VOL. 11, NO. 1, JANUARY 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

AN IMPROVED ALGORITHM OF FINITE ELEMENT MESH GENERATION FOR FRACTURE MECHANICS ANALYSIS

K. H. Leong and F. Yusof
School of Mechanical Engineering, University of Science Malaysia, NibongTebal, Penang, Malaysia
E-Mail: mefeizal@usm.my

ABSTRACT

The purpose of this paper is to propose a set of algorithms that can be used to develop an automated crack-tip finite element mesh for numerical fracture mechanics analysis. The algorithms were developed in MATLAB based on a two-dimensional boundary layer formulation. It was further shown that the nodes and elements of the models can be numbered in a consistent pattern by controlling the order of generating the nodes and the elements in the models. It was also demonstrated that the nodes can be connected in a systematic order to form second order quadrilateral element using the proposed algorithms.

Keywords: fracture mechanics, finite element analysis, matlab algorithms.

INTRODUCTION

Computational fracture mechanics through finite element method has developed extensively in the last few decades, following the rapid revolution in the computation capability of computers. The finite element computation output could be the determination of elastic stress intensity factor [1], elastic-plastic energy release rate from crack domain (J-integral) [2], the stress-strain deformation field of the near crack fields and so on.

All these numerical determination can be conducted either in two dimensional plane problems (2D) or using three dimensional fields (3D). In plane problem, the common method used to analyze the near tip crack field is boundary layer formulation (BLF). Concept of BLF was firstly introduced in [3] to take the crack tip vicinity as a semi-circular region and apply the loading conditions, i.e. displacement at the outer boundary. This method is then improved later in [4] by incorporating T stress into the boundary condition, and it is then renamed as modified boundary layer formulation (mBLF). From here, the modified BLF was used extensively in computational fracture mechanics. Henceforth, the basis for the development of several important concept in the field of fracture mechanics was developed, such as the two parameterization for crack tip fields (K-T[5], J-T[6, 7], J-Q [8, 9] and J-A_2[10]). To validate the solutions from mBLF, usually a full field solution can be used as an alternative. The full field solution in plane conditions was provided by designing a finite element model with the shape of an actual specimen under fracture strength testing. There are a few standard specimens listed in ASTM E1820 to be used in fracture strength testing, which includes single edged cracked bending specimen (SENB), center-cracked specimen in tension (CCP), compact tension specimen (CT) and etc.

For fracture mechanics in three-dimensional problems, similar model used in plane problem is modelled with thickness. Since thicknesses are associated in 3D model, it can captures changes of stress and strain along the thickness of the studied structure. Therefore, the three dimensional solution can provide a detailed solutions compared to plane solution. Recent focus in fracture mechanics had been shifted to crack tip deformation in three dimensional problems[11]. New parameters such as T_z[12, 13] and T_{33}[14] were introduced to characterize the stress-strain deformation field in three dimensional fracture problem.

Usually, prior to modelingfinite element analysis in 2D or 3D, a model is required and can be modeled via commercial finite element package, i.e ADINA, ANSYS, ABAQUS. However, some human intervention such as element aspect ratio, height to width ratio of specimen, and others are still needed to have a properly designed model [15]. Among them, the arrangement and numbering system of nodes is vital for post processing of finite element analysis because exact nodes must be known beforehand in order to extract data from a specific location in a model, e.g. at a fixed distance from the crack tip. Therefore, the numbering system for nodes and elements must be planned ahead carefully during the modeling process. However, the process of setting up the finite element mesh in commercial finite element package can be laborious.

In this paper, a set of algorithmsused to create full field single edge cracked model in plane problems are discussed. These algorithms were proposed to simplify the automaticmesh generation procedure for fracture mechanics finite element computation in ABAQUS. The proposed algorithms are applicable to any single edge specimen with a crack length to ligament ratio, a/W=0.5. The crack tip domain used a “spider web” mesh design, which can be implemented separately in a modified boundary layer formulation. The model created using the algorithm presented could also be modified into a three-dimensional model for other related problems.

MODEL GEOMETRY

A full field model for a single edge cracked specimen is presented in this study as shown in Figure-1a. Symmetrical features are used and only one half of the full model was used in the study (shaded region). The model was divided into four regions for meshing convenience. The first region is the intermediate vicinity of the crack-tip. The second region takes a box like shape while encompassing the first region. The third region is the bottom section of the specimen which encloses the first and second region. The fourth region is the top section of the specimen, where load, *P* is usually applied at the top end of this region. The algorithms presented in this paper were designed in MATLAB to complement all regions conditionally.The conditions were based on completion of a region before proceeding to the next region. The description of the processes is presented in Figure-2.

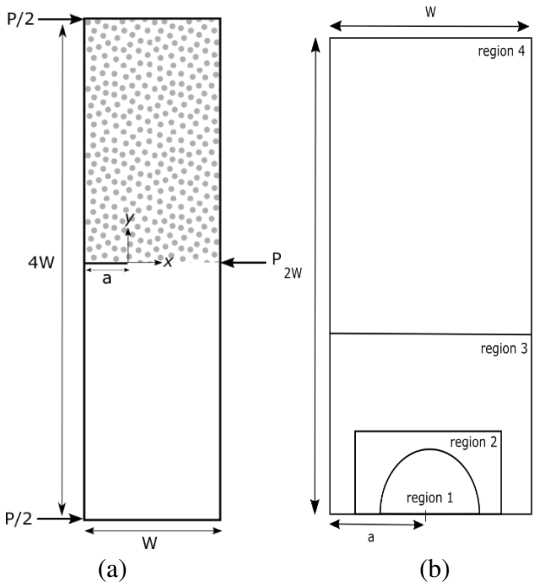

Figure-1. (a) Geometry studied (only shaded region is considered) **(b)** The four regions in model.

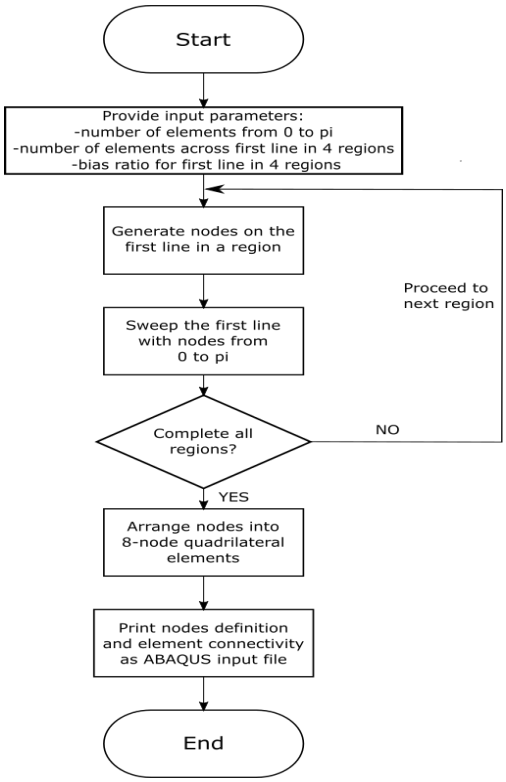

Figure-2. Process flow of the designed algorithms.

NODES GENERATION

The algorithm discussed in this work generates nodes in a consistent pattern for all regions. The origin labelled as the first node was located at the crack tip in region 1. This region wasoutlined as the crack tip domain of the model. The model presented was designed for elastic-perfectly plastic analysis, thus defining 1/r singularity is sufficient to capture the strain singularity in a collapsed quadrilateral element at crack tip. Such definition can be achieved by placing midside nodes at the middle between corner nodes [15]. The entire node generation process begins with constructing a horizontal line with nodes (representing region ahead of crack tip at 0 degree) as depicted in Figure-3.

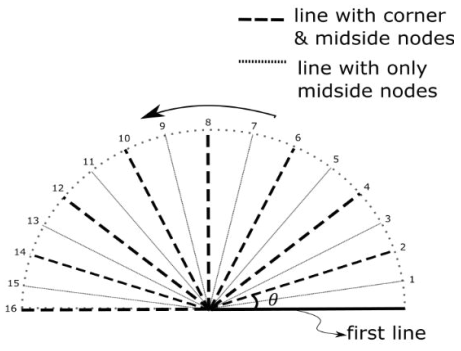

Figure-3. First line constructed in region 1(0°)

The first line was divided into segments according to the number of elements across the line, N and bias ratio, b. Bias seeding was achieved using element length ratio, c through Equation (1).

$$c = b^{(\frac{1}{N-1})} \quad (1)$$

Moving away from origin (0, 0), the initial distance between corner nodes, k will increase by factor c as seen in Figure-4. The total length of the first line equals to the radius of region 1.

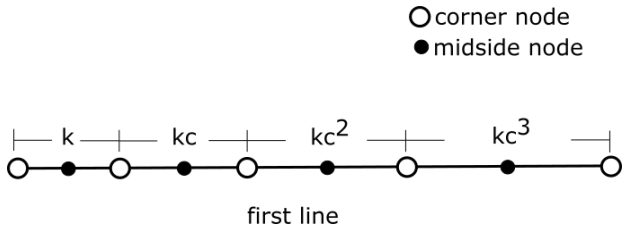

Figure-4. Nodes generated in the first line with bias seeding.

The nodes on the first line can be created using algorithmA1 as follow:

```
%Algorithm A1
origin=[0 0]; %set origin
ll=10;        %set length
br=2;         %set bias ratio
No_ele=3;     %set number of element
Cc=br^(1/(No_ele-1)); %length ratio
k =ll/sum(Cc.^[0:No_ele-1]); %get k
hk1=1/2*(Cc^(0:No_ele-1)*k);
hk2=hk1;
n_line=[ha1; ha2];
node_dist=n_line(:)';
for n from 1 to 2*No_ele
    xcoord_1(n)=origin(1,1)+
sum(node_dist(1:n));
    ycoord_1=zeros(1,2*No_ele);
end
```

Through algorithm A1, the nodes were generated starting from the origin (0, 0), and eventually ended at the end of first line. Similar process was repeated to compute the coordinates of nodes and arrange themat another line at a fixed angle interval. The lines were generated in a counter clockwise manner. The number of lines was decided by the number of elements needed within the angle from 0 to π. An algorithm as described below was written to describe this repeating process.

```
%Algorithm A2
E_w=8; %number of elements from 0 to pi
tn=E_w*((2*No_ele)+(2*No_ele+1));
no_line=2*E_w;
thetta=180/E_w;
node=zeros(tn,2)
a1=a2=0; %initialize a1 and a2
for ii from 1 to no_line
if rem(ii,2)~=0
    u1=thetta/2*(ii);
fori=1:No_ele+1
       w1=(a1*(No_ele+1))+
       (a2*(2*No_ele+1))+i;
       mn=i*2-1;
node(w1,1)=
       xcoord_1(mn,1)*cosd(u1);
node(w1,2)=
       xcoord_1 (mn,1)*sind(u1);
end
   a1=a1+1;
else
     u2=thetta*(ii/2);
fori=1:2*No_ele+1
       w1=(a1*(No_ele+1))+
          (a2*(2*No_ele+1)+i;
node(w1,1)= xcoord_1 (i,1)*cosd(u2);
node(w1,2)= xcoord_1 (i,1)*sind(u2);
end
   a2=a2+1;
end
end
```

Moving to region 2, the coefficients of the first degree of polynomial were firstly computed using the origin (0, 0) and each nodes on outermost boundary of region 1. The resulted coefficients were a slope, m and the y-intercept, c which can be expressed as:

$$y = mx + c \quad (2)$$

With known height of y1 and width of x1 values for the box region, as shown in Figure-5, the x and y-coordinates of nodes on the outermost boundary in region 2 can be calculated using the coefficients in Eqn. (2). The process of determining nodes on the outermost boundary can be achieved using algorithmA3 as shown below:

```
%AlgorithmA3
angle=linspace(0,pi,2*E_w+1);
outer_x1=ll*cos(angle);
outer_y1=ll*sin(angle);
fori from 1 to 2*E_w+1
rw=find(node(:,1)==outer_x1(i)
       &node(:,2)==outer_y1(i);
row(i)=rw;
end
outer_r1=node(row,:)
x1=100; %set width
y1=100; %set height
b=0;   %initialize b
fori=0:thetta/2:180
ifi==0
```

```
b=b+1;
    bd2_y(b)=0;
     bd2_x(b)=x1;
elseifi==90
b=b+1;
        bd2_y(b)=y1;
        bd2_x(b)=0;
elseifi==180
b=b+1;
        bd2_y(b)=0;
        bd2_x(b)=-1*x1;
elseifi>=135
b=b+1;
gc=polyfit(outer_r1(b,1),
             outer_r1(b,2),1);
        bd2_y(b)=polyval(gc,-1*x1);
        bd2_x(b)=-1*x1;
elseifi<=45
b=b+1;
gc=polyfit(outer_r1(b,1),
      outer_r1(b,2),1);
        bd2_y(b)=polyval(gc,x1);
        bd2_x(b)=x1;
else
     a=a+1;
gc=polyfit(outer_r1(b,1),
      outer_r1(b,2),1);
     bd2_y(b)=yval;
     bd2_y(b)=(outer_r1(b,2)-gc(2))/gc(1);
end
end
```

Using the nodes computed from Eqn. (2), the distance between the outermost nodes in previous two regions were determined. The distance values were divided according to the required number of elements across region 2. Eqn. 1 and algorithm A1were adopted to generate the nodes using the distance values while applying the bias seeding on the first line in region 2. The line was later swept from angle 0 to π in a similar manner as region 1. Similar node generation as shown in region 1 were repeated using algorithm A2 to complete the whole region 2. The only input parameters required for region 2 were the number of elements across the first line and its bias ratio. The algorithms A1 and A2 can be modified for generation of nodes inregion 3 and region 4 as required. Lastly, the nodes were numbered starting from 1 following the order of nodes generation. Therefore, the nodes numbering system of the model were carried in a consistent manner throughout the regions.

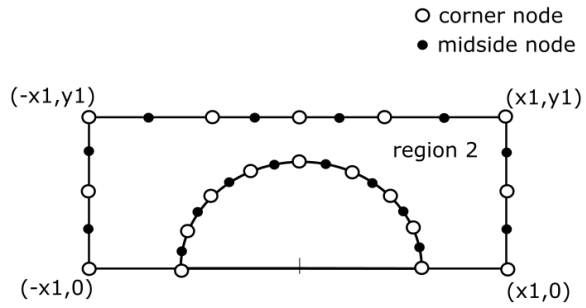

Figure-5. Nodes generated in outermost boundaries.

ELEMENT CONNECTIVITY

The model studied in this work consists of discrete second order quadrilateral elements. The nodes created must be connected in the right order to form a suitable elemental configuration. Each second order quadrilateral element consists of 8 nodes, with 4 midside nodes and 4 corner nodes. In ABAQUS, the nodes should be connected in such way by picking corner nodes with the lowest number for the connectivity definition of 4 corner nodes, followed similarly by the 4 midside nodes. The pattern in selecting the nodes for element connectivity must be consistent throughout the model. It can be in either clockwise or counter clockwise manner. The first nodes for the midside nodes connectivity definition must lies at the side of the first nodes for corner nodes connectivity definition. As shown in Figure-6, by taking element 1, the element connectivity for element 1 in this work are defined in counter-clockwise manner as (18,20,34,32,19,26,33,25).

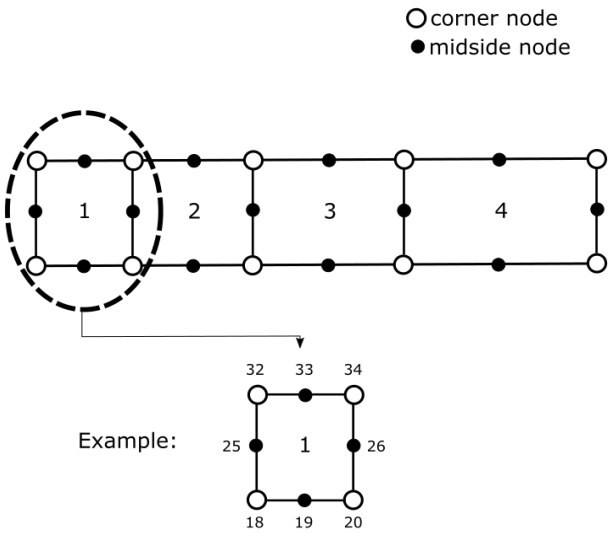

Figure-6. Elements connectivity diagram.

Algorithm A4 as shown below was used to connect nodes into 8-noded quadrilateral element in region 1. In this work, the nodes generated in other regions of the model were connected similarlyas in region 1.

```
%Algorithm A4
total_ele=No_ele*E_w;
element=zeros(total_ele,8);

for n from 1 to E_w
fori from 1 to No_ele
      e1=(n-1)*(No_ele)+i;
      m1=No_ele+(2*No_ele+1)+3;
      m2=2*No_ele+1;
      element(e1,1)=(n-1)*((No_ele +1)
            +(2*No_ele +1))+(i*2-1);
      element(e1,2)= element(e1,1)+2;
      element(e1,3)=(n-1)*((No_ele+1)
            +(2*No_ele+1))+(i*2-1)+ m1;
      element(e1,4)= element(e1,3)-2;
      element(e1,5)= element(e1,1)+1;
      element(e1,6)=(n-1)*(No_ele+1)
            +(m2*n)+i+1;
      element(e1,7)= element(e1,3)-1;
      element(e1,8)= element(e1,6)-1;
end
end
```

RESULTS

By compiling all the algorithms as described above in MATLAB, an input file for ABAQUS containing the nodes and elements information is created. An example of the designed models is depicted in Figure-7. The meshes on model can be refined by varying the number of elements across the regions and the associated bias ratio.

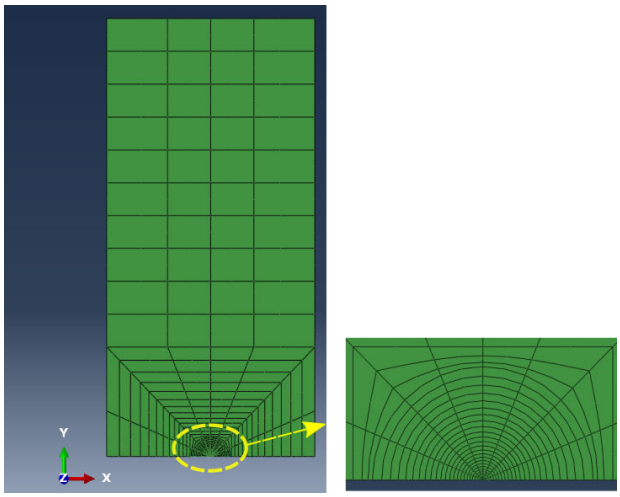

Figure-7. Full field model created using algorithms.

CONCLUSIONS

The algorithm discussed in this paper simplified the steps needed to prepare a finite element model for fracture mechanics computation. The nodes and elements were generated in a consistent manner, which was convenient for the numbering system and post processing process. However, the discussed algorithm has its own limitation where it can only be applied on single edge cracked with a/W=0.5. The development of the algorithm for specimen with different a/W ratio is currently in progress.

ACKNOWLEDGEMENTS

The present study is carried out through a MOHE grant (FRGS/1/2014/TK01/USM/02/5). The ABAQUS finite element code was made available under an academic license from Dassault Systems, K.K. Japan and thanks are due for access toa high performance computing cluster available in the School of Mechanical Engineering, Universiti Sains Malaysia.

REFERENCES

[1] Irwin, G. 1957. Relation of stresses near a crack to the crack extension force. 9th Cong. App. Mech., Brussels.

[2] Rice, J.R. 1968. A Path Independent Integral and the Approximate Analysis of Strain Concentration by Notches and Cracks. Journal of Applied Mechanics. 35(2): 379-386.

[3] Rice, J.R. 1968. Mathematical analysis in the mechanics of fracture. Fracture: an advanced treatise. 2: 191-311.

[4] Larsson†, S.G. and A.J. Carlsson. 1973. Influence of non-singular stress terms and specimen geometry on small-scale yielding at crack tips in elastic-plastic materials. Journal of the Mechanics and Physics of Solids. 21(4): 263-277.

[5] Williams, M. 1956. On the stress distribution at the base of a stationary crack. Journal of Applied Mechanics. 24(1): 109-114.

[6] Betegón, C. and J.W. Hancock. 1991. Two-Parameter Characterization of Elastic-Plastic Crack-Tip Fields. Journal of Applied Mechanics. 58(1): 104-110.

[7] Du, Z.Z. and J.W. Hancock. 1991. The effect of non-singular stresses on crack-tip constraint. Journal of the Mechanics and Physics of Solids. 39(4): 555-567.

[8] Sharma, S.M. and N. Aravas. 1991. Determination of higher-order terms in asymptotic elastoplastic crack tip solutions. Journal of the Mechanics and Physics of Solids. 39(8): 1043-1072.

[9] O'Dowd, N.P. and C.F. Shih. 1991. Family of crack-tip fields characterized by a triaxiality parameter-I. Structure of fields. Journal of the Mechanics and Physics of Solids. 39(8): 989-1015.

[10] Kim, Y., X.K. Zhu, and Y.J. Chao. 2001. Quantification of constraint on elastic-plastic 3D crack front by the J–A2 three-term solution. Engineering Fracture Mechanics. 68(7): 895-914.

[11] Guo, W.L., *et al*. 2006. Advances in three-dimensional fracture mechanics. in Key Engineering Materials. Trans Tech Publ.

[12] Guo, W. 1993. Elastoplastic three dimensional crack border field-I. Singular structure of the field. Engineering Fracture Mechanics. 46(1): 93-104.

[13] Guo, W. 1993. Elastoplastic three dimensional crack border field-II. Asymptotic solution for the field. Engineering Fracture Mechanics. 46(1): 105-113.

[14] González-Albuixech, V.F., *et al*. 2011. Influence of the t33-stress on the 3-D stress state around corner cracks in an elastic plate. Engineering Fracture Mechanics. 78(2): 412-427.

[15] Anderson, T.L. and T. Anderson. 2005. Fracture mechanics: fundamentals and applications. CRC press.