cabe destacar que el paso de avance y el tamaño característico de los elementos de la roseta son iguales.

Como se mostró en la figura 5.11, el método de propagación utiliza el algoritmo Lepp-Delaunay para mejorar los elementos después de generar la roseta simétrica. En las figuras 5.28, 5.29, 5.30 y 5.31 se muestran las mallas generadas en los pasos 1, 4, 8 y 11 respectivamente. Se puede apreciar que a medida que la grieta avanza, se genera una gran cantidad de triángulos pequeños. Éstos son generados por el algoritmo de mejoramiento. El problema de dichos triángulos es que se acumulan a medida que la grieta se propaga, y cuando están lejos de la punta de la grieta son innecesarios. El criterio usado para mejorar era el de aplicar el algoritmo Lepp-Delaunay sobre cualquier triángulo con un ángulo menor  $> 25$ . En la tabla 5.6 se muestra la cantidad de nodos y elementos de cada paso.

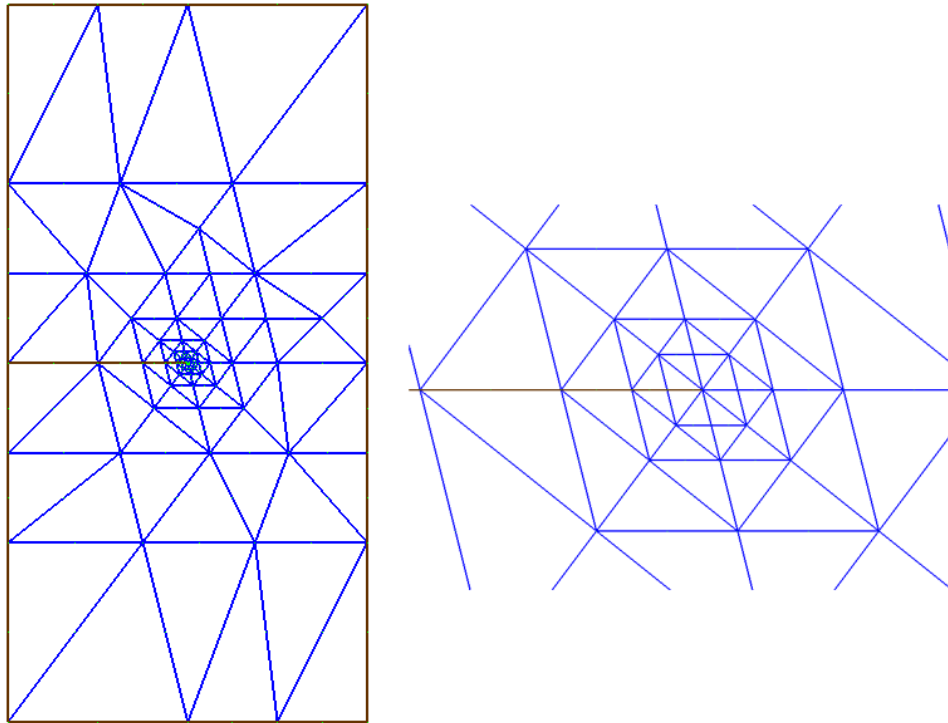


Figura 5.28: Malla del modelo de crecimiento de la grieta, para el paso 1.

#### 5.4.2 Probeta apoyada en 4 puntos con grieta simple

El siguiente caso consiste en un ensayo estándar de una probeta apoyada en cuatro puntos con una grieta centrada, como se muestra en la figura 5.32.

A esta probeta se le agrega una perforación circular con el fin de provocar una curvatura en el camino que sigue la grieta. Los parámetros del problema son  $E = 205[Gpa]$ ,  $\nu = 0,3$ ,  $P = 100[N]$  y  $a = 2,5[mm]$ . Se modela un estado de deformación plana. Los resultados se comparan con valores experimentales y con valores numéricos para exactamente el mismo problema, reportados por Castro *et al.* (2003).

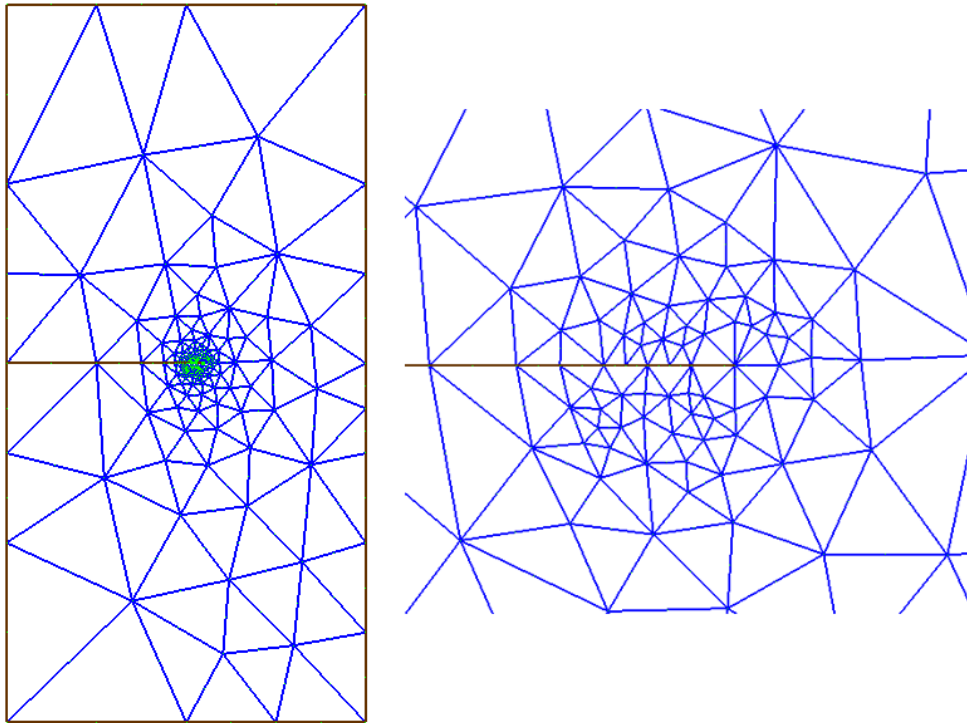


Figura 5.29: Malla del modelo de crecimiento de la grieta, para el paso 4.

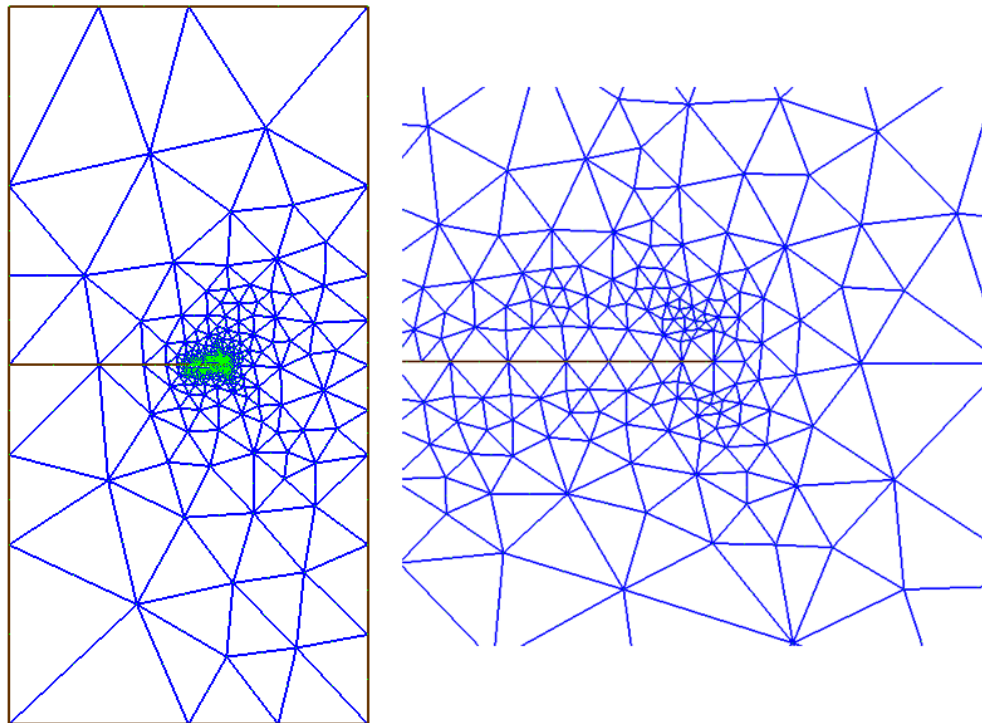


Figura 5.30: Malla del modelo de crecimiento de la grieta, para el paso 8.

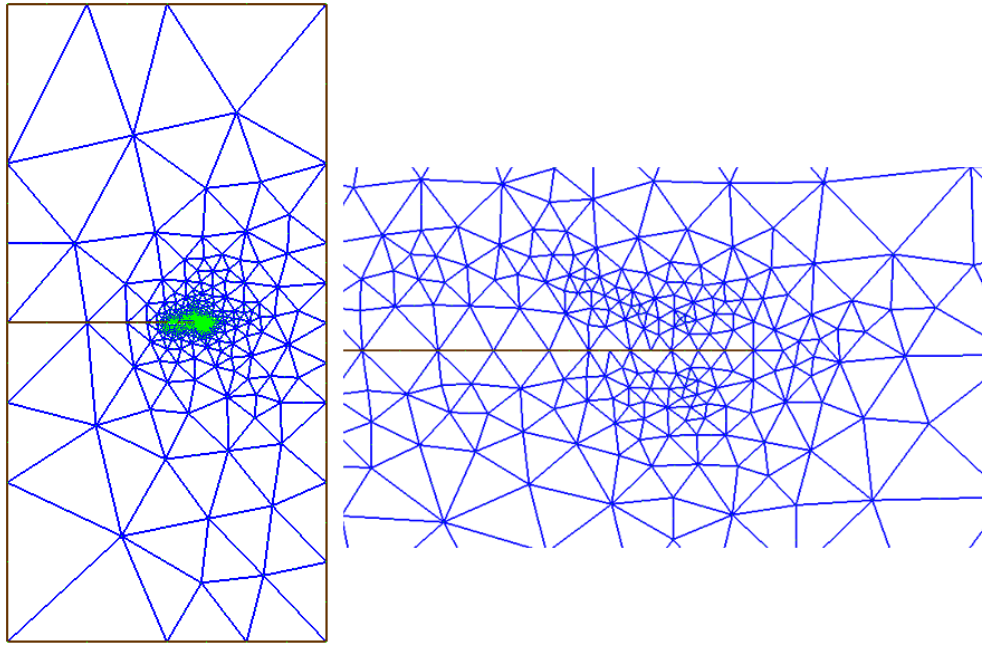


Figura 5.31: Malla del modelo de crecimiento de la grieta, para el paso 11.

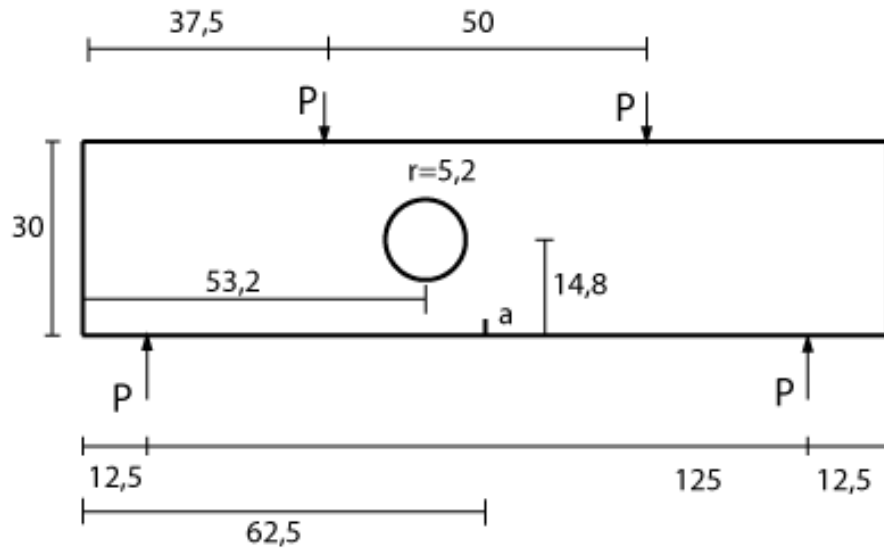


Figura 5.32: Probeta apoyada en cuatro puntos con una grieta centrada. Se incluye un agujero para provocar una curvatura en el camino a seguir por la grieta.  $a = 2,5$  (unidades en mm).



PASO	nodos	elementos
1	259	114
2	400	183
3	500	231
4	565	262
5	679	318
6	860	407
7	968	459
8	1134	541
9	1251	598
10	1372	657
11	1496	717

Tabla 5.6: Número de nodos y elementos en cada uno de los pasos de la propagación de la grieta del borde.  $l/a=1/64$ .

En la figura 5.33 se muestra el resultado después de 20 pasos, pero sin el proceso de mejoramiento entre una iteración y otra, para explicitar la necesidad no sólo de mejorar la malla después de crear la roseta (lo que podría hacerse, por ejemplo, sólo con un suavizamiento laplaciano) sino de refinar la malla a medida que la grieta avanza, dado que la punta se mueve hacia zonas donde originalmente la malla tiene elementos mucho más grandes que la roseta.

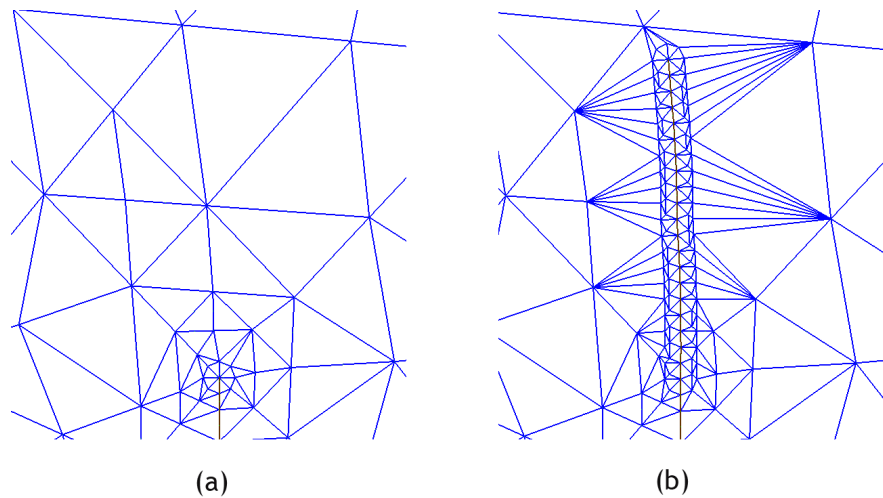


Figura 5.33: (a) Configuración inicial de la grieta. (b) Propagación de la grieta después de 20 pasos, sin mejorar la malla entre un paso y otro.

El mismo problema ahora se simula realizando un mejoramiento por medio del algoritmo Lepp-Delaunay entre un paso y otro. El algoritmo se aplica sobre todos los triángulos cuyo ángulo menor sea mayor que  $15^\circ$ . El tamaño de los elementos de la roseta es de  $l = 0,625[mm]$  ( $l/a = 1/16$ ). Además entre un paso y otro se aplica el algoritmo de desrefinamiento descrito en el capítulo 5.2.3

La figura 5.34 muestra la malla inicial, luego de 20, 50 y 92 pasos, respectivamente. La cantidad de nodos para cada una de ellas es 1278, 1738, 2518 y 3448 respectivamente. En la figura 5.35 se muestra el detalle de la malla en el paso 92. Se observa ahora que al usar un criterio para el refinamiento con el algoritmo Lepp-Delaunay de  $15^\circ$  se generan menos triángulos pequeños y aún se conserva una malla de buena calidad, por lo que este criterio es razonable para simular la propagación.

**Prueba sin desrefinamiento** Para efectos de comparación, se resuelve el mismo problema pero sin usar el método de desrefinamiento entre un paso y otro. Para las iteraciones 20, 50 y 92 se obtienen 1866, 2826 y 4043 nodos respectivamente. Al comparar con las mallas del caso con desrefinamiento (1738, 2518 y 3448 nodos) el ahorro en nodos es de 6,85 %, 12,23 % y 14,71 % respectivamente. El camino obtenido con el método sin desrefinamiento es exactamente el mismo. Por lo tanto el ahorro en nodos debido al paso de desrefinamiento es considerable, y aumenta a mayor cantidad de iteraciones. En la figura 5.36 se muestra un detalle de la malla obtenida con y sin desrefinamiento. El criterio aplicado es el de eliminar todos los pares de triángulos cuya arista común sea menor que  $2/3l$ , con  $l$  el largo característico de la roseta.

**Comparación con resultados experimentales** En la figura 5.37 se puede ver el resultado experimental reportado por Castro *et al.* (2003) y el camino calculado por el software desarrollado. Se puede ver que se obtiene un muy buen ajuste.

### 5.4.3 Probeta con dos apoyos y tres agujeros

A continuación se prueba el modelamiento hecho para la configuración que se muestra en la figura 5.38. Este caso fue probado numéricamente por Phongthanapanich & Dechaumphai (2004) y numérica y experimentalmente por Bittencourt *et al.* (1996), lo que permitirá validar los resultados.

Esta probeta tiene tres agujeros, cuyo objetivo es curvar el camino de propagación de la grieta. Dependiendo de la configuración inicial de  $a$  y  $b$  se obtienen distintos caminos. Se probarán dos casos con los valores de  $a$  y  $b$  mostrados en la figura 5.38. Para ambos casos,  $P = 1[lbf]$ ,  $E = 29 \cdot 10^6 [PSI]$  y  $\nu = 0,3$ .

**Caso I** En la figura 5.39 se muestra la malla inicial y la malla final después de 110 iteraciones, para el primer caso. La malla inicial tiene 2226 nodos y 1042 elementos, y la malla final tiene 4496 nodos y 2066 elementos. Entre un paso y otro se aplica el algoritmo Lepp-Delaunay para todo triángulo cuyo ángulo menor sea mayor que  $15^\circ$ . Este caso se dividió en dos partes. Para las primeras 15 iteraciones se utiliza una roseta con un tamaño característico de  $l = 0,09375$  ( $l/a = 1/16$ ). La malla obtenida después de las 15 iteraciones se muestra en la figura 5.40. Luego de estas iteraciones la punta de la grieta está muy cerca de uno de los agujeros, y los elementos del agujero son más pequeños que la roseta, por lo que si se continúa con la simulación se obtendrán malos resultados, así que a partir de esta iteración se utiliza una roseta con tamaño característico  $0,0234375$  ( $l/a = 1/64$ ). En la figura 5.41 se ven detalles de la malla después de 60 y 100 iteraciones, respectivamente.

**Caso II** En la figura 5.42 se muestra la malla inicial y la malla final después de 72 iteraciones, para el segundo caso. La malla inicial tiene 1950 nodos y 906 elementos, y la malla final tiene 3332 nodos y 1534

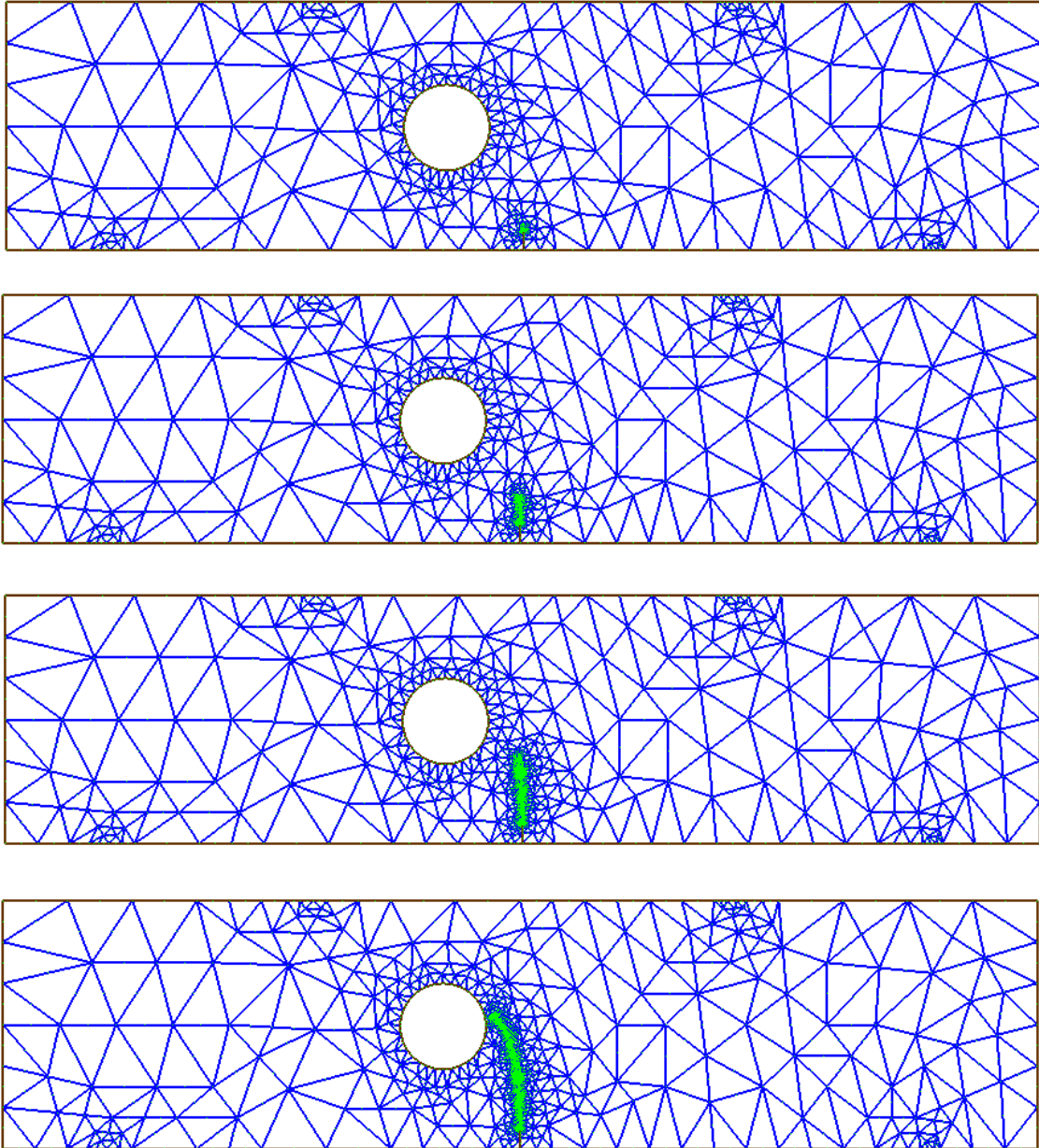


Figura 5.34: De arriba hacia abajo: Malla para 0, 20, 50 y 92 iteraciones, respectivamente.

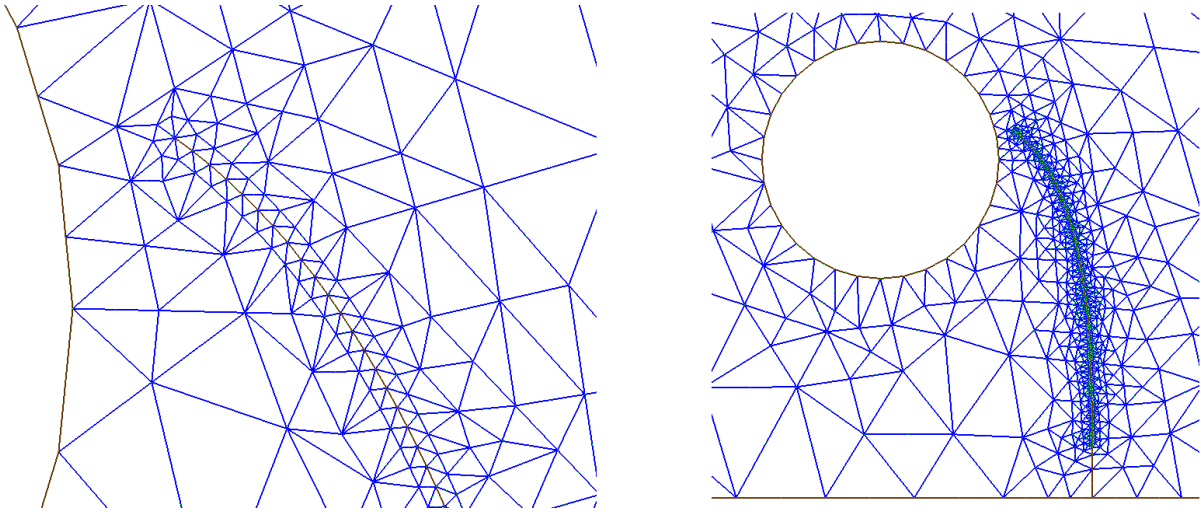


Figura 5.35: Detalles de la malla para la iteración 92.

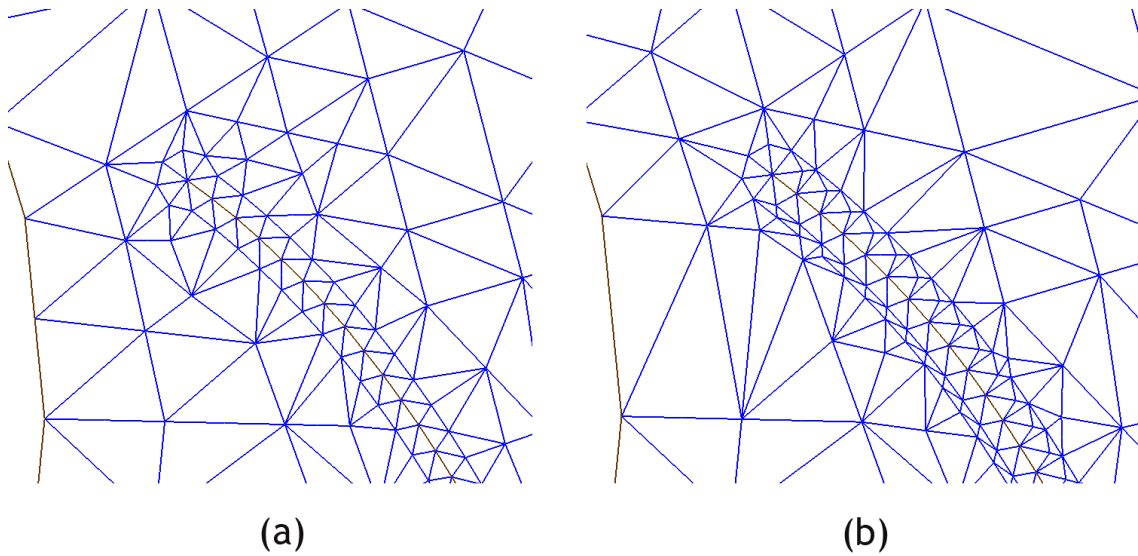


Figura 5.36: (a) Detalle de la malla sin desrefinamiento. (b) El mismo caso pero con desrefinamiento. El criterio para desrefinar es eliminar todos los pares de triángulos cuya arista común sea menor que  $2/3l$ , con  $l$  el largo característico de la roseta.

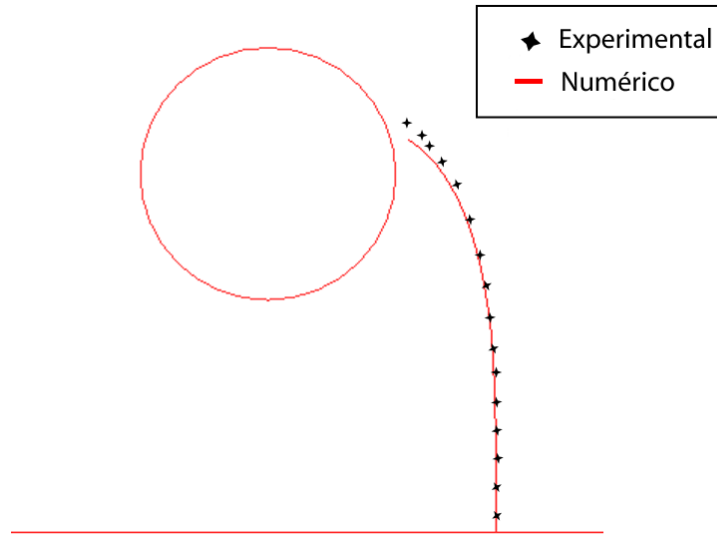


Figura 5.37: Comparación entre el resultado experimental y el camino calculado por el software para el caso de la probeta cargada en 4 puntos.

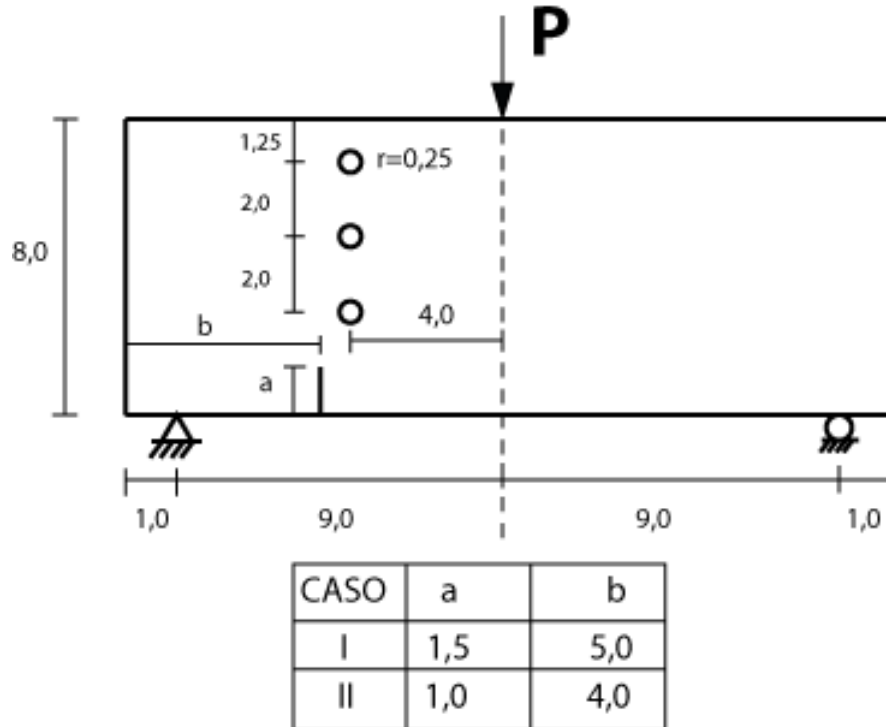


Figura 5.38: Probeta apoyada en dos puntos, con carga y grieta centrada y tres agujeros (unidades en pulgadas).

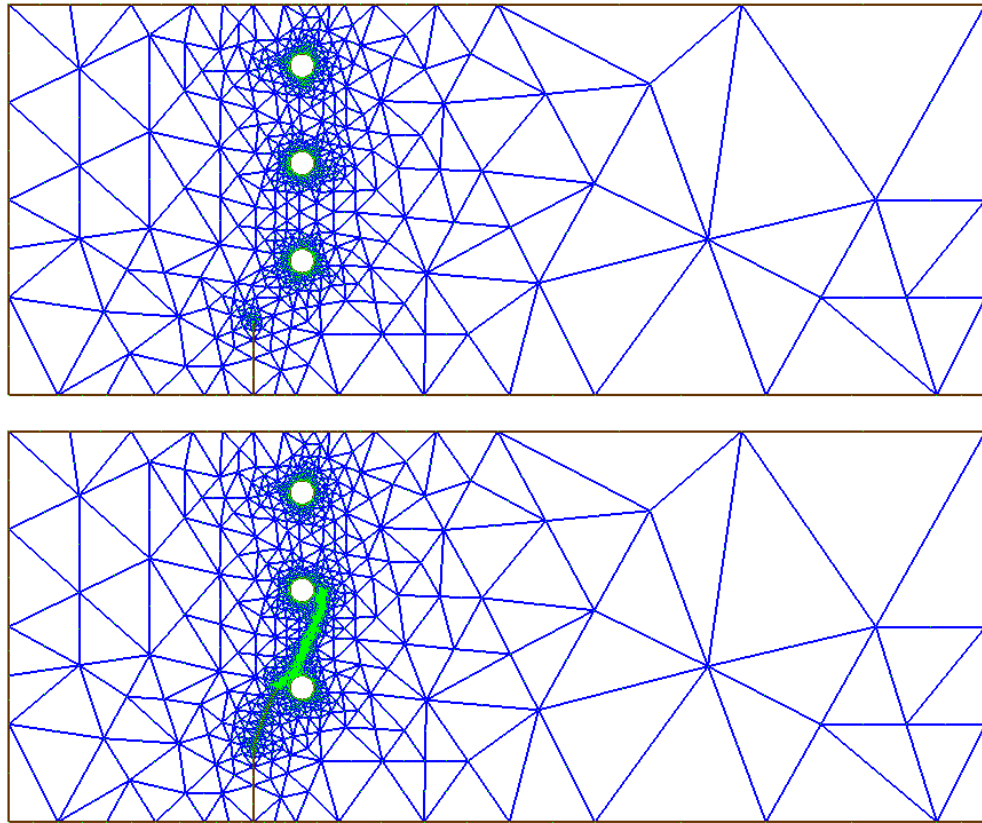


Figura 5.39: Arriba: Malla inicial del caso de la probeta con tres agujeros, caso I. Abajo: Malla final, después de 110 iteraciones.

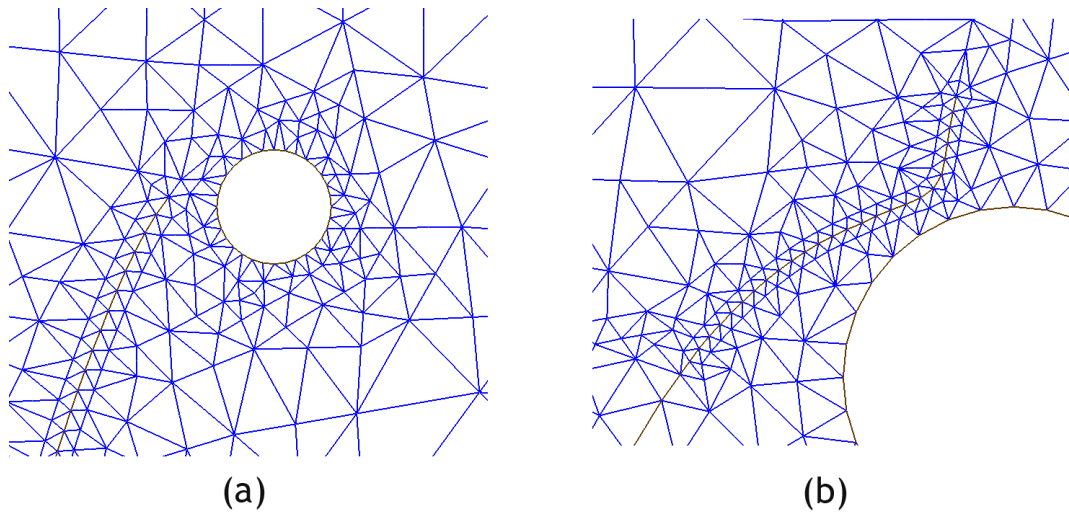


Figura 5.40: (a) Malla después de 15 iteraciones, usando un tamaño característico  $l/a = 1/16$ . (b) Al acercarse mucho al agujero, se cambia el tamaño característico a  $l/a = 1/64$ . Malla después de 40 iteraciones.

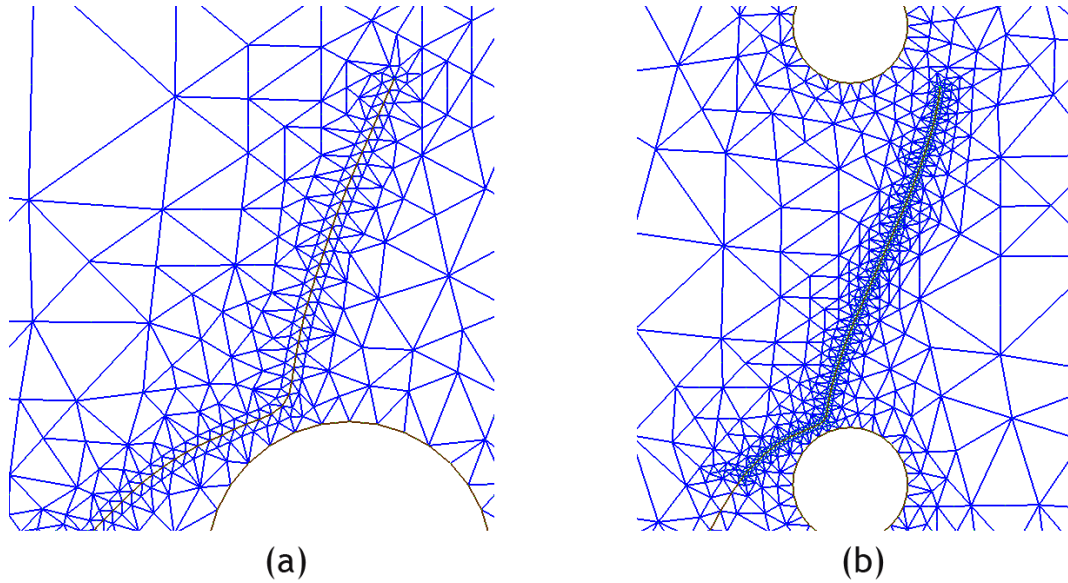


Figura 5.41: (a) Detalle de la malla después de 60 iteraciones. (b) Detalle de la malla después de 100 iteraciones.

elementos. Para este caso se utilizó el mismo tamaño de elemento  $l = 0,0625$  ( $l/a = 1/16$ ) para todas las iteraciones. Entre un paso y otro se aplica el algoritmo Lepp-Delaunay para todo triángulo cuyo ángulo menor sea mayor que  $15^\circ$ . En la figura 5.43 Se ve un detalle de la malla final.

**Comparación con valores experimentales** En las figuras 5.44 y 5.45 se compara el camino calculado por el software con las mediciones experimentales para los casos *I* y *II* de la probeta con tres agujeros, respectivamente. Se puede apreciar que existe un muy buen ajuste para los dos casos estudiados.

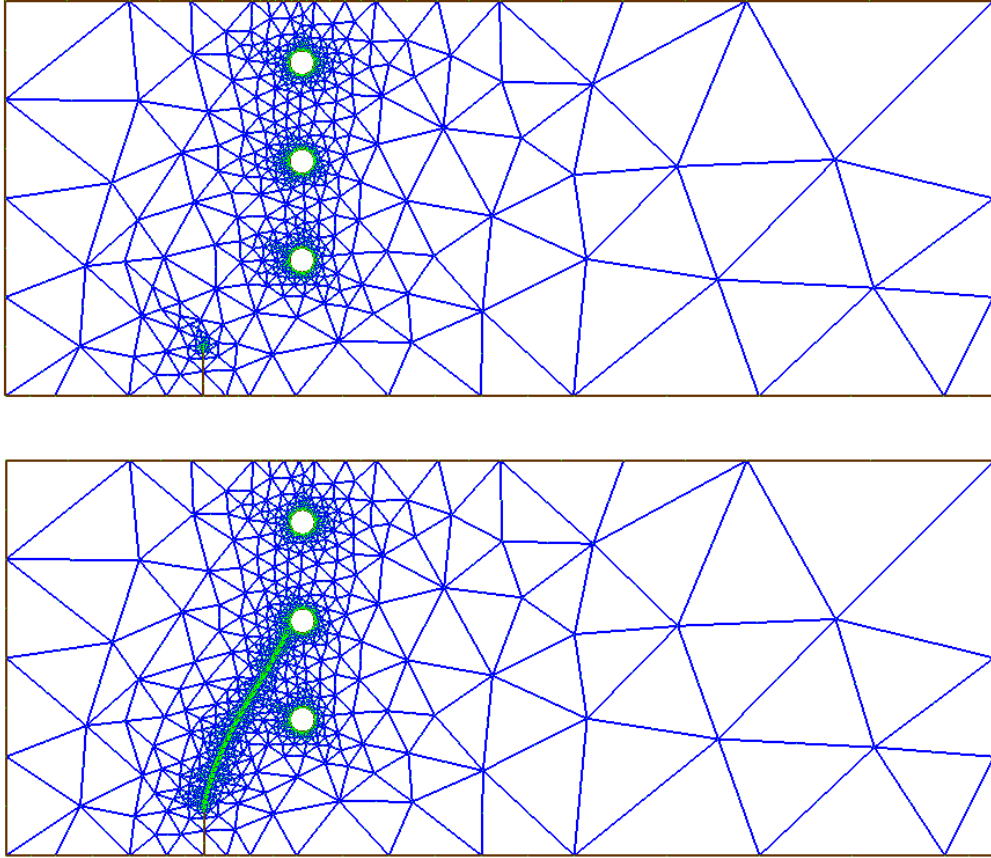


Figura 5.42: Arriba: Malla inicial Para el caso II de la probeta de tres agujeros. Abajo: Malla final, después de 72 iteraciones.

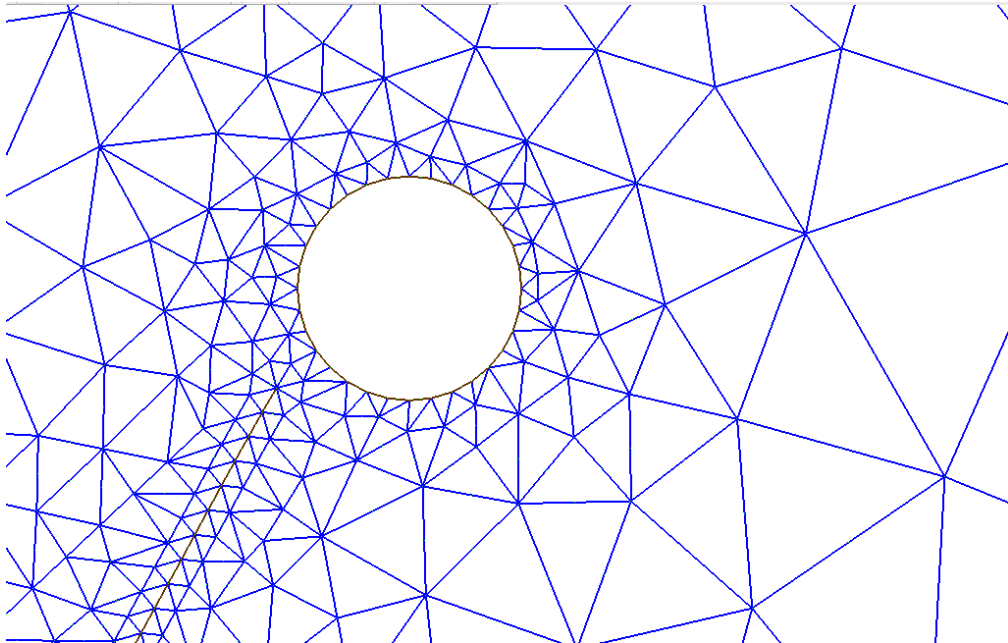


Figura 5.43: Detalle de la malla final para el caso II.



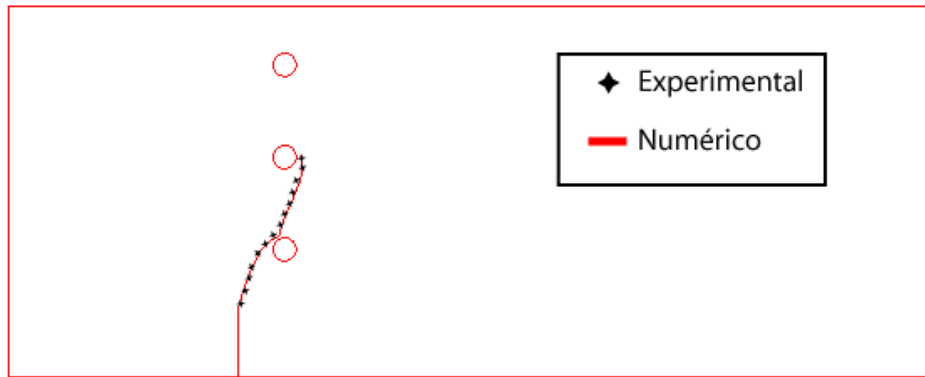


Figura 5.44: Comparación entre resultados experimentales y numéricos para la probeta de tres agujeros, caso I.

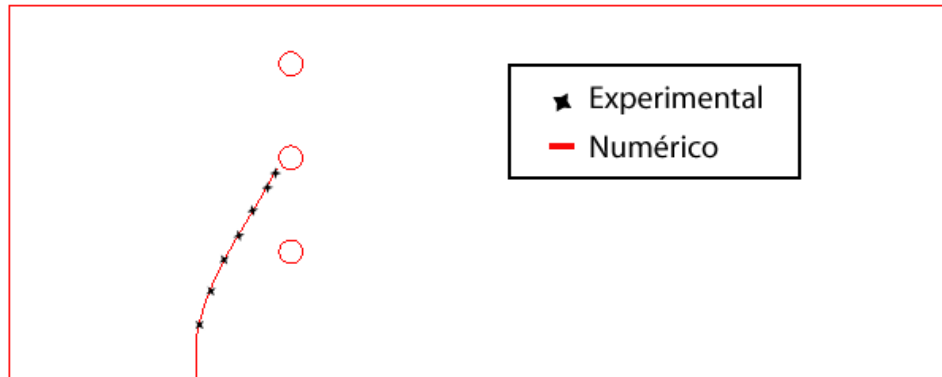


Figura 5.45: Comparación entre resultados experimentales y numéricos para la probeta de tres agujeros, caso II.

## Capítulo 6

### Conclusiones

Se implementó una metodología numérica que permite simular la propagación cuasiestática o por fatiga de una grieta arbitraria en una geometría arbitraria 2D. Para calcular los parámetros de fractura se utilizó el método de extrapolación de desplazamientos y para calcular el ángulo de propagación se utilizó el método del máximo esfuerzo tangencial. El proceso de propagación se resuelve iterativamente por medio del método de elementos finitos, generándose entre un paso y otro una nueva malla geométrica a partir de la anterior utilizando los algoritmos de refinamiento Lepp-Delaunay, que permiten refinar y mejorar una malla ya existente sin rehacerla.

Todo lo anterior se incluyó en una aplicación con una interfaz gráfica para generar modelos de elementos finitos de geometrías arbitrarias 2D que incluyen defectos lineales como son las grietas. La aplicación desarrollada puede interactuar con el software comercial ABAQUS por medio de scripts implementados en el lenguaje Python.

A continuación se presentan las conclusiones obtenidas del presente trabajo.

#### Desarrollo del software

La implementación del software se hizo a partir de otro ya existente (Calderón & Rivara, 2004) y que estaba orientado al refinamiento de mallas. Dicha aplicación utiliza los algoritmos Lepp-Delaunay para el mejoramiento/refinamiento de mallas de triángulos.

Con respecto a la generación de mallas, se agregó a este software la posibilidad de generar mallas para geometrías arbitrarias con orificios y grietas, un método de suavizamiento laplaciano, métodos para duplicar aristas y nodos para generar grietas 1D y métodos para generar una roseta en la punta de una grieta.

Además se implementó una metodología que permite cargar un archivo en formato IGES con una geometría 2D compuesta de rectas y arcos de círculo.

Junto con lo anterior, se agregó al software la posibilidad de generar un modelo de elementos finitos, definiéndose propiedades de material, condiciones de borde y condiciones de carga.

Finalmente el software permite guardar la información del modelo en un archivo en formato 'inp', que puede ser cargado y resuelto por el software comercial ABAQUS.

La implementación del software se consideró exitosa, pudiéndose generar y resolver variados problemas con distintos tipos de condiciones de borde (simetría, apoyo simple, apoyo doble, empotramiento) y de carga (cargas normales, de corte y cargas puntuales), y para geometrías de variada complejidad.

Además el software funcionó adecuadamente en la simulación de la propagación de una grieta, resolviendo en forma automatizada este tipo de problemas. Se probaron varios casos con solución conocida de distinta complejidad tanto en la generación del modelo inicial (complejidad de la geometría) como en el comportamiento de la grieta (complejidad del camino seguido por la grieta) obteniendo buenos resultados en todos los casos, en lo que respecta a generación de mallas, interacción de datos con el software Abaqus y funcionamiento de los módulos de post-proceso.

### **Cálculo de los factores de intensificación de esfuerzos**

La metodología implementada para el cálculo de los factores de intensificación de esfuerzos (método de extrapolación de desplazamientos) entregó buenos resultados, obteniéndose errores del orden del 1 – 2% en los ejemplos probados.

En los problemas de mecánica de la fractura la diferencia de tamaño entre un elemento cercano a la punta de la grieta y un elemento en las zonas más alejadas puede ser de varios órdenes de magnitud. En esta tesis, las mallas utilizadas para resolver estos problemas se generaron utilizando algoritmos de refinamiento Lepp-Delaunay. Estos algoritmos generan mallas particularmente adecuadas para este tipo de problemas, pues se genera una transición suave desde los elementos pequeños de la punta de la grieta hasta los elementos más grandes en zonas alejadas sin necesidad de métodos complementarios como un suavizamiento laplaciano o quad-tree.

El uso de los algoritmos Lepp-Delaunay significa una ventaja en comparación con el uso de mallas de cuadriláteros en lo que respecta a la transición de elementos. Al ser más difícil este tipo de transiciones en mallas de cuadriláteros, en éstas se necesitan muchos más elementos para un mismo nivel de error que en una malla de triángulos. Esto se vuelve particularmente notorio cuando se desea modelar geometrías complejas con varias singularidades, que es la regla más que la excepción en la ingeniería.

Al comparar los algoritmos Lepp-Delaunay con otros algoritmos de generación de mallas de triángulos existe una ventaja en la eficiencia de la generación de mallas pues se puede obtener una malla de buena calidad sin necesidad de métodos complementarios ni post-procesos de la malla.

### **Propagación de grietas**

La metodología propuesta para la simulación de la propagación de grietas funcionó adecuadamente. La inserción Delaunay y los algoritmos Lepp-Delaunay permiten generar una malla nueva a partir de la malla del modelo anterior modificándola sólo localmente, lo que permite un ahorro en el costo computacional de generación de mallas. La modificación consiste en la inserción secuencial Delaunay de puntos siguiendo la lógica del algoritmo Lepp, lo que al hacerse sólo en la zona cercana a la punta de la grieta minimiza la inserción de puntos. Por esto, el aumento del tamaño de la malla entre un paso y otro es marginal, y mientras más grande sea el modelo, menor es el incremento relativo en la cantidad de nodos y elementos.

Los algoritmos Lepp-Delaunay tienen interesantes ventajas en problemas de propagación de grietas. Cuando se refina con estos algoritmos, se genera una transición suave del tamaño de los elementos a través de la malla. Por lo tanto, al simular la propagación de una grieta, a medida que ésta avanza a zonas donde la malla inicial tenía elementos de mucho mayor tamaño que los de la punta de la grieta se va generando al mismo tiempo elementos más pequeños y una transición suave con los ya existentes, manteniendo la calidad y sin necesidad de un post-proceso de la malla.

Se compararon los resultados obtenidos con la metodología presentada en esta tesis con resultados experimentales y numéricos reportados en diversas publicaciones. Se obtuvieron muy buenos resultados en la predicción del camino a seguir con la grieta, con una cantidad de nodos similar o menor a la de un método numérico alternativo.

Para el cálculo del factor de intensificación de esfuerzos es necesario un tipo de elementos especiales en la punta de la grieta, que mejora considerablemente el cálculo en comparación con una malla irregular de elementos normales. Se implementó una metodología para generar una “roseta” de ocho elementos en la punta de la grieta para obtener cálculos adecuados. La roseta se obtiene por medio de inserciones de nodos nuevos y desplazamiento de nodos existentes. Al hacer esto, se generan elementos de mala calidad cerca de la punta de la grieta. Por esto se implementó una metodología de eliminación de triángulos malos que consiste en eliminar pares de triángulos que comparten una arista demasiado pequeña. Se comparó un caso de propagación de grietas sin utilizar esta metodología con el mismo caso pero utilizando esta metodología y se obtuvo una considerable disminución en la cantidad de elementos conservando la precisión del cálculo.

### **Proyecciones de este trabajo**

El trabajo presentado en esta tesis es sólo el punto de partida para realizar un interesante trabajo de investigación en el área de los métodos numéricos en la mecánica de la fractura, o en general al uso de los algoritmos Lepp-Delaunay en problemas de ingeniería. Proyecciones futuras al trabajo presentado se muestran a continuación

- Complementar la metodología para predecir el camino a seguir de la grieta con modelos de fatiga más complejos, que incluyan cargas variables y aleatorias.
- Extensión del método a 3D, usando mallas de tetraedros. Actualmente ya existen implementaciones de algoritmos de refinamiento/mejoramiento basados en el LEPP para tetraedros.
- La metodología basada en los algoritmos Lepp-Delaunay es independiente del problema o el modelo a resolver. Esto permitiría extender la aplicación para otros modelos de fractura, por ejemplo a problemas elasto-plásticos o fractura dinámica, así como a materiales anisotrópicos o sistemas con más de un material.
- Dada la capacidad de los algoritmos Lepp-Delaunay de refinar localizadamente, tienen un gran potencial para su uso en métodos adaptivos en mecánica de la fractura.

- Así mismo puede extenderse el uso de los algoritmos Lepp-Delaunay a otros problemas de la ingeniería como fluidos, campos electromagnéticos, etc., incluyendo distintos métodos numéricos sin limitarse al método de elementos finitos.

## Bibliografía

- Anderson, T.L. 1995. *Fracture Mechanics: fundamentals and applications*. CRC Press.
- Babuska, I., & Aziz, A.K. 1976. On the Angle Condition in the Finite Element Method. *SIAM Journal on Numerical Analysis*, **13**, 214–226.
- Babuska, I., & Melenk, J.M. 1997. The partition of the unity Method. *International Journal for Numerical Methods in Engineering*, **40**, 727–758.
- Banks-Sills, L., & Sherman, D. 1986. Comparison of method for calculating stress intensity factors with quarter-point elements. *International Journal of Fracture*, **32**, 127–140.
- Barsoum, R.S. 1976. On the use of isoparametric finite elements in linear fracture mechanics. *International Journal for Numerical Methods in Engineering*, **10**, 25–37.
- Barsoum, R.S. 1977. Triangular quarter-point elements as elastic and perfectly-plastic crack tip elements. *International Journal for Numerical Methods in Engineering*, **11**, 85–98.
- Bittencourt, T.N., Wawrzynek, P.A., Ingraffea, A.R., & Sousa, J.L. 1996. Quasi-automatic simulation of crack propagation for 2D LEFM problems. *Engineering Fracture Mechanics*, **55**, 321–334.
- Calderón, C. 2004. *Memoria de Ingeniería Civil en Computación. Herramienta computacional de apoyo a la investigación en algoritmos de triangulaciones*. Departamento de Ciencias de la Computación - Universidad de Chile.
- Calderón, C., & Rivara, M.C. 2004. 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*.
- Carter, B.J., Wawrzynek, P.A., & Ingraffea, A.R. 2000. Automated 3D crack growth simulation. *International Journal for Numerical Methods in Engineering*, **47**, 229–253.
- Castro, J.T.P, Miranda, A.C.O, Meggiolaro, M.A., Martha, L.F., & Bittencourt, T.N. 2003. Fatigue Life and Crack Path Predictions in Generic 2D Structural Components. *Engineering Fracture Mechanics*, **70**, 1259–1279.

- Cavalcante-Neto, J.B., Wawrzynek, P.A., Carvalho, M.T.M., Martha, L.F., & Ingraffea, A.R. 2001. An Algorithm for Three-Dimensional Mesh Generation for Arbitrary Regions with Cracks. *Engineering with computers*, **17**, 75–91.
- Chan, S.K., Tuba, I.S., & Wilson, W.K. 1970. On the finite element method in linear fracture mechanics. *Engineering Fracture Mechanics*, **2**, 1–17.
- de Lorenzi, H.G. 1982. On energy release rate and the J-integral for 3-D crack configuration. *International Journal of Fracture*, **19**, 183–192.
- de Lorenzi, H.G. 1985. Energy release rate calculations by finite element method. *Engineering Fracture Mechanics*, **21**, 129–143.
- Erdogan, F., & Sih, G.C. 1963. On the Crack Extension in Plates under Plane Loading and Transverse Shear. *Journal of Basic Engineering*, **85**, 519–527.
- Eshelby, J.D. 1970. Energy relations and the energy momentum tensor in continuum mechanics. *Inelastic Behaviour of Solids*, 77–114.
- Fish, J. 1992. The s-version of the finite element method. *Computers and Structures*, **43**, 539–547.
- Frey, J.P., & George, P. 2000. *Mesh Generation - application to finite elements*. Hermes Science.
- Gosz, M., & Moran, B. 2002. 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.
- Griffith, A.A. 1920. The phenomena of rupture flow in solids. *Philosophical transactions, Series A*, **221**, 163–198.
- Guinea, G. V., Planas, J., & Elices, M. 2000. KI evaluation by the displacement extrapolation technique. *Engineering Fracture Mechanics*, **66**, 243–255.
- Hayashi, K., & Nemat-Nasser, S. 1981. Energy-Release Rate and Crack Kinking under Combined Loading. *Journal of Applied Mechanics*, **48**, 520–524.
- Hellen, T.K. 1975. On the method of virtual crack extension. *International Journal for Numerical Methods in Engineering*, **9**, 187–207.
- Henshell, R.D., & Shaw, K.G. 1975. Crack tip finite elements are unnecessary. *International Journal for Numerical Methods in Engineering*, **9**, 495–507.
- Irwin, G.R. 1948. Fracture dynamics. *Fracturing of metals, American Society of metals, Cleveland*, 174–166.
- Irwin, G.R. 1956. Onset of fast crack propagation in high strength steel and aluminium alloys. *Sagamore Research Conference Proceedings*, **2**, 289–305.

- Irwin, G.R. 1957. Analysis of stress and strains near the end of a crack traversing a plate. *Journal of Applied Mechanics*, **24**, 361–364.
- Irwin, G.R. 1960. Plastic zone near a crack and fracture toughness. *Proceedings of the seventh sagamore ordnance materials conference, Syracuse University*, **4**, 63–78.
- Kanninen, M.F., & Popelar, C.H. 1985. *Advanced Fracture Mechanics*. Oxford University Press.
- Li, F.Z., Shih, C.F., & Needleman, A. 1985. A comparison of methods for calculating energy release rates. *Engineering Fracture Mechanics*, **21**, 405–421.
- Lin, X.B., & Smith, R.A. 1999. Finite element modelling of fatigue crack growth of surface cracked plates Part I: The numerical technique. *Engineering Fracture Mechanics*, **63**, 503–522.
- Liu, G.R., & Quek, S.S. 2003. *The finite element method, a practical course*. Butterworth Heinemann.
- Miranda, A.C.O., Meggiolaro, M.A., Castro, J.T.P, Martha, L.F., & Bittencourt, T.N. 2003. Fatigue life and crack path predictions in generic 2D structural components. *Engineering Fracture Mechanics*, **70**, 1259–1279.
- Moës, N., Dolbow, J., & Belytschko, T. 1999. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 131–150.
- Nikishikov, G.P., & Atluri, S.N. 1987. Calculation of fracture mechanics parameters for an arbitrary three-dimensional crack, by the 'equivalent domain integral' method. *International Journal for Numerical Methods in Engineering*, **24**, 1801–1821.
- Okada, H., Endoh, S., & Kikuichi, M. 2005a. On fracture Analysis using Element Overlay Technique. *Engineering Fracture Mechanics*, **72**, 773–789.
- Okada, H., Higashi, M., Kikuchi, M., Fukui, Y., & Kumazawa, N. 2005b. 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.
- Paris, P.C., Gomez, M.P., & Anderson, W.P. 1961. A rational analytic theory of fatigue. *The trend in engineering*, **13**, 9–14.
- Parks, D.M. 1974. A stiffness derivative finite element technique for determination of crack tip stress intensity factors. *International Journal of Fracture*, **10**, 487–502.
- Phongthanapanich, S., & Dechaumphai, P. 2004. Adaptive Delaunay triangulation with object-oriented programming for crack propagation analysis. *Finite Elements in Analysis and Design*, **40**, 1753–1771.
- Potyondy, D.O., Wawrzynek, P.A., & Ingraffea, A.R. 1995. An Algorithm to Generate Quadrilateral or Triangular Element Surface Meshes in Arbitrary Domains with Applications to Crack Propagation. *International Journal for Numerical Methods in Engineering*, **38**, 2677–2701.



- Rajaram, H., Socrate, S., & Parks, D.M. 2000. Application of domain integral methods using tetrahedral elements to the determination of stress intensity factors. *Engineering Fracture Mechanics*, **66**, 155–482.
- Raju, I.S. 1987. Calculation of strain-energy release rates with higher order and singular finite elements. *Engineering Fracture Mechanics*, **28**, 251–274.
- Raju, I.S., & Shivakumar, K.N. 1990. An equivalent domain integral method in the two-dimensional analysis of mixed mode crack problems. *Engineering Fracture Mechanics*, **37**, 707–725.
- Rice, J.R. 1968. A path independent integral and the approximate analysis of strain concentrations by notches and cracks. *Journal of Applied Mechanics*, **35**, 376–86.
- Rivara, M.C. 1984. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering*, **20**, 745–756.
- Rivara, M.C. 1997. New Longest-Edge algorithms for the refinement and/or improvement of unstructured triangulations. *International Journal for Numerical Methods in Engineering*, **40**, 3313–3324.
- Rivara, M.C., & Hitschfeld, N. 1999. LEPP-Delaunay algorithm: a robust tool for producing size-optimal quality triangulations. *Proceedings of the 8th Int. Meshing Roundtable*, 205–220.
- Rivara, M.C., & Levín, C. 1992. A 3-D refinement algorithm suitable for adaptive and multi-grid techniques. *Communications in applied numerical methods*, **8**, 281–290.
- Rivara, M.C., & Palma, M. 1997. 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.
- Rivara, M.C., Hitschfeld, N., & Simpson, B. 2001. Terminal-edges Delaunay (small-angle based) algorithm for the quality triangulation problem. *Computer Aided Design*, **33**, 263–277.
- Rybicki, E.F. & Kanninen, M.F. 1977. Finite element calculation of stress intensity factors by a modified crack closure integral. *Engineering Fracture Mechanics*, **9**, 931–938.
- Schöllmann, M., Fulland, M., & Richard, H.A. 2003. Development of a new software for adaptive crack growth simulations in 3D structures. *Engineering Fracture Mechanics*, **70**, 249–268.
- Shewchuk, J.R. 1998. Tetrahedral mesh generation by delaunay refinement. *Proceedings of the annual symposium on computational geometry*, 86–95.
- Shih, C.F., & Asaro, R.J. 1988. Elastic-plastic analysis of cracks on bimaterial interfaces: Part I-Small scale yielding. *Journal of Applied Mechanics*, **55**, 299–316.
- Shih, C.F., & Hutchinson, J.W. 1976. Fully plastic solutions and large-scale yielding estimates for plane stress crack problems. *Journal of engineering materials and technology*, **98**, 289–295.

- Shivakumar, K.N. 1992. An equivalent domain integral method for three-dimensional mixed mode fracture problems. *Engineering Fracture Mechanics*, **42**, 935–959.
- Shivakumar, K.N., Tan, P.W., & Newman Jr., J.C. 1988. Virtual crack-closure technique for calculating stress intensity factors for cracked three dimensional bodies. *International Journal of Fracture*, **36**, r43–r50.
- Stern, M., Becker, E.B., & Dunham, R.S. 1976. A contour integral computation of mixed-mode stress intensity factors. *International Journal of Fracture*, **12**, 1411–1426.
- Tada, H., Paris, P.C., & Irwin, G.R. 1985. *The stress analysis of cracks handbook*. Paris Production, St. Louis, MO.
- web. <http://ksm.fsv.cvut.cz/dr/papers/Thesis/node23.html>.
- Wells, A.A. 1961. Unstable crack propagation in metals: cleavage and fast fracture. *Proceedings of the crack propagation symposium*, **1**.
- Westergaard, H.M. 1939. Bearing pressures and cracks. *Journal of Applied Mechanics*, **6**, 49–53.
- Williams, M.L. 1957. On the stress distribution at the base of a stationary crack. *Journal of Applied Mechanics*, **24**, 109–114.
- Yerry, M.A., & Shephard, M.S. 1983. Finite element mesh generation based on a modified quadtree approach. *IEEE Computer Graphics and applications*, **3**, 36–46.
- Zienkiewicz, O.C., & Taylor, R.L. 2000. *The finite element method*. Vol. I. McGraw Hill.
- Zuo, J., Deng, X., & Sutton, M.A. 2005. Advances in tetrahedral mesh generation for modelling of three-dimensional regions with complex, curvilinear crack shapes. *International Journal for Numerical Methods in Engineering*, **63**, 256–275.