دانشگاه شاهد
Shahed University
Faculty of Engineering
Department of Civil Engineering

The Combination extended finite element method and adaptive finite element method in crack analysis with MATLAB code

User's Guide

By:

Kamran Modabber

Table of Contents

Introduction

Extended finite element method is one of the best methods in crack analysis where it can solve crack problems without creating new model when crack growth and we can solve the crack problems easier than solving with classical finite element method. Another method for analyzing crack problems is adaptive finite element method where it can solve crack problems base on mesh refinement. In this method with estimating errors, we can offer a new mesh that it adaptive on problem and it can deduct errors so we can rely on the results.

In this case we offer a new technique that it can solve and analyze crack problems with both of extended and adaptive finite element method. So we present a new code in MATLAB base on this technique.

Extended finite element method

Here, a simple two-dimensional plane stress and plane strain XFEM implementation within MATLAB is presented. With XFEM MATLAB code we can identify the crack path. (figure 1)

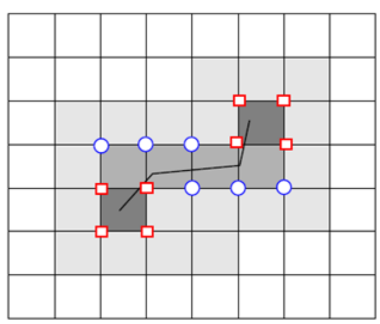

Figure 1: A typical FE mesh with an arbitrary crack.

For modeling the strong discontinuity of cracked body two enrichment functions are used. The Heaviside step function represents the discontinuity away from the crack tip, and the linear elastic asymptotic crack tip displacement fields are used to account for discontinuity at the crack tip. (Equation 1)

$$U = \sum_{i \in N} N_i(x) U_i + \sum_{i \in N_{cut}} N_i(x) H(x) \alpha_i + \sum_{i \in N_{front}} \sum_{\alpha} N_i(x) B_\alpha(x) b_{i,\alpha} \qquad (1)$$

Adaptive finite element method

Adaptive finite element methods are a fundamental numerical appliance in engineering try to automatically refine, coarsen or relocate a mesh and/or adjust the basis to achieve a solution having a specified accuracy in an optimal fashion. This method is useful in order to simulate crack problems. In the near crack tips, intensity of stress is very mush so in this part of cracks we have errors a lot. In order to overcome this condition and decrease the errors we use remeshing. In this method we estimate errors and base on errors refine meshes. By this technique, we can close our results to analytical results.

In order to estimate errors in this paper we use difference between improved results and first results. So we can use this equation to find improved results:

$$\sigma^* = N\bar{\sigma}^* \qquad (2)$$

Since the improved results are better approximation to the real results so we can use the difference between the two results as errors according to the following equation:

$$e_\sigma \approx \sigma^* - \hat{\sigma} \qquad (3)$$

In this way we can calculate errors in the entire domain we this equation:

$$\|e_\sigma\| = \|\sigma^* - \hat{\sigma}\| = \left(\int_\Omega (\sigma^* - \hat{\sigma})^T (\sigma^* - \hat{\sigma}) d\Omega \right)^{1/2} \qquad (4)$$

Finally if error be more than aim error, so we need new meshes to continue crack analysis and then our MATLAB code create a new density base on following equation:

$$h_{i_{new}} = \left[\frac{\|e_\sigma\|_{i_{aim}}}{\|e_\sigma\|_i} \right] h_{i_{old}} \qquad (5)$$

Crack growth

The direction of the crack extension plays an important role in the numerical simulation. To predict the direction of the crack path we must calculate stress intensity factor. The stress intensity factor, K, is used in fracture mechanics to predict the stress state near the tip of a crack caused by a remote load or residual stresses. To calculate stress intensity factor there are many techniques. One of those techniques is the interaction integral method.

In the interaction integral method, auxiliary fields are introduced and superimposed onto the actual fields satisfying the boundary value problem. Stresses and strains for the auxiliary state should be chosen so as to satisfy both the equilibrium equation and the traction free boundary condition on the crack surface in the A* area. These auxiliary fields are suitably selected in order to find a relationship between the mixed mode stress intensity factors and the interaction integrals. The contour J for the sum of the two states can be defined as:

$$J = J^{act} + J^{aux} + M$$

Where J act and J aux are associated with the actual and auxiliary states, respectively, and M is the interaction integral:

$$M = \int_{A^*} [\sigma_{ij} \frac{\partial u_i^{aux}}{\partial x_1} + \sigma_{ij}^{aux} \frac{\partial u_i}{\partial x_1} - W^M \delta_{1j}] \frac{\partial q}{\partial x_j} d\Gamma$$

$$M = \frac{2}{E'}(K_I K_I^{aux} + K_{II} K_{II}^{aux})$$

$$K = \frac{E'}{2} M$$

$$\left\{ \begin{array}{lll} for \ \mathrm{mod}\, I & K_I^{aux} = 1 \quad , & K_{II}^{aux} = 0 \\ for \ \mathrm{mod}\, II & K_I^{aux} = 0 \quad , & K_{II}^{aux} = 1 \end{array} \right\}$$

MATLAB code

In order to use this code we present this section and we introduce the all sectors in this code.

First we need to determining the mechanical properties, for this purpose we create XLS file where user can determinate the mechanical properties. XLS file name is “moshakhasat” and in this file there are many cells to determining the mechanical properties as following:

E: Young's modulus

nu: Poisson's ratio

plane stress/strain(1/2): for plane stress is 1 and for plane strain 2

x_ebteda_tarak: The coordinates of the beginning of the crack (x)

y_ebteda_tarak: The coordinates of the beginning of the crack (y)

x_enteha_tarak: Coordinate the end of the crack(x)

y_enteha_tarak: Coordinate the end of the crack(y)

sigma: Uniform tension in edge of the plate

noe_tarak: number of tips in crack

tedad_jahat_ niro: if we want apply uniform tension in one edge, we put 1 and if we want apply uniform tension in two edge, we put 2.

After, we must determinate geometrical parameter. For this purpose we use mesh generator GID to create initial meshes. In this time we create txt files in order to introduce them to MATLAB code. The txt input files are as following:

vorodi1: first Column is number of nodes, second and third Column is coordinate of nodes.

Vorodi2: first Column is number of element, and other columns are connection between nodes in each element.

Vorodi3: the nodes on the first edge that faced with uniform tensions.

Vorodi4: nodes for boundary conditions.

Vorodi5: the nodes on the second edge that faced with uniform tensions. (If there are)

Then, we can run the code and this code solves and analyzes the problem and show many information and result about the problem. During the run, the program ask aim error and if error be more than aim error, the program give us a new wt file in order to use in mesh generator GID and then GID create new meshes base on this wt file where this wt file is new density of nodes. The wt file name is gid.

After that we continue this cycle in order to predict crack path. at the end of each step we can see results and tecplot files that show cantor of stresses. The tecplot file name is xfem.

Example

To express the efficiency of the proposed method, four numerical examples are investigated. In both examples, the model with an arbitrary meshing by using extended finite element method has been solved. Then, based on the extracted errors, new density is calculated and mesh modification process is completed.

For the analysis, normal quadratic element is used and we consider aim error 15%. If the error is greater than aim error, we will refine meshes. The first example is edge crack and the second example is Slanted crack with 45 degree.

For first example, consider the two-dimensional rectangular sheet that it faced tensile stress from up and down. (See figure). The length of sheet is b=25 Cm, height is 2h=100 Cm and initial crack length is a=8 Cm. in this problem we assume σ=1 Kg/Cm2 .the other mechanical parameters such as modulus of elasticity and Poisson's ratio are respectively $E = 1000 \, {}^{Kg}\!/_{Cm^2}$ and $\nu = 0.3$.

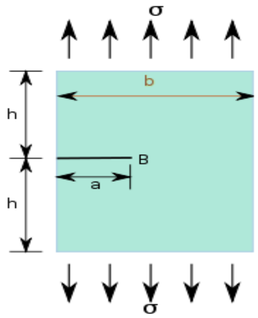

figure 2: edge crack

After running MATLAB code, we can have many results and information about this problem. For example the crack path growth is as following figure:

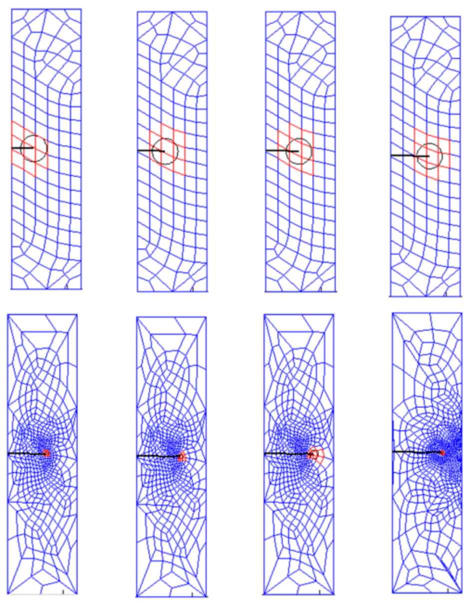

figure 3: crack path growth