CS598 APK: Fast Algorithms and Integral Equations
Final Project Report
RCIP Method for a 2D Edge Crack Problem

Alfredo Sanchez Rivadeneira
NetID: snchzrv2

1 Introduction

Linear Elastic Fracture Mechanics (LEFM) is a field that deals with the study of the behavior of cracked materials. It is usually particularly concerned with crack propagation problems, which consist in predicting how and in which direction a crack will propagate in a body under applied loads.

A useful set of quantities for determining how a crack will propagate are the Stress Intensity Factors (SIFs), which are related to the asymptotic behavior of the stress field close to the crack tip. Therefore, if we are interested in problems involving cracks, we are usually interested in obtaining accurate representations of the stress field in our numerical schemes.

However, the stress field around the crack tip in an elastic material exhibits singular behavior, therefore, naive numerical scheme implementations might fail to provide a proper approximation for stresses close to this region.

Common numerical schemes used to solve these kind of problems are the Finite Element Method and its variants such as the eXtended/Generalized Finite Element Method (XFEM/GFEM) and Integral Equation Methods. A lot of effort and ongoing research is put in improving each of these methods, and certainly they all have their pros and cons.

In this project, we are going to analyze and compare the performance of an Integral Equation Method and the Generalized Finite Element Method in solving a 2D Edge Crack Problem.

Integral Equation Methods are popular due to its ability to obtain relatively accurate solutions cheaply, since only the boundary has to be discretized. The resulting system to solve at the end, is a small (but dense) system. On the other hand XFEM/GFEM is also a very popular method that extends the capabilities of regular FEM to obtain accurate solution for problems where FEM performance is suboptimal. As FEM, a mesh of the whole domain is required, therefore, the resulting system is expected to be larger than IEM (but sparse).

2 Problem Statement

The problem to solve is a very common problem in the literature. It consists in a rectangular (square) domain of 16×16 length units, with a straight edge crack spanning from the midpoint of the left side of our domain to the center of it. Regarding to the Boundary Conditions, we consider Neumann BCs only which correspond to applied tractions for this type of problem of 1 traction unit applied outwards on top and bottom sides of the domain. Also, the crack surface (line) is considered traction free.

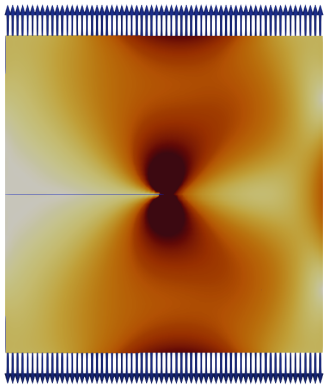


Figure 1: Schematic representation of the problem geometry and boundary conditions

3 Integral Equation Method Formulation

Note: The formulation procedure derived here is more or less the same throughout all the references by J. Helsing and J. Englund cited in this report. So, unless some very specific commentary or definition, references won't be cited individually in this section.

Let's denote D and D' as the domain interior and exterior respectively and Γ_0 and Γ_c the domain boundary and crack surface, such that all our source points lie on $\Gamma = \Gamma_0 \cup \Gamma_c$.

In order to obtain accurate results for our problem we proceed to set up a stress formulation for the elasticity problem based on the Airy stress function $W(x, y)$ that can be represented as $W(x, y) = \Re\{\bar{z}\phi(z) + \chi(z)\}$, where $\phi(z)$ and $\chi(z)$ are analytic functions of the complex variable $z = x + yi$.

Let $\Phi(z) = \phi'(z)$ and $\Psi(z) = \chi''(z)$, via the Kolosov relations we can obtain the stress field everywhere on $D \cup D'$ by

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

where σ_{ij} are the components of the Cauchy stress tensor field at point z.

Our formulation considers both the exterior and interior of the domain, which according to [4] leads to a better integral equation from the numerical point of view, effectively eliminating unbounded stresses around the domain's corners as seen in the following figure from that article:

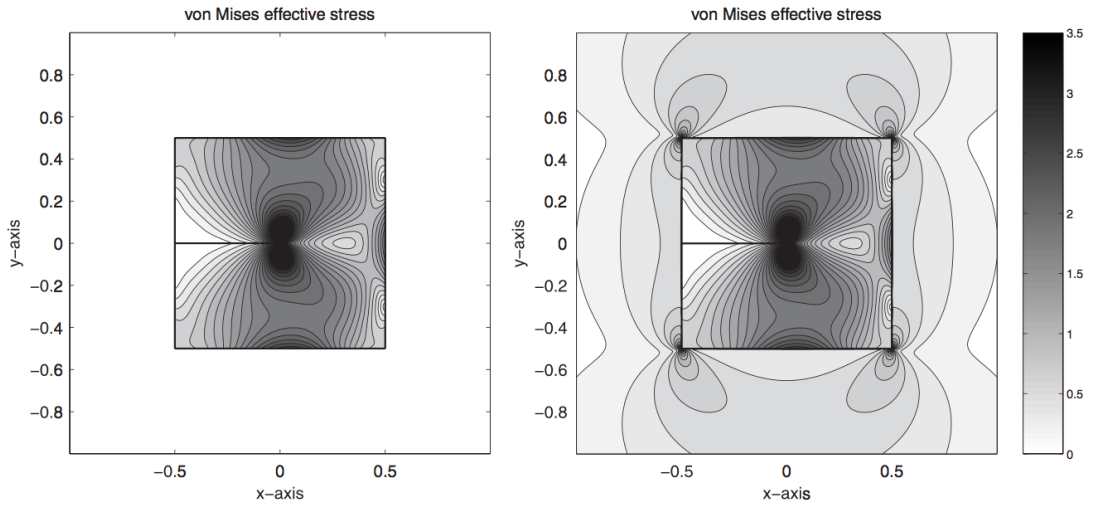


Figure 2: Effect of considering D and D' in problem formulation (left) vs. considering just D (right). Figure borrowed from [4]

Within this formulation, let $t^+(z) = t_x(z) + t_y(z)$ be the applied tractions to D along Γ and $t^-(z) = 0$ along Γ and $\lim_{z\to\infty} \sigma_{ij}(z) = 0$ for D' (*i.e.* D' is "unloaded").

There are a couple of conditions that the boundary applied tractions should satisfy to ensure the existence of a solution: the body should be in equilibrium, that is, the resultant net force and moment produced by the tractions should be zero. Putting this in notation, let $n = n(z)$ be the outward unit normal vector (wrt. D) at $z \in \Gamma$. Define the operators

$$Qf = \frac{1}{\pi i}\int_\Gamma f(\tau)d\tau, \qquad P_0 f = \frac{1}{2A}\Re\left\{\int_{\Gamma_0} f(\tau)\bar{\tau}d\tau\right\}$$

where A is the area of D.

Then, equilibrium of forces and moments is equivalent to

$$Q\bar{n}t^+ = 0, \qquad P_0\bar{n}t^+ = 0$$

respectively.

Define now the potentials $\Phi(z)$ and $\Psi(z)$ as

$$\Phi(z) = \frac{1}{2\pi i}\int_\Gamma \frac{\omega(\tau)d\tau}{\tau - z}, \quad z \in D \cup D'$$

$$\Psi(z) = -\frac{1}{2\pi i}\left[\int_\Gamma \frac{\bar{\omega}(\tau)d\tau}{\tau - z} + \int_\Gamma \frac{\bar{z}\omega(\tau)d\tau}{(\tau - z)^2} + \int_\Gamma \frac{\bar{n}(\tau)(\bar{t}^+(\tau) - \bar{t}^-(\tau))d\tau}{\tau - z}\right], \quad z \in D \cup D'$$

where $\omega(z)$ is the unknown density for which we will solve the following integral equation

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

where

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

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

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

Solving the integral equation as it stands, although possible, is not recommended from a numerical point of view. In order to build an equation better suited for numerics we define the *fundamental function*:

$$\rho(z) = \rho_0(z)(z - \gamma^{bp})^{1/2} \cdot (z - \gamma^{tp})^{-1/2}$$

$$\rho_0(z) = \begin{cases} -1, & z \in D \cup \Gamma_c \\ 1, & z \in D' \cup \Gamma_0 \end{cases}$$

where γ^{bp} and γ^{tp} are the crack intersection with the boundary of D and the crack tip respectively.

In the references it is shown that

$$M_1\rho M_1\rho^{-1} = I$$

which means that $\rho M_1\rho^{-1}$ is a right inverse of M_1.

Taking advantage of that, we now consider $\Omega(z)$ such that $\omega(z) = \rho M_1\rho^{-1}\Omega(z)$. The equation to solve now becomes

$$\Omega(z) - M_3\rho M_1\rho^{-1}\Omega(z) = g(z), \qquad z \in \Gamma$$

which is better suited for numerics [9].

Furthermore, to achieve even a better performance out of the integral equation, we proceed to make the some modifications as in [4] to arrive to the final form of the equation to solve:

$$(I - M_3^*\rho M_1\rho^{-1})\Omega(z) = g(z), \qquad z \in \Gamma$$

where

$$M_3^* = M_3 + h(iP_0 + \bar{z}Q)$$

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

4 RCIP Method

Note: This section is roughly speaking a summary of the useful parts for our purposes of the RCIP tutorial written by J. Helsing [10].

The *Recursive Compressed Inverse Preconditioning* method (RCIP) is a method developed by Prof. J. Helsing, which conceptually uses integral transforms whose inverses modify the kernels of the Integral Equation, so that the layer density becomes piecewise smooth (Inverse part of the name), furthermore, those inverses are constructed recursively on local temporary meshes (compression and recursion part of the name).

Discretization on two meshes

To start explaining the generalities of the method, we should consider two set of discretizations of a problem of the form

$$(I_c + K_c)\sigma_c = g_c$$
$$(I_f + K_f)\sigma_f = g_f$$

where the subindex c corresponds to a coarse mesh discretization, while the subindex f correspond to a fine discretization which is obtained by refining the coarse mesh by bisecting the panels closest to a corner recursively n times.

The idea is to compress the information from the fine discretization into the coarse problem *i.e.* solving a coarse size problem with the same accuracy as the fine problem.

Operator Splitting

Denote Γ^* as the portion of a boundary Γ which lies within two panels away from a corner. We consider the following operator splitting

$$K_c = K_c^* + K_c^\circ$$
$$K_f = K_f^* + K_f^\circ$$

where the superscript $*$ represents the interaction between points belonging to Γ^*

In order to interpolate function values between the coarse and fine meshes, we define the "Prolongation Operator" P and "Weighted Prolongation Operator" $P_W = W_f P W_c^{-1}$ which are based on Vandermonde matrices for polynomial interpolation between the quadrature points of the coarse grid to the fine grid.

Compressed System

After some algebraic manipulations we arrive to a compressed version of the fine discretization that reads as

$$(I_c + K_c^\circ R)\tilde{\rho}_c = g_c$$

where

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

and $\tilde{\rho}_c$ is a modified density, from which ρ_f can be recovered in postprocessing theoretically without loss of accuracy.

Recursive Compression

Notice however, that the expression for R presented above is not efficient to calculate, since it involves calculating an inverse of an operator of the size of the fine problem. However, we can take advantage of the recursive fashion from which the fine grid is generated.

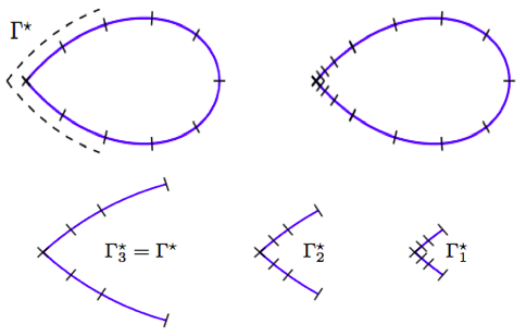


Figure 3: Schematic of the recursive compression scheme [10]

Consider a subgrid consisting of the six adjacent panels to a corner on a refined grid (as shown in the lower part of the picture above), and P_{bc} the prolongation operator from a four (equally sized) panel discretization of the same subgrid (such as the inner four panels of Γ_3^* wrt. Γ_2^* or the inner four panels of Γ_2^* wrt. Γ_1^*). In general, we differentiate all the operators related to a six adjacent panel configuration with the subscript b. In other words, the six panel subgrid correspond to one step of refinement from a four panel grid. We can define now R recursively from Γ_{i-1}^* to Γ_i^* as

$$R_i = P_{Wbc}^T(\mathbb{F}\{R_{i-1}^{-1}\} + I_b^\circ + K_{ib}^\circ)^{-1}P_{bc} \qquad i = 1, 2, ..., n$$
$$\mathbb{F}\{R_0^{-1}\} = I_b^* + K_{1b}^*$$

Composed Operators

Recall that our integral equation is of the form

$$(I + MK)\sigma_1 = g$$

The RCIP method can be adapted fairly straightforward to the case of composed operators by considering the following system

$$\sigma_1 + M\sigma_2 = g$$
$$-K\sigma_1 + \sigma_2 = 0$$

In matrix form (for a fine grid)

$$\left(\begin{bmatrix} I_f & 0 \\ 0 & I_f \end{bmatrix} + \begin{bmatrix} 0 & M_f \\ -K_f & 0 \end{bmatrix}\right)\begin{bmatrix} \sigma_{1f} \\ \sigma_{2f} \end{bmatrix} = \begin{bmatrix} g_f \\ 0 \end{bmatrix}$$

Considering a splitting of the matrix R corresponding to this augmented system gives (from a standard RCIP)

$$\left(\begin{bmatrix} I_c & 0 \\ 0 & I_c \end{bmatrix} + \begin{bmatrix} 0 & M_c \\ -K_c & 0 \end{bmatrix}\begin{bmatrix} R_1 & R_3 \\ R_2 & R_4 \end{bmatrix}\right)\begin{bmatrix} \tilde{\sigma}_{1c} \\ \tilde{\sigma}_{2c} \end{bmatrix} = \begin{bmatrix} g_c \\ 0 \end{bmatrix}$$

If we consider now the variable substitution

$$\tilde{\sigma}_{1c} = \tilde{\sigma}_c - R_1^{-1}R_3K^\circ R_1\tilde{\sigma}_c \tag{1}$$
$$\tilde{\sigma}_{2c} = K^\circ R_1\tilde{\sigma}_c \tag{2}$$

We can rewrite all of these as a single equation

$$(I_c + M_c^\circ(R_4 - R_2R_1^{-1}R_3)K_c^\circ R_1 + M_c^\circ R_2 - R_1^{-1}R_3K_c^\circ R_1)\tilde{\sigma}_c = g_c$$

5 GFEM Basics

A brief explanation of the Generalized Finite Element Method and the scheme adopted to tackle this problem is presented next.

The Generalized/eXtended Finite Element Method is a way to overcome the difficulties of classical FEM in problems where the solution cannot be well captured by polynomial basis. These include problems where the solution field present sharp gradients, or weak/strong discontinuities. In the particular case of LEFM, classical FEM disadvantages include that its mesh has to fit the crack discontinuity, and use double nodes at it, also, some workarounds has to be done to accurately capture the singular behavior at the crack tip (*e.g.* quarter point elements). Challenge and computational cost increases dramatically in problems involving crack propagation, since the domain has to be remeshed at each step to fit the crack and the solution has to be extrapolated from the old mesh points to the new mesh each time.

In that sense, GFEM/XFEM is superior, since the approximation field can be "enriched" locally using a Partition of Unity Method to resolve fields that are hard to approximate by polynomials. Benefits for LEFM include having a mesh that is independent of the crack geometry by using discontinuous enrichments on nodes of elements completely cut by the crack, and the asymptotic singular behavior at the crack tip can also be approximated by special singular enrichment functions derived from the exact solutions of basic LEFM problems.

The GFEM characteristics adopted for our problem are as follows:

- We've considered 4 regular meshes of triangular elements: 16×16, 32×32, 64×64 and 128×128 (this numbers represent the number of cells of discretization in each direction).
- Underlying polynomial basis $p = 2$ (Quadratic FEM).
- Nodes that lie along the crack surface are enriched with Heaviside functions that replicate the discontinuity.
- Nodes that lie within a radius of 2 units from the crack tip are enriched with singular functions that correspond to the first term of the asymptotic expansion for a Mode I, Mode II and Mode III crack problems.
- Nodes that are both on the crack surface and within the singular enrichment region are enriched with functions described above.

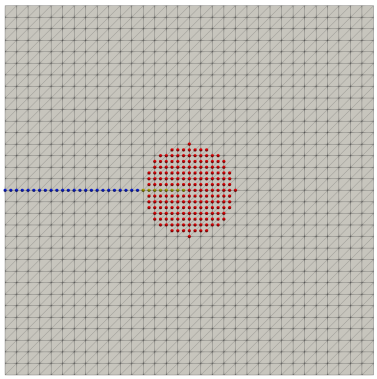


Figure 4: Enrichment strategy adopted in the GFEM solution of our problem. Red spheres represent nodes enriched with singular functions, blue spheres represent nodes enriched with Heaviside functions and yellow spheres represent nodes enriched with both

6 Numerical Implementation

In order to numerically solve the equation, in the last form that it stands, we used a Nystrom scheme using composite Gauss quadrature. The panel discretization of Γ has to be performed such that no corner or junction point lies in the interior of a panel (*i.e.* corners and junctions are always panel endpoints). Equal length panels were chosen for the ease of the implementation. Both considerations imply that the number of panels along Γ is restricted now to the form $9k$ for an positive integer k.

For the discretization we consider two compound operators: $M = -M_3^* \rho$ and $K = M_1 \rho^{-1}$, such that the discretized equation can be read as

$$(I + MK)\Omega = g$$

Despite the appearances, M_3 is the easiest operator to discretize, since it actually has finite value at the boundary, however, M_1 and g are not really defined at the boundary and has to be interpreted in the Cauchy's principal value sense.

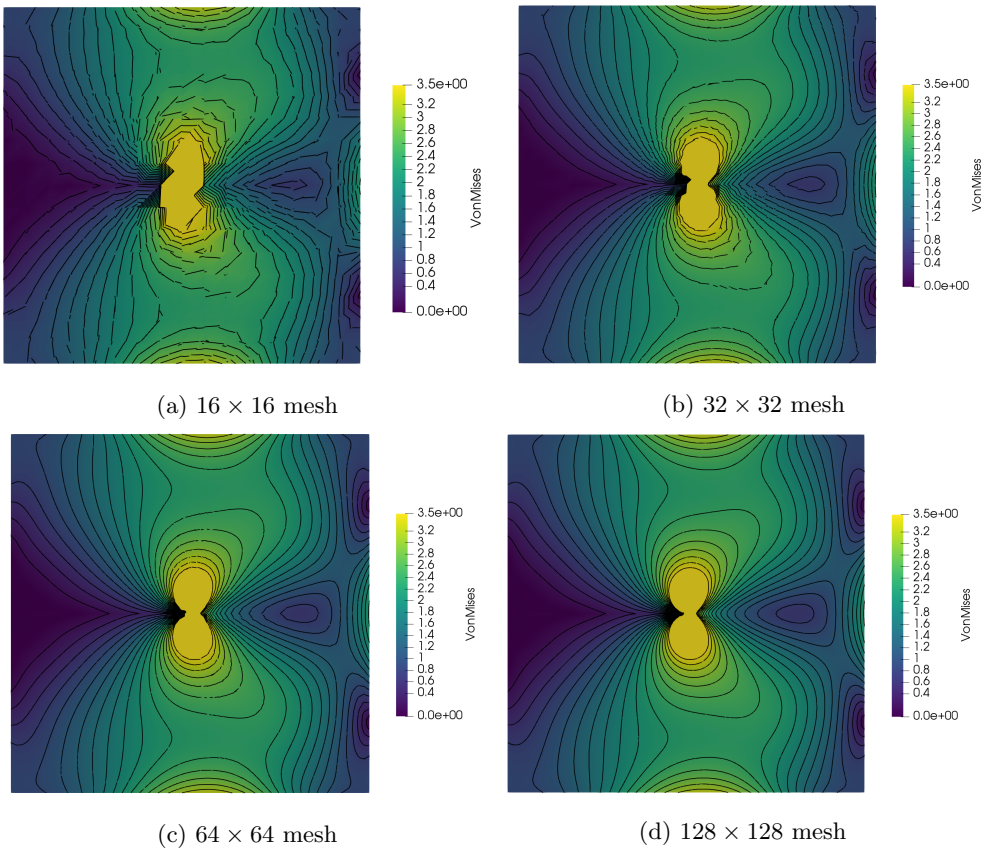
A special 16 point Gauss-Jacobi Quadrature with $\alpha = 0$ and $\beta = -0.5$ is used for the panel closest to the crack tip in order to integrate more accurately the asymptotic singular behavior near this point, while for the rest of the panels we use a regular 16 points Gauss-Legendre quadrature.

To handle the behavior of the density at the corners and junction points, we implement the RCIP method, explained in the previous section. From the RCIP solution of the "coarse" grid problem we can recover the density solution of the associated "fine" grid. We've also considered that an adaptive refinement near the crack tip is important, therefore, we refine towards the crack tip in the same fashion as in the corners and junction of the RCIP method (however, the crack tip refinement is not compressed, leading to a relatively small increase of size of the "coarse" problem while considering more levels of refinement).

For the integral equation formulation, all codes were entirely written in Python, two different codes exist (with and without the RCIP method). The RCIP version had been based on various demo codes available online by Prof. Helsing as aiding material for his tutorial [10]. While for the GFEM counterpart, I'm using the code developed by my research group, written in C++.

7 Results and Conclusions

In this section, we present the results obtained by the Singular Integral Equation scheme and GFEM. The analysis of the results is carried in two parts. First we do a qualitatively analysis of the stress field obtained for each mesh/refinement considered in each method since we know how the stress field looks like for this problem (fig. 2). Second, we obtain the vonMises stress values at several points close to the crack tip for each method and mesh.



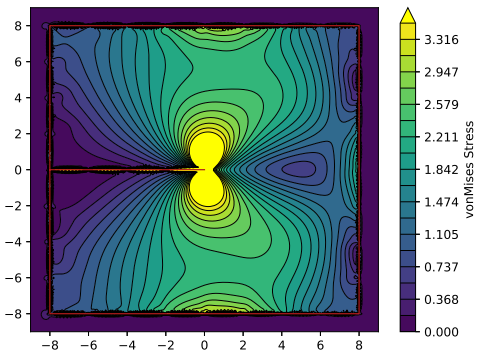
(a) 16×16 mesh

(b) 32×32 mesh

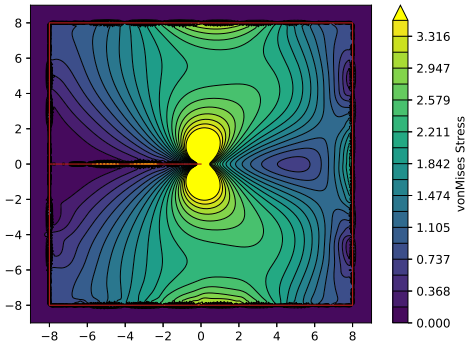
(c) 64×64 mesh

(d) 128×128 mesh

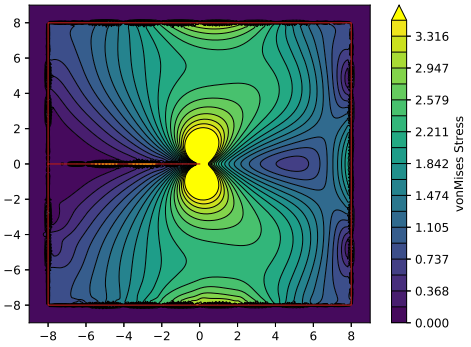
Figure 5: vonMises Stress contour plots for GFEM scheme



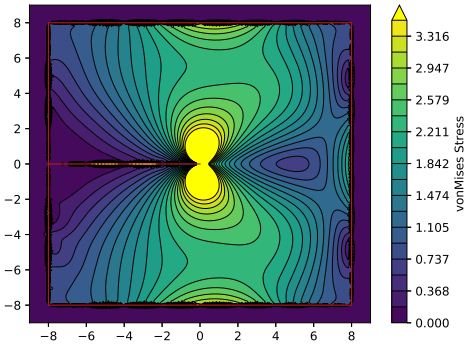
(a) 0 refinement



(b) 2 times refined towards tip, corners and junctions

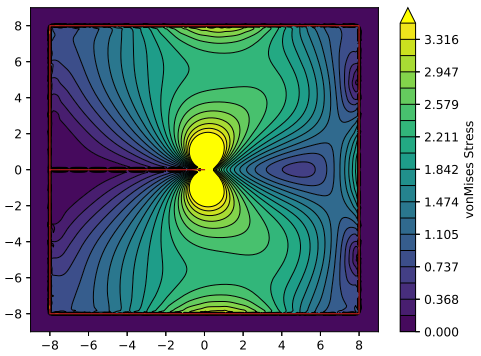


(c) 4 times refined towards tip, corners and junctions

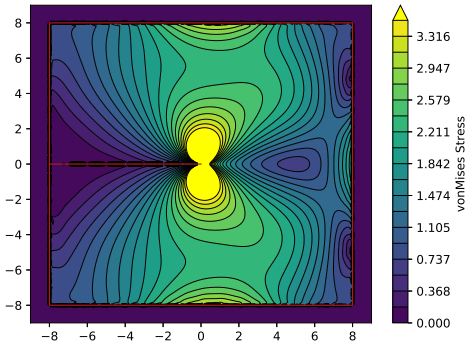


(d) 8 times refined towards tip, corners and junctions

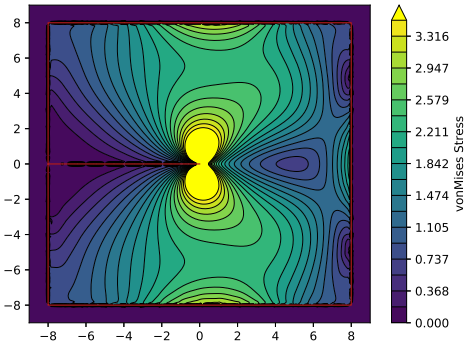
Figure 6: vonMises Stress contour plots for Integral Equation scheme (36 coarse panels)



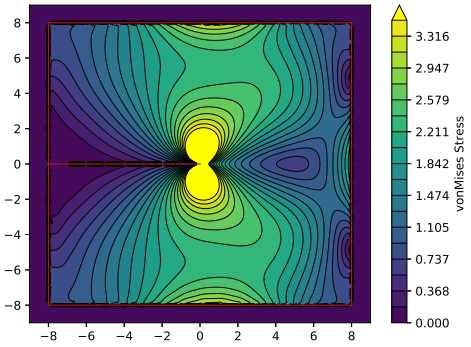
(a) 0 refinement



(b) 2 times refined towards tip, corners and junctions



(c) 4 times refined towards tip, corners and junctions



(d) 8 times refined towards tip, corners and junctions

Figure 7: vonMises Stress contour plots for Integral Equation scheme(72 coarse panels)

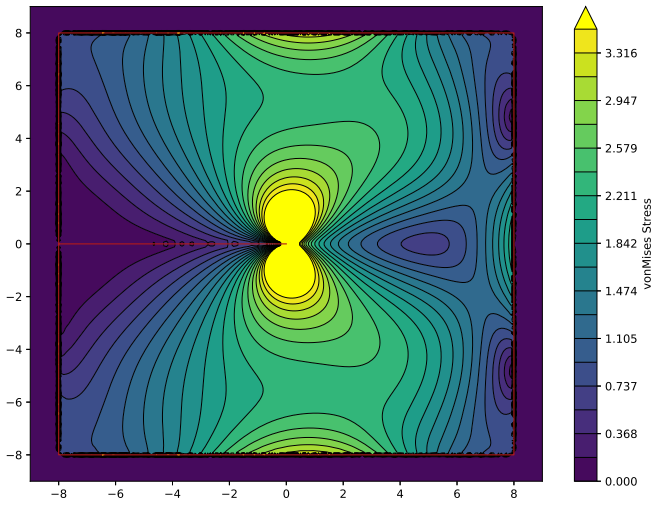


Figure 8: Integral Equation vonMises Stress Field for a 162 uniform panel mesh with 8 recursive refinements towards tip, corners and junctions (4128 points, 8256 degrees of freedom)

From the plots presented, we can observe the following:

- GFEM vonMises solution for the coarsest mesh doesn't seem accurate at all, however, it is due to two reasons: only nodal quantities are dumped to the postprocessor leaving the interpolation between nodal values to it. Since we haven't output other sampled points between nodes, it makes sense that the solution for coarse problems look bad.
- GFEM 64×64 and 128×128 contours already look accurate enough over the whole domain.
- On the integral equation side we have to note that a uniform mesh of 200×200 points between -9 and 9 was sampled for all the cases considered. Notice how that helps us realize that the sampled stresses outside the domain are effectively zero as the equation formulation promised.
- The 36 panel case with no refinement visually presents a bad solution towards the boundary of the domain. Notice that the adaptive refinement seems to improve relatively fast the plot towards the corners, tip and junction, however, there are some portions of the boundary between these critical points that are not still that well represented, which indicates that out base coarse grid is probably not fine enough.
- After the results for the 36 base panel mesh, we opted for a 72 base uniform panel mesh, where we observe an overall better solution even without adaptive refinement. However, notice that some small corner issues are resolved very fast with adaptive refinement.

Finally, we present some quantitatively results in tabular form. Two set of sampling points were considered for the vonMises stress: the first set consists of points lying within one unit directly ahead the crack tip (the asymptotic behavior towards the tip along this line is specially important for crack propagation) while the second set are points lying within one unit directly

above the crack tip. For GFEM we've only considered the nodal output, therefore not all the meshes will contain values for all the points, while for our Integral Equation method all points are always sampled.

			Point							
Mesh	**ndofs**	**nnonzeros**	**(0.125,0)**	**(0.25,0)**	**(0.375,0)**	**(0.5,0)**	**(0.625,0)**	**(0.75,0)**	**(0.875,0)**	**(1,0)**
16x16	2406	34665	-	-	-	-	-	-	-	2.179695
32x32	9302	136801	-	-	-	3.283010	-	-	-	2.197116
64x64	36598	545041	-	4.764499	-	3.286946	-	2.611955	-	2.198705
128x128	145190	2176969	6.809139	4.766188	3.845186	3.287499	2.901343	2.612363	2.384824	2.199105

Table 1: GFEM vonMises stress at selected sample points ahead the crack tip

				Point							
panels	**loc. ref.**	**ndofs_f**	**ndofs_c**	**(0.125,0)**	**(0.25,0)**	**(0.375,0)**	**(0.5,0)**	**(0.625,0)**	**(0.75,0)**	**(0.875,0)**	**(1,0)**
36	0	1152	1152	6.831005	4.776235	3.850694	3.290504	2.902733	2.612619	2.384238	2.197870
	2	1920	1216	6.796478	4.757435	3.838242	3.281587	2.896117	2.607649	2.380523	2.195156
	4	2688	1280	6.801960	4.761236	3.841047	3.283805	2.897968	2.609256	2.381960	2.196470
	8	4224	1408	6.802234	4.758922	3.838188	3.280808	2.894947	2.606249	2.378976	2.193510
72	0	2304	2304	6.831821	4.778890	3.854034	3.294187	2.906611	2.616614	2.388306	2.201982
	2	3072	2368	6.820214	4.772853	3.849898	3.291034	2.904090	2.614555	2.386611	2.200588
	4	3840	2432	6.824574	4.774733	3.850857	3.291576	2.904421	2.614774	2.386770	2.200718
	8	5376	2560	6.828402	4.775707	3.851107	3.291553	2.904278	2.614576	2.386548	2.200488

Table 2: Integral Equation Method vonMises stress at selected sample points ahead the crack tip

			Point							
Mesh	**ndofs**	**nnonzeros**	**(0,0.125)**	**(0,0.25)**	**(0,0.375)**	**(0,0.5)**	**(0,0.625)**	**(0,0.75)**	**(0,0.875)**	**(0,1)**
16x16	2406	34665	-	-	-	-	-	-	-	5.150217
32x32	9302	136801	-	-	-	7.373082	-	-	-	5.113693
64x64	36598	545041	-	10.577610	-	7.371153	-	5.953063	-	5.110803
128x128	145190	2176969	15.119983	10.582510	8.570901	7.372418	6.555542	5.953567	5.486652	5.111102

Table 3: GFEM vonMises stress at selected sample points above the crack tip

				Point							
panels	**loc. ref.**	**ndofs_f**	**ndofs_c**	**(0,0.125)**	**(0,0.25)**	**(0,0.375)**	**(0,0.5)**	**(0,0.625)**	**(0,0.75)**	**(0,0.875)**	**(0,1)**
36	0	1152	1152	15.229772	10.665349	8.644095	7.440003	6.619130	6.013961	5.544300	5.166294
	2	1920	1216	15.077128	10.554757	8.558921	7.371958	6.562013	5.963617	5.498128	5.122687
	4	2688	1280	15.099539	10.591772	8.585877	7.388942	6.572262	5.969720	5.501724	5.124760
	8	4224	1408	15.133143	10.592134	8.580686	7.382432	6.565407	5.962834	5.494931	5.118115
72	0	2304	2304	15.202801	10.638422	8.616682	7.412543	6.591957	5.987305	5.518313	5.141073
	2	3072	2368	15.136566	10.615087	8.609504	7.408979	6.587371	5.980200	5.508505	5.128899
	4	3840	2432	15.188249	10.637869	8.616772	7.410324	6.586324	5.978095	5.505900	5.126051
	8	5376	2560	15.203872	10.640300	8.617014	7.409779	6.585402	5.976968	5.504655	5.124737

Table 4: Integral Equation Method vonMises stress at selected sample points above the crack tip

A few observations about the presented results above:

- We've assumed the crack tip at $(0, 0)$.
- Sadly, there's no exact (analytical) solution for the stress field at a point in the domain for this problem to compare the performance of the methods, but we can observe some agreement between their quantities for the sample points ahead the crack tip, while some less agreement for the sample points above the crack tip.
- Notice that GFEM in general seems to converge to a value from below, while the integral equation method converges from above.
- We shouldn't compare directly number of degrees of freedom between both methods, since GFEM produces a sparse matrix, while IEM produces a dense matrix. Number of nonzero entries of the matrix system (`nnonzeros`) is tabulated in the GFEM scheme which might lead to a fair comparison with the square of the IE degrees of freedom.
- For the IEM both the number of degrees of freedom of the coarse and fine mesh are tabulated. The reason is as follows: Using the RCIP method we technically solve a system of coarse size, but in order to achieve the accuracy of the fine problem, one most recover the densities at the fine size mesh (done recursively) and then sample at every point. Therefore, although we might save computational resources while solving the system for the densities, we don't save anything in postprocessing (there are some quantities of interest that can be obtained directly from the transformed coarse problem directly, but not in our case).
- In the code implementation, I mistakenly recovered the coarse density solution from the transformed coarse problem, leading to exactly the same coarse solution as not doing RCIP (with slightly less number of iterations). The RCIP method itself seems to be implemented correctly, however, implementing the recursion to obtain the fine solution is the same amount of work that took me to make the RCIP method work. Theoretically you can recover the fine solution without information loss, therefore all the results tabulated here are obtained by running a directly refined problem (no RCIP).

References

[1] Helsing, J., Jonsson, A. *On the computation of stress elds on polygonal domains with V-notches*. International Journal for Numerical Methods in Engineering (2002); **53**:433453.

[2] Englund, J. *Fast, accurate, and stable algorithm for the stress field around a zig-zag-shaped crack*. Engineering Fracture Mechanics (2003); **70**:355364.

[3] Englund, J. *Stable algorithm for the stress field around a multiply branched crack*. International Journal for Numerical Methods in Engineering (2005); **63**:926946.

[4] Englund, J. *Efficient algorithm for edge cracked geometries*. International Journal for Numerical Methods in Engineering (2006); **66**:1791-1816.

[5] Englund, J. *A Nystrom scheme with rational quadrature applied to edge crack problems*. Communucations in Numerical Methods in Engineering (2007); **23**:945960.

[6] Helsing, J., Ojala, R. *Corner singularities for elliptic problems: integral equations, graded meshes, quadrature, and compressed inverse preconditioning*. Journal of Computational Physics (2008); **227**:88208840.

[7] Helsing, J. *Integral equation methods for elliptic problems with boundary conditions of mixed type*. Journal of Computational Physics (2009).

[8] Helsing, J., Ojala, R. *Elastostatic computations on aggregates of grains with sharp interfaces, corners, and triple-junctions*. International Journal of Solids and Structures (2009); **46**:44374450

[9] Helsing, J. *A Fast and stable solver for singular integral equations on piecewise smooth curves*. Journal of Scientific Computing (2011); **33**:153-174.

[10] Helsing, J. *Solving integral equations on piecewise smooth boundaries using the RCIP method: a tutorial*. Available through `http://www.maths.lth.se/na/staff/helsing/Tutor`