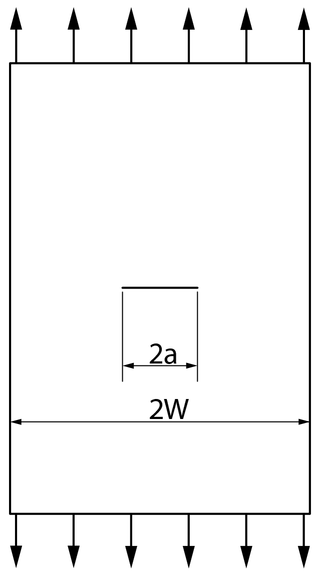
LEFM Analysis of a Center Cracked Specimen

Fergyanto E. Gunawan

Department of Mechanical Engineering

Toyohashi University of Technology



Objectives:

- Using finite element analysis for computing the stress intensity factor.
- Using singular element
- Learn ANSYS/APDL programming

Model Description

Figure 1 shows a center cracked tension specimen. For the data given in Table 1, compute the stress intensity of the opening mode.

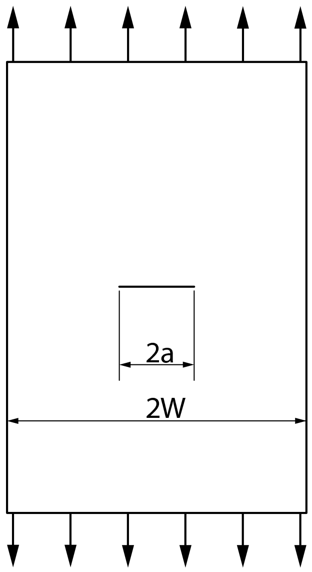


Figure 1: A center cracked tension specimen

Table 1: Data of the center cracked tension specimen.

Parameter	Value
crack length ($2a$)	8.0 in
plate width ($2W$)	20.0 in
plate length ($2L$)	20.0 in
thickness (t)	0.1 in
Young's modulus	1000.0 psi
Poisson's ratio	0.3
Applied force	1.0 lb

References

[1] D. R. J. Owen and A. J. Fawkes. *Engineering Fracture Mechanics: Numerical Methods and Applications.* Pineridge Press Ltd., 1983.

[2] T. L. Anderson. *Fracture Mechanics: Fundamentals and Applications.* CRC Press, second edition, 1995.

Introduction of the Fracture Mechanics

In the classical strength material approach to the structural design, the largest stress in the structure is compared to the material strength via a failure theory for determining the structural safety. However, when a flaw/crack exists, the engineering structure may also fail at a stress lower than the material strength. Therefore, a new approach that considering the flaw is required. The fracture mechanics is a branch in the solid mechanics that takes into account the influence of the flaw to the material strength. Unlike the strength material approach that utilizes the material strength, the fracture mechanics approaches utilizing the fracture toughness in determining the safety of a structure. Figure 2 provides a schematic comparison of the strength material approach to the fracture mechanics approach.

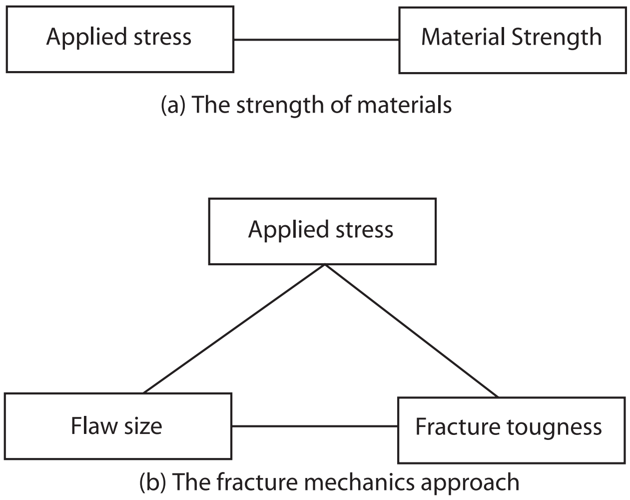


Figure 2: Comparison of the fracture mechanics approach to design with the traditional strength of material approach.

Now, we consider an infinite size two-dimensional plate having a crack as depicted in Fig. 3. At the

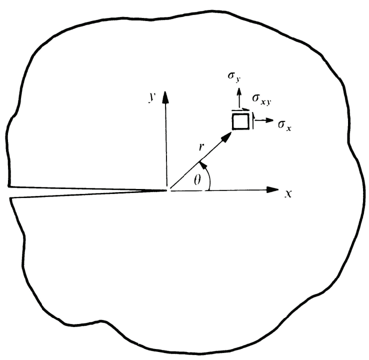


Figure 3: The stresses ahead of the crack tip [1].

vicinity of the crack tip, the stresses and displacements can be expressed as

$$\sigma_x = \frac{K_I}{\sqrt{2\pi r}}\left(1 - \sin\frac{\theta}{2}\sin\frac{3\theta}{2}\right)\cos\frac{\theta}{2} \quad (1)$$

$$\sigma_y = \frac{K_I}{\sqrt{2\pi r}}\left(1 + \sin\frac{\theta}{2}\sin\frac{3\theta}{2}\right)\cos\frac{\theta}{2} \quad (2)$$

and

$$u = \frac{K_I}{4\mu}\sqrt{\frac{r}{2\pi}}\left((2\kappa - 1)\cos\frac{\theta}{2} - \cos\frac{3\theta}{2}\right) \quad (3)$$

$$v = \frac{K_I}{4\mu}\sqrt{\frac{r}{2\pi}}\left((2\kappa + 1)\sin\frac{\theta}{2} - \sin\frac{3\theta}{2}\right), \quad (4)$$

where K_I is the opening mode of fracture, and is the main parameter in the fracture mechanics analysis of linear elastic materials; the parameter μ is $E/(2(1+\nu))$ and the parameter κ is obtained by

$$\kappa = \begin{cases} (3-\nu)/(1+\nu) & \text{for the plane stress} \\ 3 - 4\nu & \text{for the plane strain} \end{cases}. \quad (5)$$

Our focus: To compute the stress intensity factor, K_I

Equations (1)–(4) show that K_I can be expressed in σ_x, σ_y, u, or v; therefore, once we have the data of σ_x, σ_y, u, or v, we may infer the stress intensity factor. As you have seen in five previous modules, those data even for a complex structure can be easily obtained by means of the finite element analysis.

However, Eqs. (1) and (2) reveals that the cracked structure possess a singular stress field that proportional to $1/\sqrt{r}$; in plain English, **the stress gradient at the vicinity of the crack tip is extremely high.** Without any special consideration, deliberately use the finite element method for such a problem would not lead you to accurate data of σ_x, σ_y, u, or v.

The problem of the high stress gradient can be addressed by two approaches. You have used the first approach when dealing with the stress concentration on the plate with hole where the high stress gradient problem exists on the stress distribution along the ligament. The normal stress, as can be seen in Fig. 4, gradually increases as the location approaching the hole edge. On the hole edge, the computed stress is lower than that given by Peterson. However, when the mesh size is reduced as depicted in Fig. 5, the computed normal stress steeply increases to a plateau near the exact solution.

Therefore, it is clear that to accurately simulate the stress singularity, a very fine mesh is required in the region near to the crack tip. In addition, one also may use a special element so called the singular element

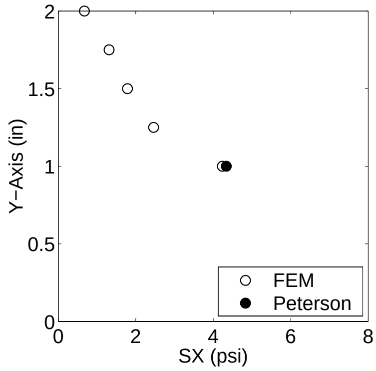


Figure 4: The normal stress along the ligament of the plate with hole.

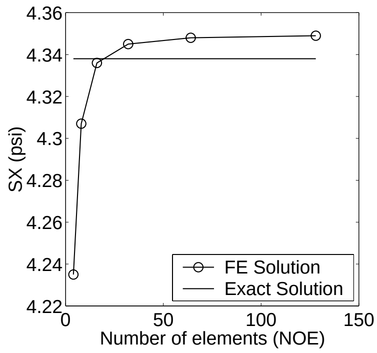


Figure 5: The effect of mesh-size to the largest normal stress on the plate with hole.

that designed to capture the singular stress field.

The aim of the text is to present the most applicable numerical techniques employed by engineers for the solution of practical fracture problems and their implementation using the ANSYS Parametric Design Language (APDL).

The Singular Element

The singular element is an element that possess the strain singularity of $1/\sqrt{r}$. This behavior can be achieved by use a quadratic element where three of their nodes are joined—Nodes 1, 7, and 9 for the case that depicted in Fig. 6—and the mid-side nodes are moved to the quarter point adjacent to the crack tip node.

When the singular element is being used, the stress intensity factors for the tension and shear modes can be directly obtained by solving Eq. (6) and (7):

$$K_I \left\{ \begin{matrix} (2\kappa - 1)\cos\frac{\theta}{2} - \cos\frac{3\theta}{2} \\ (2\kappa + 1)\sin\frac{\theta}{2} - \sin\frac{3\theta}{2} \end{matrix} \right\} = 4\mu\sqrt{\frac{2\pi}{L}} \left\{ \begin{matrix} 4u_2 - u_3 - 3u_1 \\ 4v_2 - v_3 - 3v_1 \end{matrix} \right\}, \tag{6}$$

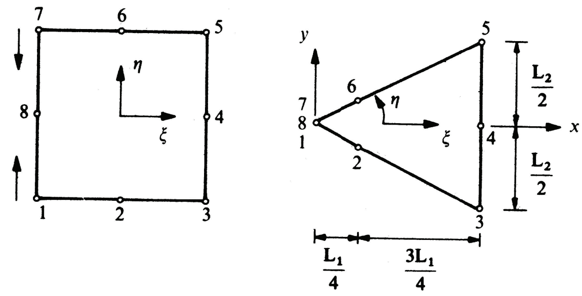


Figure 6: The singular element.

and

$$K_{II}\begin{Bmatrix} -(2\kappa+3)\sin\frac{\theta}{2}-\sin\frac{3\theta}{2} \\ (2\kappa-3)\cos\frac{\theta}{2}-\cos\frac{3\theta}{2} \end{Bmatrix} = 4\mu\sqrt{\frac{2\pi}{L}}\begin{Bmatrix} 4u_2-u_3-3u_1 \\ 4v_2-v_3-3v_1 \end{Bmatrix}, \tag{7}$$

where u_i and v_i is the displacements in x and y directions, respectively, of node i. The index i is 1, 2, or 3.

The best practice for meshing the region surrounding the crack tip in the LEFM is by use a spider-mesh such as that shown in Fig. 7. For the elasto-plastic analysis, the singular element is not required.

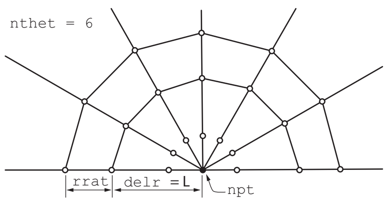


Figure 7: Singular elements around the crack-tip.

ANSYS Implementation of the Singular Element

In ANSYS, the singular element can be generated via by issuing the crack tip location before the mesh is generated:

ANSYS Main Menu: *Preprocessor* ▷ *Meshing* ▷ *Size Cntrls* ▷ *Concentrat KPs* ▷ *Create*

In addition, the element is always accessible via a command line:

```
kscon, npt, delr, kctip, nthet, rrat
```

where `npt` is the keypoint number at concentration, `delr` is L in Eqs. (6) and (7) (see also Fig. 7), `kctip` = 1, `nthet` is number of elements in circumferential direction, and `rrat` is ratio of 2nd row element size to `delr`.

Exact Solution of the Center Cracked Tension Specimen

Reference [2] provides the exact solution of the stress intensity factor as:

$$K_I = \frac{P}{B\sqrt{W}} \cdot f\left(\frac{a}{W}\right), \tag{8}$$

where P is the applied load, B is the specimen thickness, and $f(a/W)$ is given by

$$f\left(\frac{a}{W}\right) = \sqrt{\frac{\pi a}{4W} \cdot \sec\left(\frac{\pi a}{2W}\right)}\left[1 - 0.025\left(\frac{a}{W}\right)^2 + 0.06\left(\frac{a}{W}\right)^4\right]. \tag{9}$$

Finite Element Simulation of the Center Cracked Tension Specimen

Pre-Processing Phase

1. Define some parameters: **ANSYS Pulldown Menu**

   Parameters ▷ Scalar Parameters

   Selection: `L = 10.0`

   Accept

   Selection: `W = 10.0`

   Accept

   Selection: `a = 4.0`

   Accept

   Selection: `thick = 0.1`

   Accept

   Selection: `singularRadius = 0.4*0.1*a`

   Accept

   Selection: `young = 1000.0`

   Accept

   Selection: `nu = 0.3`

   Accept

   Selection: `appliedStress = 1.0`

   Accept

   Close

2. Turn on the keypoint number, the area numbers and the line numbers: **ANSYS Pulldown Menu**

   PlotCtrls ▷ Numbering

   KEYPOINT Keypoint numbers ☒ On

   AREA Area Numbers ☒ On

LINE Line numbers ☒ On

OK

3. Select an element type: **ANSYS Main Menu**

Preprocessor ▷ Element Type ▷ Add/Edit/Delete

Add

Solid | 8node 183

OK

Options

Element behavior K3 : Plane strs w/thk

OK

Close

4. Define the plate thickness: **ANSYS Main Menu**

Preprocessor ▷ Real Constants ▷ Add/Edit/Delete

Add

Type 1 PLANE183

OK

Thickness THK `thick`

OK

Close

5. Define material properties: **ANSYS Main Menu**

Preprocessor ▷ Material Props ▷ Material Models

Structural ▷ Linear ▷ Elastic ▷ Isotropic

EX young

PRXY `nu`

OK

Material ▷ Exit

6. Create two rectangles: **ANSYS Main Menu**

Preprocessor ▷ Modeling ▷ Create ▷ Areas ▷ Rectangle ▷ By 2 Corners

WP X	-a
WP Y	0
Width	W
Height	L

Apply

7. Create two keypoints: **ANSYS Main Menu**

Preprocessor ▷ Modeling ▷ Create ▷ Keypoints ▷ In Active CS

NPT Keypoint number	101		
X, Y, Z Location in active CS	0.2*a	0.0	0.0

Apply

NPT Keypoint number	102		
X, Y, Z Location in active CS	0.2*a	L	0.0

Apply

NPT Keypoint number	103		
X, Y, Z Location in active CS	0.0	0.0	0.0

Apply

NPT Keypoint number 104

X, Y, Z Location in active CS	0.0	L	0.0

Apply

NPT Keypoint number	105		
X, Y, Z Location in active CS	-0.2*a	0.0	0.0

Apply

NPT Keypoint number	106		
X, Y, Z Location in active CS	-0.2*a	L	0.0

Apply

NPT Keypoint number	107		
X, Y, Z Location in active CS	-a	+0.2*a	0.0

Apply

NPT Keypoint number	108		
X, Y, Z Location in active CS	W-a	+0.2*a	0.0

OK

8. Create four lines: **ANSYS Main Menu**

Preprocessor ▷ *Modeling* ▷ *Create* ▷ *Lines* ▷ *Lines* ▷ *Straight line*

< Pick Keypoints: 101, and then 102 >

< Pick Keypoints: 103, and then 104 >

< Pick Keypoints: 105, and then 106 >

< Pick Keypoints: 107, and then 108 >

Cancel

9. Break down the Area A1: **ANSYS Main Menu**

Preprocessing ▷ *Modeling* ▷ *Operate* ▷ *Booleans* ▷ *Divide* ▷ *Area by Line*

[Pick All]

[Pick All]

10. Create a circle: **ANSYS Main Menu**

Preprocessor ▷ *Modeling* ▷ *Create* ▷ *Area* ▷ *Circl* ▷ *Partial Annulus*

WP X	0.0
WP Y	0.0
Rad-1	0.0
Theta-1	0.0
Rad-2	0.1*a
Theta-2	90

[OK]

11. Subtract Area A5 by A4: **ANSYS Main Menu**

Preprocessing ▷ *Modeling* ▷ *Operate* ▷ *Booleans* ▷ *Substract* ▷ *With Options* ▷ *Areas*

< Pick Area A6 >

[OK]

< Pick Area A1 >

[OK]

KEEP2 Subtracted areas will be [Kept]

[OK]

12. Merges coincident Keypoints: **ANSYS Main Menu**

Preprocessor ▷ Numbering Ctrls ▷ Merge Items

Label Type of item to be merge | Keypoints |

| OK |

13. Create a line: **ANSYS Main Menu**

Preprocessor ▷ Modeling ▷ Create ▷ Lines ▷ Lines ▷ Straight line

< Pick Keypoints: 10, and then 6 >

| Cancel |

14. Divide Area A1 and A10: **ANSYS Main Menu**

Preprocessing ▷ Modeling ▷ Operate ▷ Booleans ▷ Divide ▷ Area by Line

< Pick Area A10 and A1 >

| OK |

< Pick Line L6 >

| OK |

15. Define the vertex of the singular elements: **ANSYS Main Menu**

Preprocessing ▷ Meshing ▷ Size Cntrls ▷ Concentrat KPs ▷ Create

< Pick Keypoint 10 >

| OK |

DELR Radius of 1st row of elems | singularRadius |

NTHET No of elems around circumf | 2 |

KCTIP midside node position | Skewed 1/4pt |

| OK |

16. Control the mesh density: **ANSYS Pulldown Menu**

    Preprocessor ▷ *Meshing* ▷ *MeshTool*

    Lines [Set]

    < Pick Lines L7 and L24 >

    NDIV No. of element divisions [2]

    [OK]

17. Mesh the Areas A6 and A11: **ANSYS Main Menu**

    Preprocessor ▷ *Meshing* ▷ *Mesh* ▷ *Free*

    < Pick Area A6 and Area A11 >

18. Select the Area A12 and A13: **ANSYS Pulldown Menu**

    Select ▷ *Entities*

    [Area]

    [OK]

    < Pick A12 and A13 >

    [OK]

19. Plot the selected Areas: **ANSYS Pulldown Menu**

    Plot ▷ *Area*

20. Select everything under the selected areas: **ANSYS Pulldown Menu**

    Select ▷ *Everything Below* ▷ *Selected Area*

21. Mesh the selected areas of A12 and A13: **ANSYS Pulldown Menu**

Preprocessor ▷ Meshing ▷ MeshTool

Lines [Set]

[Pick All]

NDIV No. of element divisions [2]

[OK]

Mesh: [Areas]

Shape: [• Quad]

Shape: [• Mapped]

[Mesh]

< Pick All >

[Close]

22. Select everything: **ANSYS Main Menu**

Select ▷ Everything

23. Mesh the Area A7: **ANSYS Main Menu**

Preprocessor ▷ Meshing ▷ MeshTool

Lines [Set]

< Pick Lines L26 and L19 >

NDIV No. of element divisions [10]

SPACE Spacing ratio [8]

[Apply]

< Pick Line L27 >

NDIV No. of element divisions [2]

SPACE Spacing ratio [1]

[OK]

Mesh: [Areas]

Shape: [• Quad]

Shape: [• Mapped]

[Mesh]

< Pick the Area A7 >

[Close]

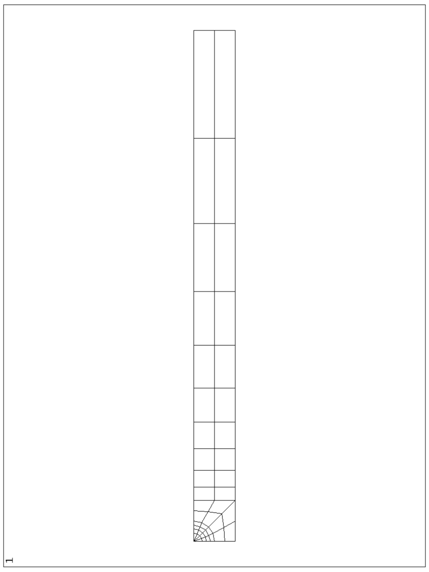


Figure 8: Mesh of a small portion of the center cracked tension specimen.

24. Reflect the areas that contain elements: **ANSYS Main Menu**

Preprocessor ▷ *Modeling* ▷ *Reflect* ▷ *Area*

< Pick Areas A6, A11, A12, A13, and A7 >

25. Merge nodes and keypoints: **ANSYS Main Menu**

Preprocessor ▷ *Numbering Ctrls* ▷ *Merge Items*

Label Type of item to be merge [Nodes]

[Apply]

Label Type of item to be merge [Keypoints]

[OK]

26. Mesh the rest of the areas: **ANSYS Main Menu**

Preprocessor ▷ *Meshing* ▷ *MeshTool*

Lines [Set]

< Pick Lines L18 and L21 >

NDIV No. of element divisions [10]

SPACE Spacing ratio [1/8]

[Apply]

< Pick Lines L13 and L17 >

NDIV No. of element divisions [5]

SPACE Spacing ratio [1/8]

[Apply]

< Pick Line L16 >

NDIV No. of element divisions [5]

SPACE Spacing ratio [8]

[Apply]

< Pick Lines L14 and L10 >

NDIV No. of element divisions [2]

SPACE Spacing ratio [1]

Apply

< Pick Lines L9 and L20 >

NDIV No. of element divisions: 4

SPACE Spacing ratio: 1/4

Apply

< Pick Line L12 >

NDIV No. of element divisions: 4

SPACE Spacing ratio: 1/4

OK

Mesh: Areas

Shape: • Quad

Shape: • Mapped

Mesh

< Pick the Area A3, A4, A2, and A5 >

Close

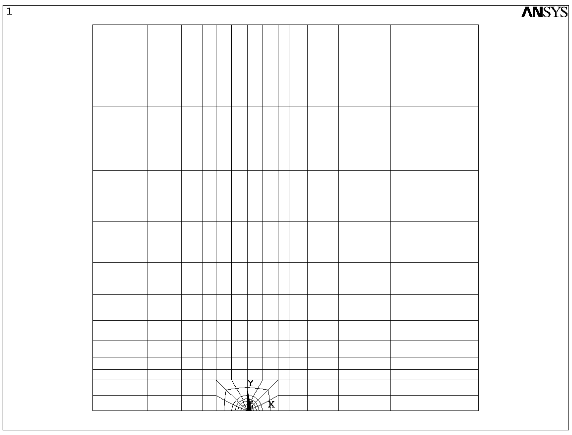


Figure 9: Finite element mesh of the center cracked tension specimen.

Solution Phase

1. Define the analysis type: **ANSYS Main Menu**

   Solution ▷ *Analysis Type* ▷ *New Analysis*

   [ANTYPE] Type of analysis: [• Static]

   [OK]

2. Applied the uniform stress on the specimen end: **ANSYS Main Menu**

   Solution ▷ *Define Loads* ▷ *Apply* ▷ *Structural* ▷ *Pressure* ▷ *On Lines*

   < Pick lines L20, L30, L27, and L17 >

   [OK]

   VALUE Load PRES value [-appliedStress]

   [OK]

3. Applied the symmetry constraints

   Solution ▷ *Define Loads* ▷ *Apply* ▷ *Structural* ▷ *Displacement* ▷ *Symmetry* ▷ *B.C On Lines*

   < Pick lines L3, L5, L13, and also L10 and L21 >

   [OK]

4. Solve: **ANSYS Main Menu**

   Solution ▷ *Solve* ▷ *Current LS*

Post Processing Phase

Checking Singular Stress Field

Equation 6 implies

$$\sigma_y \sim \frac{1}{\sqrt{r}} \tag{10}$$

at the vicinity of the crack tip. A good finite element model should be able to capture the stress singularity. Therefore, For case of LEFM: check the stress singularity

In the practical application, the σ_y is often plotted against $\sqrt{r}$ in the logarithmic scale. For the present case, we obtain a singular stress field as shown in Fig. 10. The figure uncovers two aspects: (i) the term

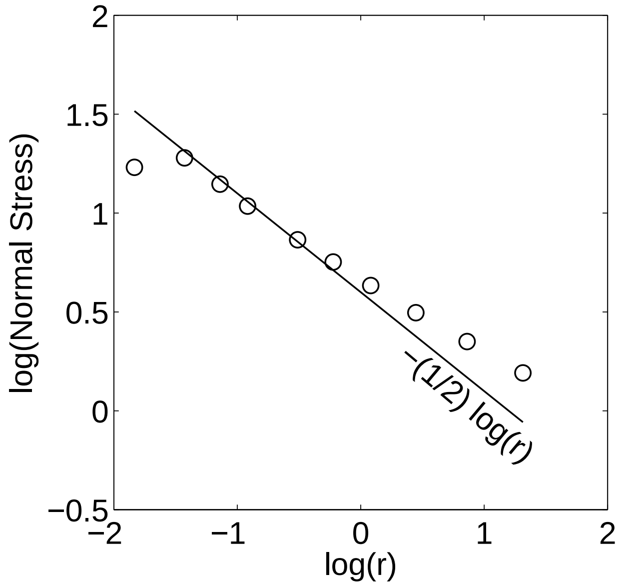


Figure 10: The singular stress field near vicinity of the crack-tip.

vicinity for the present case is a region that $\log(r) < 0$; and (ii) the present model does not entirely accurate in capturing the singular stress field.

Stress Intensity Factor

With a spider-mesh such that shown in Fig. 11, we may compute the stress intensity factor for at various angles: $0, 22.5, \ldots, 180$ degrees.

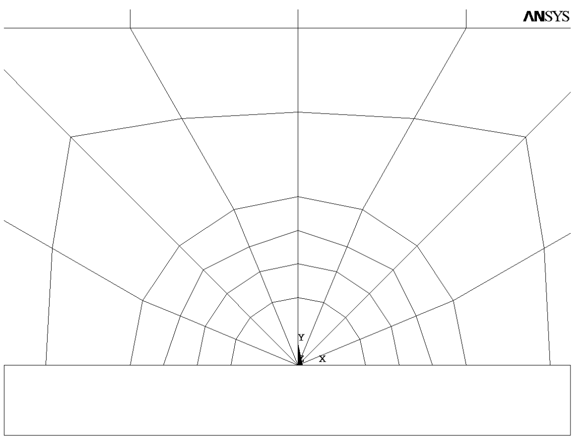


Figure 11: The spider-mesh surrounding the crack tip.

Equation (6) has been implemented in an ANSYS-APDL code, presented in Appendix B, such that the function will return the stress intensity factor for given some necessary data. The use of the macro is summarized in Table 2.

Table 2: Summary of `SIFsingular.mac`

Syntax	: `SIFsingular, nodeCenter, Radius, Angle, Young, nu`
Description	: The macro `SIFsingular.mac` computes the SIF based on Eq. (6). The macro requires five input arguments: `nodeCenter` is the node where the crack tip is located; `Radius` is the length, L, of the singular element; `Angle` is an angle measured from $x-$axis; `Young` is the Young modulus; and `nu` is the Poisson's ratio
Outputs	: output is written to K1 and K12

Following, we provide an ANSYS code that utilizes `SIFsingular.mac` in computing the stress intensity factor at various angles. The results are presented in Table 3.

```
/post1
    nodeCenter = node(0.0, 0.0, 0.0)      ! Node at the crack-tip
    exact_sif, a, w, thick, appliedStress*thick*2*w, 'cct'
    theta = 0.0                           ! The first angle
    *cfopen, SIFbySingularElements, dat   ! Open a file for output
    *do, i, 1, 9                          ! Looping for angles: 0, 30, ..., 180 deg
        SIFSingular, nodeCenter, singularRadius, theta, Young, nu
        error1 = abs(kExact - K1)/kExact*100.0
        error2 = abs(kExact - K12)/kExact*100.0
        *vwrite, theta, kExact, K1, error1, K12, error2   ! Print outputs to the file
        (F5.1, 2X, F5.3, 2X, F5.3, 2X, F5.1, 2X, F5.3, 2X, F5.1)
        theta = theta + 22.5              ! Increase theta by 22.5 deg.
    *enddo
    *cfclos                               ! Close the file
finish
```

Table 3: The exact and computed stress intensity factors.

Theta	Exact SIF	Estimated SIF			
(degree)	(psi$\sqrt{\text{in}}$)	(psi$\sqrt{\text{in}}$)	(%)	(psi$\sqrt{\text{in}}$)	(%)
0.0	4.149	4.255	2.6	0.000	100.0
22.5	4.149	4.272	3.0	4.127	0.5
45.0	4.149	4.310	3.9	4.183	0.8
67.5	4.149	4.349	4.8	4.238	2.1
90.0	4.149	4.380	5.6	4.279	3.2
112.5	4.149	4.405	6.2	4.308	3.8
135.0	4.149	4.426	6.7	4.327	4.3
157.5	4.149	4.455	7.4	4.339	4.6
180.0	4.149	0.000	100.0	4.342	4.7

Appendix A: A Macro for Computing the Exact Solution

```
/nopr
*if, arg1, eq, 911, then
   /com exact_sif - exact sif
   /com
   /com  DESCRIPTION : To compute the exact SIF for various standard
   /com                specimen
   /com
   /com  USAGE       : exact_sif, crack, width, thickness, p, specimen
   /com
   /com                crack     : the crack length. For cct, half of the crack length
   /com                width     : specimen width, for cct and dent, the actual width is 2*width
   /com                thickness : specimen thickness
   /com                p         : applied load
   /com                specimen  : 'sent' for
   /com                            'cct' for the center cracked tension specimen
   /com                            'dent' for
   /com                            'cts' for for the compact tension specimen
   /com
   /com  OUTPUT      : kExact
   /com
   /com  REFERENCE   : Fracture Mechanics: Fundamentals and Applications
   /com                T. L. Anderson. Table 2.4. P.63
   /com
   /com  Fergyanto E Gunawan (gunawan@mech.tut.ac.jp)
   /com  Mechanical Engineering Department, TUT
   /com  Sunday, November 28, 2004
*else

    *afun, rad
    *msg, info
    *** the macro turns the unit of angle to radian ***

    _crack = arg1
    _width = arg2
    _thickness = arg3
    _p = arg4
    _specimen_type = arg5
    pi = acos(-1)

    _aw = _crack/_width
    _aw2 = _aw*_aw
    _aw3 = _aw2*_aw
    _aw4 = _aw2*_aw2

    _force = _p/(_thickness*sqrt(_width))

    *if, _specimen_type, eq, 'sent', then
        _top_part = sqrt( 2*atan(0.5*pi*_aw) )
        _bot_part = cos( 0.5*pi*_aw )
        _last_part = 0.752 + 2.02*_aw + 0.37*(1-sin(0.5*pi*_aw))*(1-sin(0.5*pi*_aw))*(1-sin(0.5*pi*_aw))
        _beta = _top_part/_bot_part*_last_part
        kExact = _force*_beta
    *endif
    *if, _specimen_type, eq, 'cct', then
        _term_1 = sqrt(0.25*pi*_aw/(tan(0.5*pi*_aw)))
        _term_2 = 1 - 0.025*_aw2 + 0.06*_aw4
        _beta = _term_1*_term_2
        kExact = _force*_beta
    *endif
    *if, _specimen_type, eq, 'dent', then
        _top_part = sqrt( 0.5*pi*_aw )
        _bot_part = sqrt( 1 - _aw )
        _last_part = 1.122-0.561*_aw-0.205*_aw2+0.471*_aw3+0.190*_aw4
        _beta = _top_part/_bot_part*_last_part
        kExact = _force*_beta
    *endif
    *if, _specimen_type, eq, 'cts', then
        _top = 2+_aw
        _bot = sqrt( (1-_aw)*(1-_aw)*(1-_aw) )
        _last = 0.886+4.64*_aw-13.32*_aw2+14.72*_aw3-5.6*_aw4
        _beta = _top/_bot*_last
        kExact = _force*_beta
    *endif
    *msg, info
    *** output of the macro is kExact ***
*endif
/gopr
```

Appendix B: A Macro for Computing SIF

```
/nopr
*get,_ar20,active,,rout
*if, arg1, eq, 911, then
    /com,
    /com, *********************************************************************
    /com,
    /com, DESCRIPTION :
    /com,
    /com, USAGE       : SIFsingular, nodeCenter, Radius, Angle, Young, nu
    /com,
    /com, WHERE       : Angle in degree
    /com,
    /com, OUTPUT      : output is written to K1 and K12, see the equation
    /com,
    /com, AUTHOR      : Fergyanto E. Gunawan (gunawan@mech.tut.ac.jp)
    /com,               Department of mechanical engineering, tut
    /com,
    /com,
    /com, *********************************************************************
    !
    ! Lesson from a short intention:
    !   I wanna a short code; but the only way ended up in an irrational code.
    !   FEG.06.27.06
    ! 10 lessons:
    !  1. Avoid using /eof
    !  2. No Tab, instead of Space
    !  Lessons 2--10 come later. :)
    !
*elseif,_ar20,ne,31
    *msg, error
    ***macro must be used in /post1 ***
*else
    ! DEFINE A CONSTANT
    *afun, rad
    _pi = acos(-1.0)

    ! PREFER TO USE LOCAL VARIABLES INSTEAD OF MEANINGLESS VARIABLE NAME
    _nodeCenter = arg1
    _radius = arg2
    _theta = arg3*_pi/180.0     ! Convert from degree to rad
    _young = arg4
    _nu = arg5

    _kappa = (3 - _nu)/(1 + _nu)
    _mu = _young/(2*(1 + _nu))

    ! GET THE DISPLACEMENTS AT THE SINGULAR ELEMENTS
    _node1 = _nodeCenter
    _node2 = node(0.25*_radius*cos(_theta), 0.25*_radius*sin(_theta), 0.0)
    _node3 = node(_radius*cos(_theta), _radius*sin(_theta), 0.0)
    _u1 = ux(_node1)
    _u2 = ux(_node2)
    _u3 = ux(_node3)
    _v1 = uy(_node1)
    _v2 = uy(_node2)
    _v3 = uy(_node3)

    ! Computing distance between Node 1 and Node 3
    _dx = nx(_node1) - nx(_node3)
    _dy = ny(_node1) - ny(_node3)
    _L = sqrt(_dx*_dx + _dy*_dy)

    _left1 = (2*_kappa - 1)*cos(0.5*_theta) - cos(1.5*_theta)
    _right1 = 4*_mu*sqrt(2*_pi/_L)*( 4*_u2 - _u3 - 3*_u1 )
    _eps = 1.0e-10
    *if, abs(_left1), lt, _eps,then
        K1 = 0.0
    *else
        K1 = _right1/_left1
    *endif

    _left = (2*_kappa + 1)*sin(0.5*_theta) - sin(1.5*_theta)
    _right = 4*_mu*sqrt(2*_pi/_L)*(4*_v2 - _v3 - 3*_v1)
    *if, abs(_left), lt, _eps, then
        K12 = 0.0
    *else
        K12 = _right/_left
    *endif
*endif
```

```
/gopr
```