

UNIVERSITY OF ILLINOIS  
AT URBANA-CHAMPAIGN

# RCIP Method for a 2D Edge Crack Problem

Alfredo Sanchez  
December 8<sup>th</sup> 2017



[illinois.edu](http://illinois.edu)

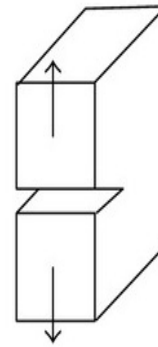


## LEFM – Brief Review

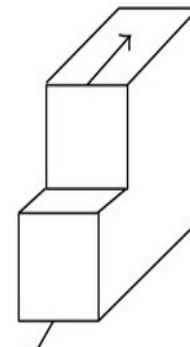
Linear Elastic Fracture Mechanics (LEFM) deals with the study of cracks in materials, especially concerning crack propagation

It is of particular importance to obtain accurate solutions to quantities such as the stress field.

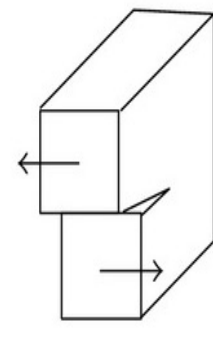
Specially stresses around the crack tip, since they provide useful information to determine the direction of the crack propagation (SIFs).



Mode 1  
(opening)



Mode 2  
(sliding)



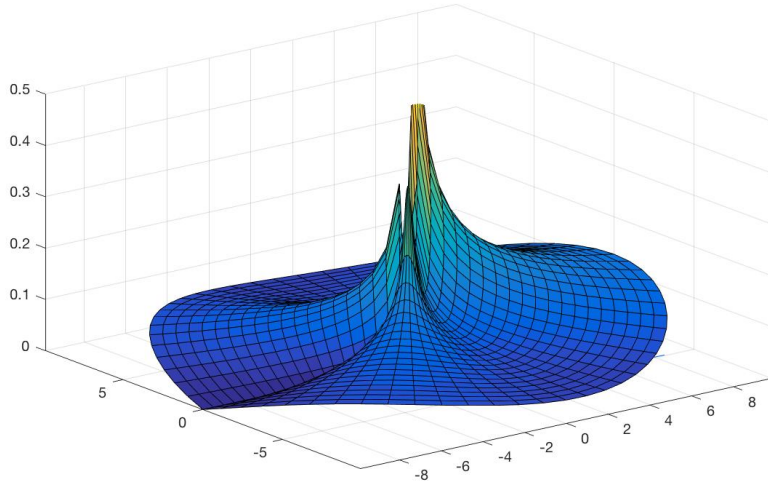
Mode 3  
(tearing)

HOWEVER...

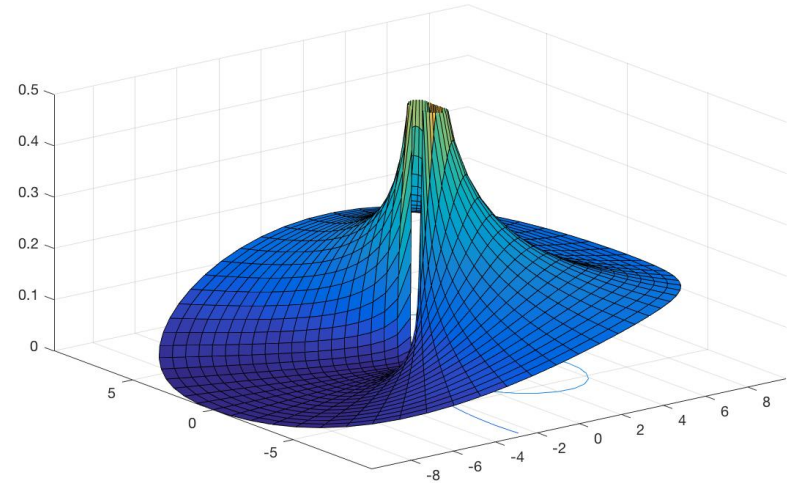


# LEFM – Brief Review

Mode I stress field for  $\sigma_{xx}$



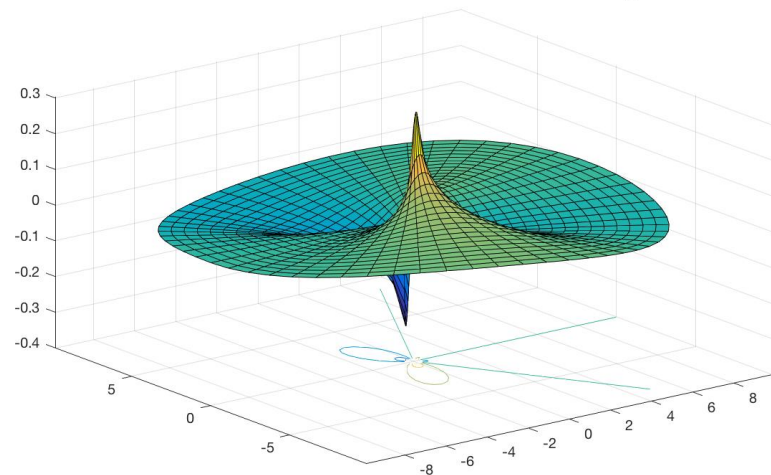
Mode I stress field for  $\sigma_{yy}$



Singular behavior at the crack tip!!!

Hard to obtain accurate solutions numerically in the neighborhood...

Mode I stress field for  $\sigma_{xy}$





# Numerical Treatment of Crack Problems

---

## Finite Element Method:

### Pros:

- Widely available/used and/or easy to implement.
- Can handle complex domain geometries easily.
- Formulation easily adapted to add features to problems and to higher dimensions.

### Cons:

- Mesh has to adapt to the crack geometry, inconvenient when dealing with crack propagation.
- Standard FEM polynomial basis cannot capture properly the behavior close to the singularity (workarounds like quarter point elements).
- Even at its best, sub-optimal solution.



# Numerical Treatment of Crack Problems

---

## XFEM/GFEM:

### Pros:

- Retain most advantages of classical FEM.
- Easy to implement on top of an existing FEM formulation.
- Mesh doesn't have to adapt to the crack geometry.
- Singular enrichments around crack tip can lead to optimal solution.

### Cons:

- Enrichment functions lead to very ill-conditioned matrices (workarounds, such as SGFEM).



# Numerical Treatment of Crack Problems

---

## Singular Integral Equations (Method Discussed):

### Pros:

- Information for all the domain obtained only from Boundary (way less degrees of freedom).
- Kernels designed to contain enough information provide optimal solutions.

### Cons:

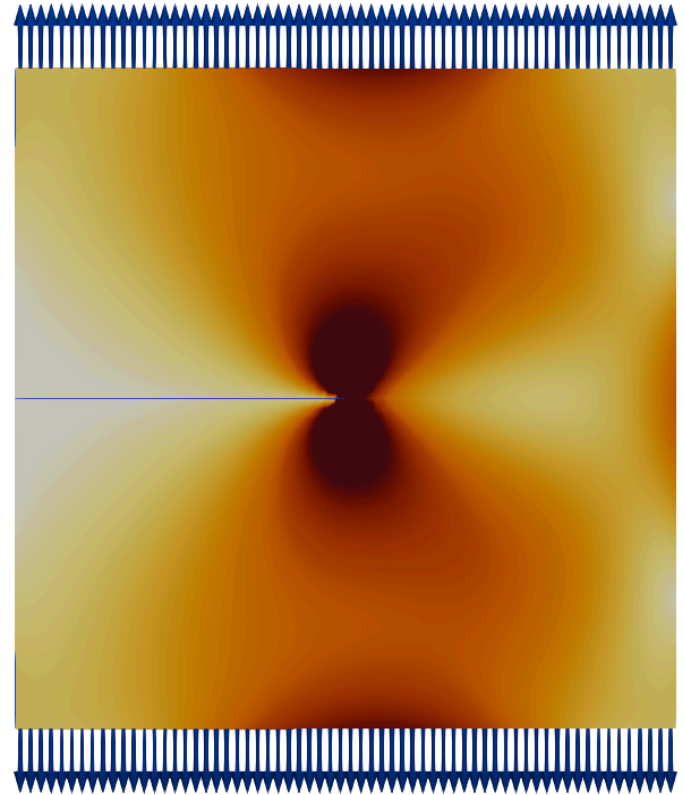
- Kernels have a scary appearance.
- Stress singularity requires special quadrature rules for some regions.
- Cannot handle non-smooth boundary geometries by itself (workarounds like RCIP method)



# Our Problem

## Problem Description:

- Square 2D domain  $16u \times 16u$ .
- Crack length  $8u$ .
- Crack Tip at the center of the domain.
- Crack Surface is traction free.
- Neumann BCs (Tractions) of  $1u$  applied outwards on top and bottom.
- Plane Strain Linear Elasticity.





# Our Formulation

Original Formulation:

$$(M_1 - M_3)\omega(z) = g(z), \quad z \in \Gamma$$

Making it more "numerically friendly":

$$(I - (M_3 + h(iP_0 + \bar{z}Q))\rho M_1\rho^{-1})\Omega(z) = g(z), \quad z \in \Gamma$$

$$Q\omega(z) = \frac{1}{\pi i} \int_{\Gamma} \omega(z) dz, \quad P_0\omega(z) = \frac{1}{A} \Re \left\{ \int_{\Gamma} \bar{z}\omega(z) dz \right\}$$

$$M_1\omega(z) = \frac{1}{\pi i} \int_{\Gamma} \frac{\omega(\tau) d\tau}{\tau - z}, \quad \rho M_1\rho^{-1}\omega(z) = \frac{\rho(z)}{\pi i} \int_{\Gamma} \frac{\omega(\tau) d\tau}{\rho(\tau)(\tau - z)},$$

$$M_3\omega(z) = \frac{1}{2\pi i} \left[ \int_{\Gamma} \frac{\omega(\tau) d\tau}{\tau - z} + \frac{\bar{n}(z)}{n(z)} \int_{\Gamma} \frac{\omega(\tau) d\tau}{\bar{\tau} - \bar{z}} + \int_{\Gamma} \frac{\bar{\omega}(\tau) d\bar{\tau}}{\bar{\tau} - \bar{z}} + \int_{\Gamma} \frac{(\tau - z)\bar{\omega}(\tau) d\tau}{(\bar{\tau} - \bar{z})^2} \right],$$

$$g(z) = \frac{\bar{n}(z)t(z)}{2} + \frac{\bar{n}(z)}{n(z)} \frac{1}{2\pi i} \int_{\Gamma} \frac{n(\tau)t(\tau) d\bar{\tau}}{\bar{\tau} - \bar{z}}, \quad z \in \Gamma$$





# Our Formulation

Let  $\Gamma_c$  be the crack surface,  $\Gamma_0$  the domain boundary,  $D$  the domain interior and  $D'$  the exterior:

$$\rho(z) = \begin{cases} -(z - \gamma^i)^{1/2}(z - \gamma^t)^{-1/2}, & z \in D \cup \Gamma_c \\ (z - \gamma^i)^{1/2}(z - \gamma^t)^{-1/2}, & z \in D' \cup \Gamma_0 \end{cases}$$

where  $\gamma^i$  and  $\gamma^t$  are the crack intersection with the edge and the crack tip respectively.

$$h(z) = \begin{cases} 1, & z \in \Gamma_0 \\ 0, & z \in \Gamma_c \end{cases}$$

How this whole mess relates to the quantities of interest????



# Our Formulation

Finally, once the system is solved for  $\Omega(z)$ :

$$\omega(z) = \rho M_1 \rho^{-1} \Omega(z)$$

$$\Phi(z) = \frac{1}{2\pi i} \int_{\Gamma} \frac{\omega(\tau) d\tau}{\tau - z}, \quad \Psi(z) = -\frac{1}{2\pi i} \left[ \int_{\Gamma} \frac{\bar{\omega}(\tau) d\tau}{\tau - z} + \int_{\Gamma} \frac{\bar{z}\omega(\tau) d\tau}{(\tau - z)^2} + \int_{\Gamma} \frac{\bar{n}(\tau)\bar{t}(\tau) d\tau}{\tau - z} \right]$$

$$\begin{cases} \sigma_{xx}(z) + \sigma_{yy}(z) & = 4\Re\{\Phi(z)\} \\ \sigma_{yy}(z) - \sigma_{xx}(z) + 2i\sigma_{xy}(z) & = 2(\bar{z}\Phi'(z) + \Psi(z)) \end{cases}$$

- The formulations shown make the singular equation “numerically solvable” (with additional quadrature considerations).
- We have to still deal with boundary non-smoothness!!

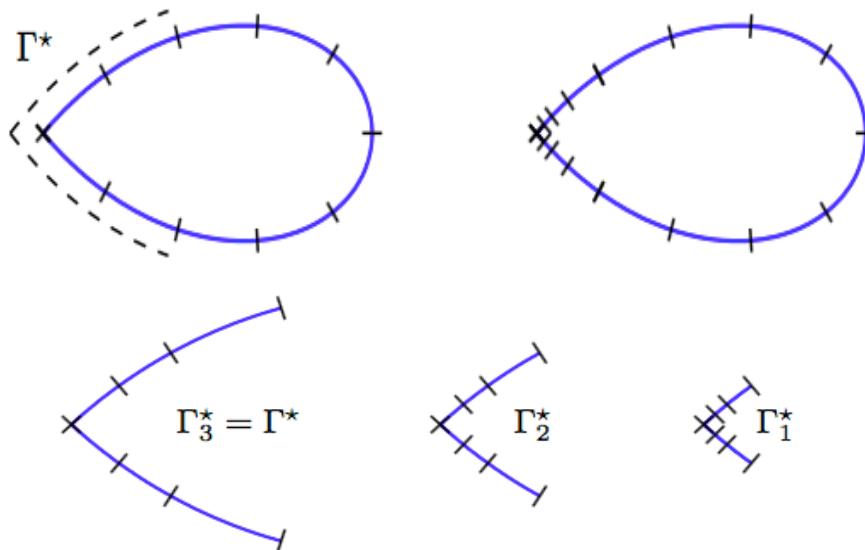


# RCIP Method

RCIP = *Recursive Compressed Inverse Preconditioning*.

Developed by Prof. Jonas Helsing, Lund University (2008)

- Conceptually, uses integral transforms whose inverses modify the kernels of the IE so that layer density becomes piecewise smooth. (Inverse part)
- Such inverses are constructed recursively on local temporary meshes. (Compression and Recursion)



- Pick a sub region of two panels on each side of the corner.
- Split the closest panel on each side by half.
- Do it in a recursive fashion.



# RCIP Method

## Discretization on two meshes:

Consider the following set of two discretizations of an IE:

$$(I_c + K_c)\sigma_c = g_c$$

$$(I_f + K_f)\sigma_f = g_f$$

Where the subscript  $c$  represents the “coarse” mesh, and the subscript  $f$  represents the “fine” mesh (where we applied recursively the refinement explained previously several times).

Our goal is to compress the information from the second equation into the first one, therefore, solving a much smaller problem.



# RCIP Method

## Operator splitting:

To do that, consider the following splitting of the operators:

$$K_c = K_c^* + K_c^o$$

$$K_f = K_f^* + K_f^o$$

Where the superscript \* represents the interaction between points belonging to the four subpanels adjacent to the corner.

We also define the “Prolongation Operator”, which is just a interpolation matrix from the quadrature points of the coarse grid to the fine grid.

$$P_W = W_f P W_c^{-1}$$

$$P_W^T P = I_c$$



# RCIP Method

## Compressed System:

$$(I_f + K_f^*)\sigma_f = P\tilde{\sigma}_c$$

$$K_f^o = PK_c^o P_W^T$$

Notice that the last equation is just a low rank decomposition of the fine operator. We can now define the “compressed weighted inverse” which is the heart of the RCIP method:

$$R = P_W^T (I_f + K_f^*)^{-1} P$$

And proceed to solve the compressed system instead:

$$(I_c + K_c^o R)\tilde{\sigma} = g_c$$

Notice however, that the computation of R is not efficient, since it involves inverting a matrix which size is of the fine problem.

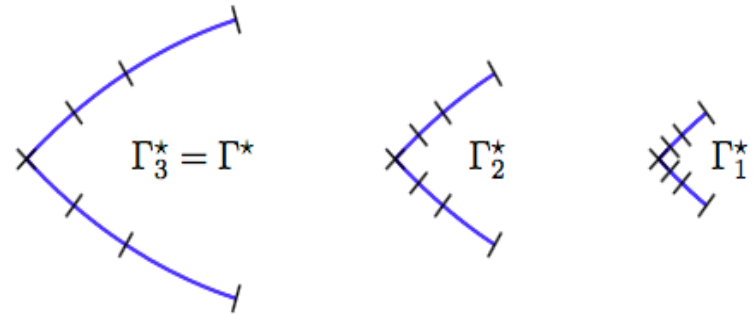


# RCIP Method

## Recursive Compression:

We opt instead of taking advantage of the recursive nature of the grid refinement scheme.

Denote by the subscript "b" a subgrid consisting on six panels adjacent to a corner, and the subscript "i" the level of nestedness of such a subpanel (i.e. how many times we've applied the recursion).



Then we can obtain the compressed inverse for any level of refinement by:

$$R_i = P_{Wbc}^T (\mathbb{F}\{R_{i-1}^{-1}\} + I_b^o + K_{ib}^o)^{-1} P_{bc}$$

$$\mathbb{F}\{R_0^{-1}\} = I_b^* + K_{1b}^*$$



# RCIP Method

## Taking advantage of scale invariance:

A couple of techniques to accelerate this recursion are discussed in Prof. Helsing's RCIP tutorial. However, it is worth noting that in the neighborhood of the corner is scale invariant (corner of a polygon for example), we can realize that the K operator is independent of the subscript  $i$ , therefore we can consider the previous recursion simply as:

$$R_i = P_{Wbc}^T (\mathbb{F}\{R_{i-1}^{-1}\} + I_b^o + K_b^o)^{-1} P_{bc}$$

$$\mathbb{F}\{R_0^{-1}\} = I_b^* + K_{1b}^*$$

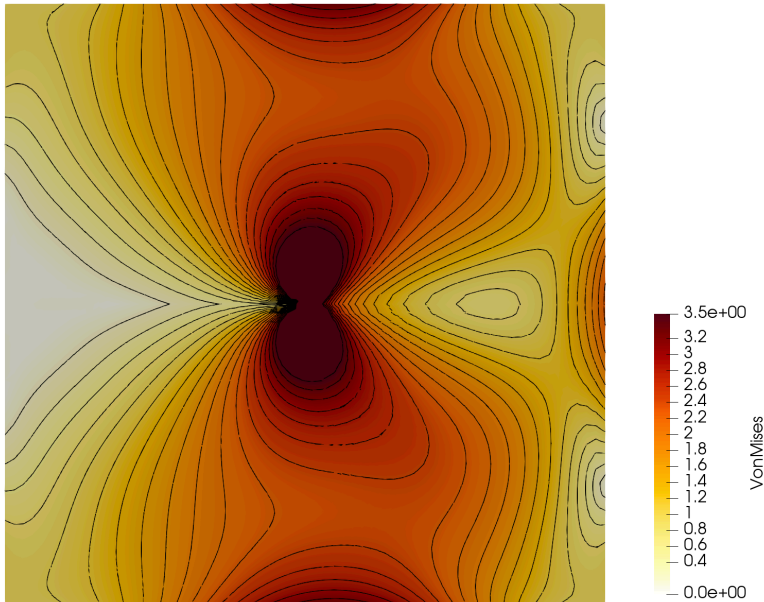
And solve this as a fixed point iteration problem until reaching a converged value for  $R$ .

By doing so, we don't pick in advance how many subdivisions it will take.





# Comparative Results



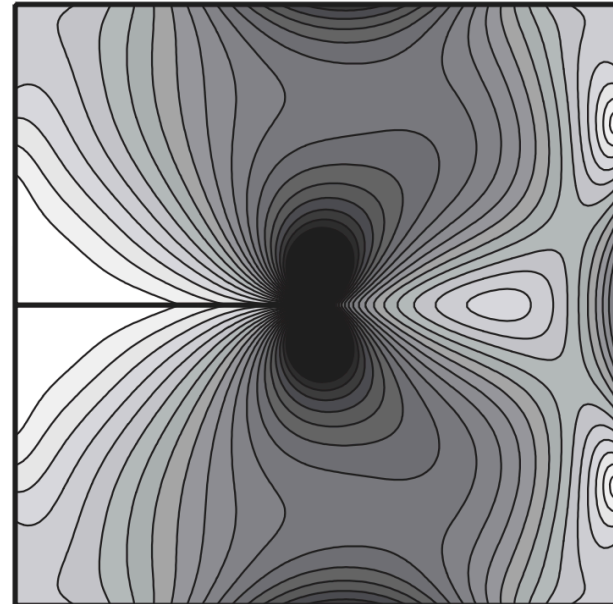
GFEM von Mises Stress Contour Solution

-

A. Sanchez, sometime around this week

-

64x64 mesh – 36598 dofs



Singular IE von Mises Stress Contour Solution

-

J. Englund, Lund University, 2005

-

1660 discretization points



---

***THANKS!***

Special thanks to Prof. Helsing for the vast amount of helping material available online on this topic, and for happily being willing to help when I contacted him with my doubts.

***QUESTIONS?***