Numerical Modeling of Crack Growth Using the Level Set Fast Marching Method

Gordana Jovičić
Teaching and Research Assistant

Miroslav Živković
Associate Professor

Nebojša Jovičić
Associate Professor

University of Kragujevac
Faculty of Mechanical Engineering

The Fast Marching method is a numerical technique for modeling arbitrary cracks, holes and material interfaces (inclusions) without meshing the internal boundaries. This technique is computationally attractivealternat for problems of strictly monotonically advancing fronts. The evolving interface is represented as a level contour of a function of one dimension higher. This technique is based on finite difference methods for hyperbolic conservation laws enabling the accurate and stable evolution of sharp corners and cusps in interface.
The Fast Marching method assures the fully automated crack growth simulation. In this study the methodology, algorithm and implementation of the planar crack growth of the single crack is presented

Keywords: *Fast Marching method, Hamilton-Jacobi equation, crack growth*

1. INTRODUCTION

The Level Set Method is a numerical scheme tailored, among others, to model arbitrary cracks, holes and material interfaces (inclusions), without meshing the internal boundaries. Osher and Sethian (1988) introduced the Level Set Method (LS method) to represent interface as the zero level surface of a function of one dimension higher. The LS method is established on Initial value formulation. This technique is based on finite difference method for hyperbolic conservation laws enabling the accurate and stable evolution of sharp corners and cusps in interface.

In the context of the LS method, the Fast Marching Method (FM method) was introduced first by Sethian [1], and later was improved through papers of Sethian [2] and Chopp [3]. The FM method is established on Boundary value formulation. The FM method employs no time step, and hence is not subject to time step restriction (CFL conditions), unlike LS methods.

These techniques have been used in a large variety of applications, including problems in: fluid interface motion, two phase flow simulation [3], combustion, dendrite solidification [4], etching and deposition semi-conductor manufacturing [1], [5], robotic navigation and path planning [5], computation of seismic travel times [6], image segmentation in medical imaging scans.

These techniques require an adaptive methodology to obtain computational efficiency. In the case of LS and FM methods this leads to an adaptive methodology, the Narrow Band Method [7]. The narrow band is a region surrounding the interface, and the FM computation needs only to be done in that region. In the case of the crack growth, the level set function is then updated only in this region called the narrow band.

Using these techniques, simulation of planar crack growth is carried out in this study.

Received: May 2004, Accepted: December 2004.

Correspondence to: Gordana Jovičić
Faculty of Mechanical Engineering, University of Kragujevac
Sestre Janjića 6, Kragujevac 34000, Serbia and Montenegro
E-mail: gjovicic@ept.kg.ac.yu

2. EVOLVING THE INTERFACE WITH LEVEL CONTOUR

In view of interface propagation the LS method approximates the solution of an initial value partial differential equation, while the FM method approximates the solution of a boundary value partial differential equation. Both techniques rely on viscosity solutions of Hamilton-Jacobi equation, using upwind schemes for hyperbolic conservation laws. In both methods, the evolving interface is represented as a level contour of a function of one higher dimension.

In this paper we present the methodology for solving a boundary value partial differential equation for the problems of the crack growth, where the front speed F depends only on position.

2.1. An initial and a boundary value formulation

We suppose that the initial position of the front is the zero level set of a higher dimension function ψ. We can then identify the evolution of this function ψ with propagation of the front itself through a time dependent initial value problem. At any time, the front is given by the zero level set of the ψ [8] in all points in the computational domain.

In order to derive an equation of the motion for this LS-function ψ, we note that the zero level set always matches the propagating hyper-surface. It means that:

$$\psi(\mathbf{x}(t),t) = 0\,. \tag{1}$$

In order to derive the partial differential equation for the time evaluation of ψ, one can use the chain rule:

$$\psi_t + \nabla\psi(\mathbf{x}(t),t)\,\mathbf{x}'(t) = 0\,. \tag{2}$$

Then it follows:

$$\mathbf{x}'(t)\cdot\mathbf{n} = F\,. \tag{3}$$

In eqn. (3) F is a speed in the outward normal direction, where the outward normal direction is obtained from the level set function ψ, namely:

$$\mathbf{n} = \frac{\nabla\psi}{\|\psi\|}\,. \tag{4}$$

The equation (2) then becomes:

$$\psi_t + F\|\nabla\psi\| = 0\,, \tag{5a}$$

$$\psi(\mathbf{x}, t=0) = \pm\mathrm{d}(\mathbf{x})\,, \tag{5b}$$

where the right hand side of the second equation is given. The term $\pm\mathrm{d}(\mathbf{x})$ is the signed distance from **x** to initial front. Eqn. (5b) is the level set equation introduced by Osher and Sethian [6], [8]. This formulation can be applied for the arbitrary speed function F. As it is analyzed by Sethian [8], the efficient solution of these front propagation problems requires the use of upwind difference schemes borrowed from the solution of hyperbolic conservation laws.

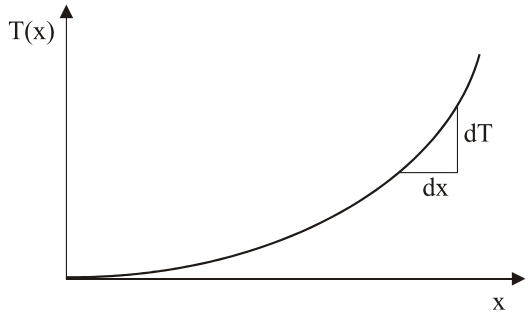


Figure 1. Setup for the Boundary Value formulation

We suppose that the initial position of the front is the zero level set of a higher dimension function ψ. We can then identify the evolution function ψ with propagation of the front itself through a boundary value problem. At any time, the front is given by the level set of the time ψ [8] in all points in the computational domain. One way to characterize the position of this expanding front is to compute the arrival time $T(x,y)$ of the front as the crosses of each point (x,y) [9]. The equation that describes this arrival surface $T(x,y)$) is derived using: *distance=rate*time*, ($\Delta x = F\Delta t$), see Fig. 1 and:

$$1 = F\frac{\mathrm{d}T}{\mathrm{d}x}\,. \tag{6}$$

In multi dimension, the spatial derivative of the solution surface T becomes the gradient, and hence we have:

$$\|\nabla T\| = \frac{1}{F}\,. \tag{7}$$

Equation (7) represents the boundary value partial differential equation describing the interface motion [9]. By solving the eqn. (7) one can obtain the crossing time mapping, the front arrival time that corresponds to each point **x**. If the speed F depends only on position, then the equation reduces to the familiar Eikonal equation.

If ψ is a signed distance function so that:

$$\|\nabla\psi\| = 1 \tag{8}$$

is one of the eqn. (7) solutions. The function ψ remains the signed distance function for all time in all regions where ψ and F are both smooth [10], [11] if:

$$\nabla F\cdot\nabla\psi = 0\,. \tag{9}$$

Equation (9) assures that function ψ remains the signed distance function that satisfies eqn. (7) for all the time. Solution of eqn. (9) gives the extension velocity F. Also, with (9) it is assured that $F = \text{const}$ along the normal direction on the front.

For the boundary value problem, with relations (8) and (9), the front motion is characterized as a solution of the eqn. (7) [12].

The boundary value perspective is restricted to the front that always moves in the same direction, i.e. outward, because it requires crossing time ($T=\psi$) at each grid point, and hence a point cannot be revisited.

For certain speed F, eqn. (5) or (7) reduces to some familiar equations. For example, for F=1, the equation becomes the Eikonal equation for a front moving with the constant speed. For $F = 1-\varepsilon\cdot\kappa$, where κ is the curvature of the front, eqn. (5) becomes a Hamilton-Jacoby equation with parabolic right-hand-side, as it is discussed in [10]. For $F = \kappa$, eqn. (5) reduces to the equation for the mean curvature flow. The curvature κ may be determined from level set function ψ, from divergence of the unit normal vector to the front, i.e.

$$\kappa = \frac{\psi_{xx}\ \psi_y^2\ -2\ \psi_x\psi_y\ \psi_{xy}\ +\psi_{yy}\ \psi_x^2}{\left(\psi_x^2\ +\ \psi_y^2\right)^{3/2}}\,. \tag{10}$$

Therefore, depending on option of the speed F, eqn. (5) or (7) has been used in a large variety of applications, including the problems in: fluid interface motion, combustion, dendritic solidification, etching and deposition semi-conductor manufacturing, robotic navigation and path planning, computation of seismic travel times, image segmentation in medical imaging scans. In this paper, we have made assumption that the front velocity F depends only on the front position as it is referred to in [13], [14].

Solving equations (7) and (8) can be done by using the FM method which is the optimal technique for solving Eikonal equation, coupled with a bicubic interpolation scheme for initialization.

3. THE ZERO LEVEL SET FOR LOCATING THE CRACK FRONT

The zero value level set function represents the location of the interface [15]. It is necessary for initialization of the FM method that will be described below.

We consider an elliptical planar crack with semi-major (a) and semi-minor lengths (b), with axes oriented along the local $\hat{x}_1$, $\hat{x}_2$ axes, respectively. Let $\hat{\mathbf{e}} = \mathbf{R} \cdot \mathbf{e}$ be the mapping between the global and the local unit base vectors. Now, the coordinates of a point in the local orthogonal coordinate system, with the origin at the center of the ellipse, are given by: $\hat{\mathbf{x}} = \mathbf{R} \cdot (\mathbf{x} - \mathbf{x}_c)$, where $\mathbf{x}_c$ is the center of the ellipse.

The level set function for elliptical planar crack front is [16], [17]:

$$\psi(\mathbf{x}, 0) = f(\xi), \tag{11}$$

where:

$$f(\xi) = \left\| \xi_1^2 + \xi_2^2 \right\| - 1, \qquad \xi = \left(\frac{\hat{x}_1}{a}, \frac{\hat{x}_2}{b} \right). \tag{12}$$

The zero level set function (11) represents elliptical planar crack front. Note that eqn. (11) is a level set function but not a signed distance function.

Now, consider a polygonal planar crack front, which consists of p segments:

$$I_1 = [\mathbf{x}_1, \mathbf{x}_2],\ I_2 = [\mathbf{x}_2, \mathbf{x}_3], ..., I_p = [\mathbf{x}_p, \mathbf{x}_1].$$

The level set function for polygonal interface is given by:

$$\psi(\mathbf{x}, 0) = \left\| \mathbf{x} - \mathbf{x}_{\min} \right\| \operatorname{sgn}((\mathbf{x} - \mathbf{x}_{\min}) \cdot \mathbf{n}_{\min}), \tag{13}$$

where:

$$\operatorname{sgn}(\varsigma) = \begin{cases} 1 \ if \ \varsigma \geq 0, \\ -1 \ if \ \varsigma \geq 0, \end{cases}$$

$$\left\| \mathbf{x} - \mathbf{x}_{\min} \right\| = \min_{\substack{\mathbf{x}_i \in I_i \\ i=1,2,..,p}} \left\| \mathbf{x} - \mathbf{x}_i \right\|.$$

In eqn. (13) $\mathbf{x}_{\min}$ is the orthogonal projection of $\mathbf{x}$ on the interface and $\mathbf{n}_{\min}$ is outward normal to the interface at $\mathbf{x}_{\min}$. If no unique normal is defined at $\mathbf{x}_{\min}$, the sign is positive if $\mathbf{x} - \mathbf{x}_{\min}$ belongs to the cone of normal at $\mathbf{x}_{\min}$ and negative otherwise.

Level set function for a polygonal planar front and an elliptical planar front are shown in the Fig. 2. The zero value of the level set functions (11), i.e. (11-12), represents crack front in the finite element mesh.

The level set functions (11) or (13a) are used to obtain the one-dimensional segment of the crack (interface, void, etc.), which intersect element e (Fig. 3). The edges of the element e are intersected with the crack (interface, void, etc.) when LS-function for the nodes I and J satisfy the condition $\psi_I^t \psi_J^t < 0$. Then the intersection point $\mathbf{x}_p$ is determined by:

$$\mathbf{x}_p = \mathbf{x}_I^t + \xi(\mathbf{x}_J^t - \mathbf{x}_I^t), \qquad \xi = -\frac{\psi_I^t}{\psi_J^t - \psi_I^t}. \tag{14}$$

The intersection points of the crack segment are found using the above relation. The above approach is accurate and feasible in two dimensions, since discretization of the crack into linear segment is readily performed.

The values of the level set functions are stored only at nodes. The functions are interpolated over the mesh by the same shape functions as the displacement in FEM:

$$\psi(\mathbf{x}, t) = \sum_{j \in J} \psi_j(t) N_j(\mathbf{x}), \tag{15}$$

where ψ_j is the value LS-function at the j^{th} node and N_j is the shape function at the j^{th} node.

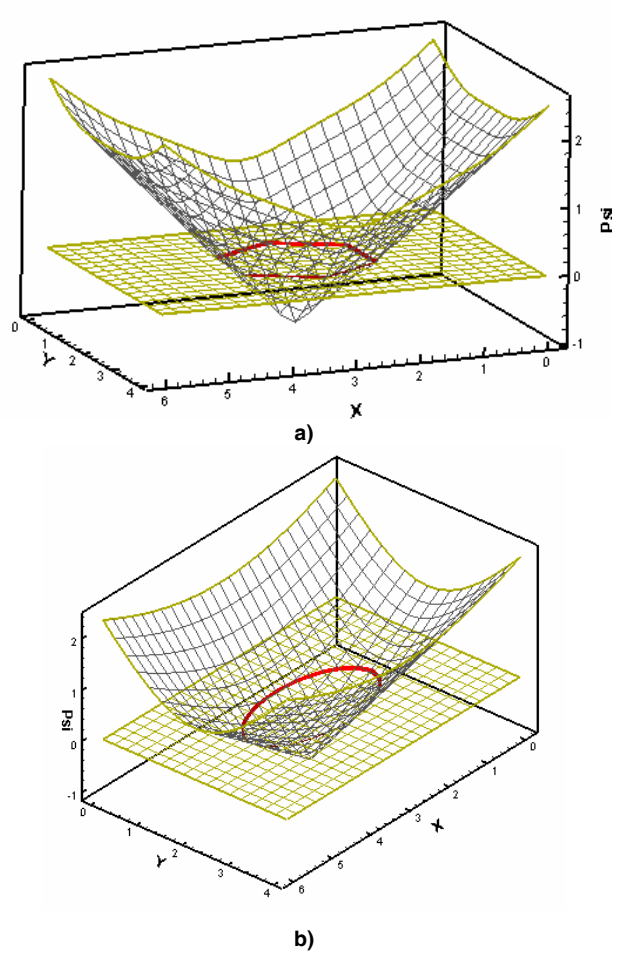


Figure 2. The distance value from 0 to 3 of LS-functions for a) polygonal and b) elliptical planar front

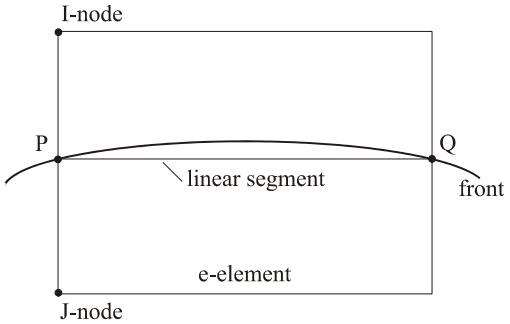


Figure 3. The linear segment of the crack on the elemente

4. THE FAST MARCHING METHOD

The functions for a polygonal planar front (13) and an elliptical planar front (11-12) are level set functions but are not signed distance functions. In order to obtain

the signed distance function φ for arbitrary planar front, one applies the FM method, which optimally solves an Eiconal equation of the form (7).

4.1. Upwind construction for boundary value problem

Consider a two-dimensional version of Eikonal equation, where the boundary value is known. We analyze a numerical scheme to approximate the eqns. (7-9). It is known that central difference approximation to the gradient produces the wrong weak solution, so we exploit the technology of hyperbolic conservation laws in devising schemes that maintain sharp corners in the evolving hyper-surface and choose the correct entropy-satisfying weak solution [14]. Let ψ_{ij}^n be the numerical approximation to the solution ψ at the point $i\Delta x, j\Delta y$ and at time $n\Delta t$, where Δx and Δy is the grid spacing and Δt is the time step. One of the simplest schemes for correct weak solution of eqn. (5) is given in [1], namely:

$$\nabla\psi_{ij}\Big| \approx \left[\begin{array}{l}\left(\min(D_x^-\psi_{ij},0)\right)^2+\left(\max(D_x^+\psi_{ij},0)\right)^2+\\ \left(\min(D_y^-\psi_{ij},0)\right)^2+\left(\max(D_y^+\psi_{ij},0)\right)^2\end{array}\right]^{1/2} = f_{ij}\cdot \quad (16)$$

Here, it refers to the backward difference in the x-direction:

$$D_x^-\psi_{i,j} = \frac{\psi_{i,j}-\psi_{i-1,j}}{\Delta x}. \quad (17)$$

The other difference operators are defined similarly.

Eqn. (16) is an upwind scheme. Choosing grid points in the approximation, depends on the direction of the flow information [14]. Upwind means that if a wave progresses from left to right, then one should use a difference scheme which reaches upwind to the left in order to get information to construct the solution downwind to the right [6].

4.2. The narrov band perspective

Considerable computational speedup in the FM method comes from the use of the Narrow Band LS method, introduced by Adalsteinsson and Sethian [7]. The key is that the upwind difference structure of eqn. (16) propagates information in "one way" that is from smaller values of ψ to larger values, because the time ($T=\psi$, which is solution of eqn. (16) for $f_{ij}=1/F_{ij}$) is monotonically increasing value. The FM method builds the solution from eqn (16) outwards from the smallest ψ value. We may start the algorithm by marching "downwind" from the known value, computing new values at each neighboring grid points. We can step the solution outwards from the boundary condition in a downwind direction. The algorithm is again made fast by confining the "building zone" to narrow band around the front. The key is in the selection of which grid point to update in the narrow band. All points in the computational domain, called narrow band, we divide in to three zones: zone with the accepted values, zone with the trial, and zone with the faraway values (see Fig. 4)

The close set nodes are set nodes of elements which are cut with front. The points with specified initial condition are first tagged as accepted set points. Then, we tag all neighbors of accepted grid points as trial set points. All points that do not belong to accepted and trial set points will be considered as a faraway set points. We can march the solution (from eqn. (16)) outwards always selecting the narrow band grid point with minimal trial value for T, and freezing the node with minimal trial value and remove this node from Trial set to Accepted set. Next step is bringing all freezing node neighbors from a faraway set to a trial set. Next step is re-computing values of T at all nodes from the trial set.

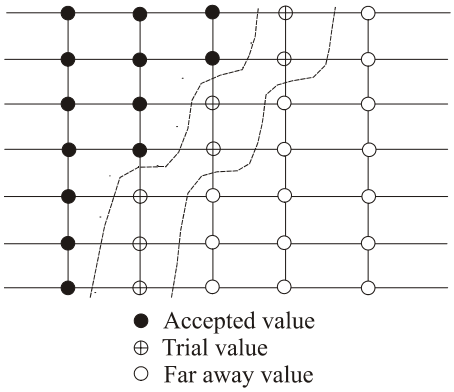


Figure 4. Upwind construction of computational domain

This algorithm works properly because the process of re-computing the T values at trial set nodes cannot yield a value smaller than any of the accepted set points.

4.3 Initial condition

In order to apply the FM method, the velocity field F itself must be defined on the entire domain of ψ, not just the zero level set corresponding to the interface itself. Thus, it is necessary to extend F from the interface into the domain of ψ. In [9], a technique was introduced for building this extension velocity field in a highly efficient and accurate manner. Construction of extension velocities need only be done to points lying in the narrow band, as opposed to all points in the computational domain. This can greatly reduce the computational time.

In order to build extension velocity, it is necessary to construct signed distance function ϕ. We can use the FM method to compute the signed distance function ϕ^{temp} by solving Eikonal equation:

$$\|\nabla\phi\| = 1. \quad (18)$$

The solution (18) will be our temporary signed distance function ϕ^{temp}. The FM method is run separately for grid points outside (LS-function is $\psi>0$) and inside (LS-function is $\psi<0$) the front.

The initialization stage of the FM method is computation of the approximate distances and extension

velocities of the close set nodes. The close nodes are a set of nodes of the elements that are intersected with the front, i.e. with zero level set function ψ. We must assign values at these points to approximate the distances to the front.

In close grid points it is necessary to obtain values from extension velocity in order to begin the FM method. Our aim is to build an extension velocity F_{ext} such that if $\|\nabla\phi\| = 1$ then updating under this extension velocity maintains this unit gradient. This extension should extend the speed in a continuous manner close to the front. We want to construct a speed that satisfies the equation:

$$\nabla F_{\text{ext}} \cdot \nabla \phi^{\text{temp}} = 0 . \tag{19}$$

The eqn. (19) is satisfied for all grid points, except the points along the front itself.

For building extension velocities at grid nodes near the front, we take a weighted average of speed values at the points that are used in computing the signed distance. The weight is proportional to one over the square of the distance; this is equivalent to solving the equation (19) [9].

One goal is to simultaneously construct the signed distance using (18) and extension velocity using (19) at the same close node. The above expression assumes that the speed of the interface is given at the intersection points of the front with the grid lines. There are five possible cases that need to be considered, and are shown in Fig. 5.

In Fig. 5.a, only one of the neighboring points from grid point (i, j) is on the other side of the front. The distance d from the point (i, j) to the intersection front point is:

$$d = s , \tag{20}$$

where the meaning of s is shown in Fig. 5.a. The sign of the distance d depend on the sign of LS function ψ. That distance is positive if $\psi_{ij} > 0$ and distance is negative otherwise.

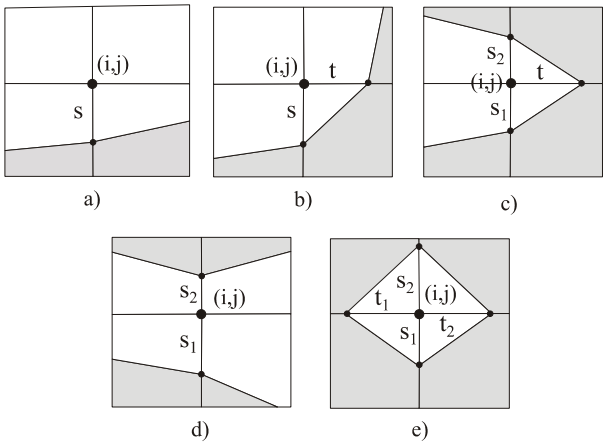


Figure 5. The initial conditions

In this case extension velocity at grid point (i, j) would be copying the speed of the closest front point

$$f(i, j) = f(i, j - s) . \tag{21}$$

The value of the speed function at the closest point on the front $(i, j - s)$ is used as the extension velocity at the grid point (i, j).

In Fig. 5.b two of the neighbors are on the other side of the front. In this case the value is defined as the exact distance to the line segment between the intersection points. The exact distance d of point (i, j) satisfies:

$$\left(\frac{d}{s}\right)^2 + \left(\frac{d}{t}\right)^2 = 1 . \tag{22}$$

The left-hand side is an upwind approximation to the gradient of the distance function, since distance is zero at the intersection points.

In Fig. 5.b the extension speed is given by:

$$f(i, j) = \frac{\frac{1}{t^2} f(i + t, j) + \frac{1}{s^2} f(i, j - s)}{\frac{1}{t^2} + \frac{1}{s^2}} . \tag{23}$$

In Fig. 5.c the distance d is the positive solution of

$$\left(\frac{d}{\min(s_1, s_2)}\right)^2 + \left(\frac{d}{t}\right)^2 = 1 . \tag{24}$$

The extension velocity for the case shown in Fig. 5.c is:

$$f(i, j) = \frac{\frac{1}{t^2} f(i + t, j) + \frac{1}{s^2} f(i, j + s)}{\frac{1}{t^2} + \frac{1}{s^2}} , \tag{25}$$

where $s = s_1$ if $|s_1| < |s_2|$, $s = s_2$ otherwise.

In Fig. 5.d, the distance d is:

$$d = \min(s_1, s_2) . \tag{26}$$

In this case the extension velocity at grid point (i, j) is:

$$f(i, j) = f(i, j + s) , \tag{27}$$

where s is chosen as in the term before.

In Fig. 5.e, the distance d is positive solution of equation:

$$\left(\frac{d}{\min(s_1, s_2)}\right)^2 + \left(\frac{d}{\min(t_1, t_2)}\right)^2 = 1 . \tag{28}$$

The extension velocity for this case is:

$$f(i, j) = \frac{\frac{1}{t^2} f(i + t, j) + \frac{1}{s^2} f(i, j + s)}{\frac{1}{t^2} + \frac{1}{s^2}} , \tag{29}$$

where s is chosen as in eqn. (25), while t is chosen as $t = t_1$, if $|t_1| < |t_2|$ and $t = t_2$ otherwise.

With the above equations one has shown the approximate construction of signed distance and extension velocity at close grid points from the front. It is necessary for starting the FM method. The distance

value at trial set points is updated using the FM method on eqn. (18) and the extension velocity at trial set is chosen such that $\nabla F_{\text{ext}} \cdot \nabla \phi^{\text{temp}} = 0$, where the gradients of F_{ext} and ϕ^{temp} are calculated using the same points that are used to calculate distance value.

Let $(i+1, j)$ and $(i, j-1)$ be the points used in updating the distance at point (i, j), if F_{ij} is unknown extension velocity at this point, it then has to satisfy upwind version of eqn. (19):

$$\left(\frac{\phi_{i+1,j}^{\text{temp}} - \phi_{i,j}^{\text{temp}}}{h}, \frac{\phi_{i,j}^{\text{temp}} - \phi_{i,j-1}^{\text{temp}}}{h} \right) \cdot \left(\frac{F_{i+1,j} - F_{i,j}}{h}, \frac{F_{i,j} - F_{i,j-1}}{h} \right) = 0 . \quad (30)$$

Since nodes $(i+1, j)$ and $(i, j-1)$ have been accepted, the above equation can be solved with respect to F_{ij}:

$$F_{i,j} = \frac{F_{i+1,j}(\phi_{i,j}^{\text{temp}} - \phi_{i+1,j}^{\text{temp}}) + F_{i,j-1}(\phi_{i,j}^{\text{temp}} - \phi_{i,j-1}^{\text{temp}})}{(\phi_{i,j}^{\text{temp}} - \phi_{i+1,j}^{\text{temp}}) + (\phi_{i,j}^{\text{temp}} - \phi_{i,j-1}^{\text{temp}})} . \quad (31)$$

The speed at any point on the front can be precisely calculated. This speed is then extended onto the surrounding grid points. This leads to a more accurate detection of the front location.

5. CRACK GROWTH ALGORITHM

The crack is assumed to grow in the direction normal to the crack front and in the plane of the crack. Fatigue crack growth is assumed to be governed by Paris law [12]:

$$\frac{\mathrm{d}a}{\mathrm{d}N} = C(\Delta K)^m , \quad (31)$$

where C and m are material constants, N is the number of fatigue cycles, and ΔK is the stress intensity factor range. Mode I of the stress intensity factor (SIF) is the problem considered here, so we have $\Delta K = K_I$. The normal increment Δa is computed at discrete points on the crack front. Let n be the number of points on the crack front at which the SIF is evaluated, and $\Delta a_{\max}$ is the maximum user-specified increment normal to the crack front. Then:

$$\frac{\Delta a_i}{\Delta a_{\max}} = \left(\frac{K_I^i}{K_I^{\max}} \right)^m , \quad (32)$$

which gives the normal growth increment at any point $\mathbf{x}_i$ $(i = 1, 2, ..., n)$ on the front.

The crack growth may be explained algorithmically as follows:

1. Initialization $t < t_{\max}$ **($t_{\max}$ is user-specified)**

a) Define level set function ψ of crack front at all grid points (where ψ =0 is value on the crack front, ψ <0 in the crack interior, ψ >0 in the crack exterior).

b) Define the intersection points crack front $\psi = 0$ with lines of grid.

c) Computing the SIF by using Finite Element method, and increment crack growth according to eqn. (32) at all intersection points elements and front;

d) Close points: Let Close be the set of all grid points at which the approximate values of: signed distance function, extension velocity and time T were determined. The Close nodes are set nodes of elements which are cut with the front, i.e. with zero level set function ψ. Close points will be the Accepted set for the next.

e) Trial set: Let Trial be the set of all grid points which are neighbors of Accepted set nodes.

f) Faraway set: Let Faraway be the set of all the rest of the grid points.

2. Marching forwards

a) Begin Loop:
Compute (re-compute): signed distance function; extension velocity and crossing time map at all grid points from Trial set according to the eqn. (18), (19) and (16) respectively;

b) Let $(i_{\min}, j_{\min})$ be the point in Trial set with the smallest value for arrival time;

c) Add the point $(i_{\min}, j_{\min})$ to Accepted; remove it from Trial set;

d) If any points in Faraway is neighbor to $(i_{\min}, j_{\min})$, remove it from that set and add it to the Trial set.

e) Return to the top of Loop.

3. If $t < t_{\max}$ **then increment** $t(t \leftarrow t+1)$ **and go to step 2.**

6. RESULTS

The results shown here were obtained by using developed numerical algorithm that is presented in previous chapters. In the Fig. 6, a polygonal edge crack, an embedded shape crack and an elliptical edge crack in the bi-unit cube are shown. All cracks are planar crack.

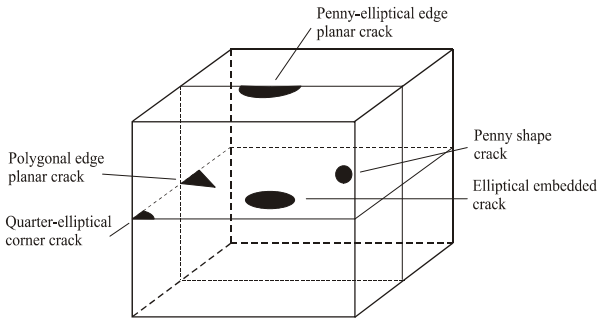


Figure 6. Typical crack shapes

Figure 7 shows arrival time (crossing time mapping) for the penny shape singular planar crack front, which is embedded in a bi-unit cube. For the Fast Marching method we used a narrow band with 12x8 elements in the plane of the planar crack with bilinear interpolation in each grid cell. The start position of the penny shape crack front is obtained by using LS-function of the crack front (11)-(12) where semi-minor and semi-major

lengths are $a \times b \times R$. The intersection points crack front with elements is given by using (14). The time $T = 0$ denotes initial position of the crack front. In Fig. 7 the numerical simulation of the crack front evolution is shown for 10 time step. All intersection points of the crack fronts with elements have equal velocity (SIF is constant along the crack front), because the crack eventually grows to a penny-shape that is in accordance with [15].

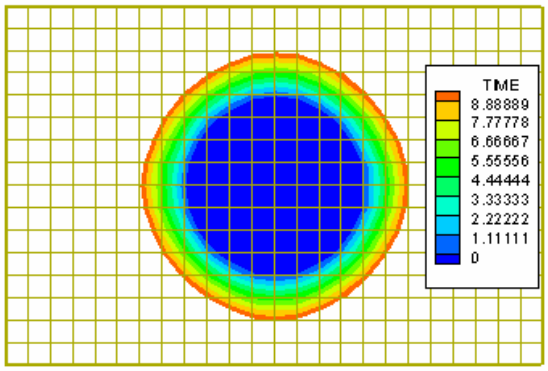


Figure 7. The arrival time of planar penny shaped crack front

We consider an initial elliptical planar edge crack of semi-major and semi-minor axes $a = 1,5$ and $b = 0,5$ respectively, which is embedded in a bi-unit cube. The parameters used are: $\Delta a_{\max} = 0,05a$, $n = 40$. For the FM method, we used a Narrow Band with 48x32 elements in plane of planar edge elliptical crack.

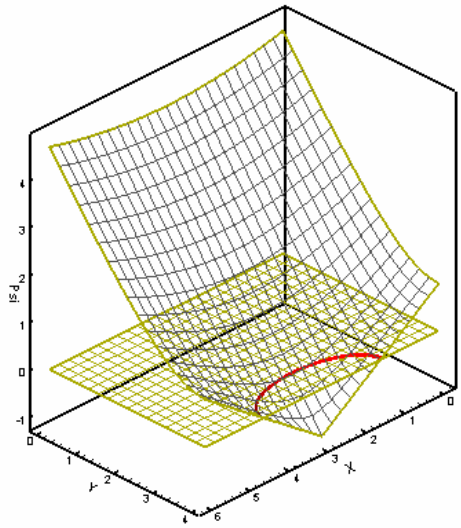


Figure 8. The signed distance level set function of the edge planar semi- elliptical crack

The arrival time is obtained by using eqn. (16). All intersection points of the crack front with the grid elements have velocities that are calculated with growth increments obtained by eqn. (32). In Fig. 9 the numerical simulation of the semi-elliptical edge planar crack front evolution is shown for 300 time steps.

Figure 10 shows enlarged configuration of the above crack front at the initial time $T = 0$ and the time step $T = 300$.

Figure 11 presents arbitrary edge planar crack whose initial shape is specified as three segment LS-function (13). The initial conditions for FM-method are obtained by using equations from Section 4.3. The signed distance LS function and arrival time in computational domain are obtained with eqn. (16) where $f_{ij} = 1$ and $f_{ij} = 1/F_{ij}$, respectively. In the Fig. 11 the numerical simulation of the crack front evolution is shown for 100 time steps.

For the FM method we used 48x32 elements in the plane of the planar crack. In this example all intersection points have equal velocities due to the fact that all lines of crossing time map are equidistant.

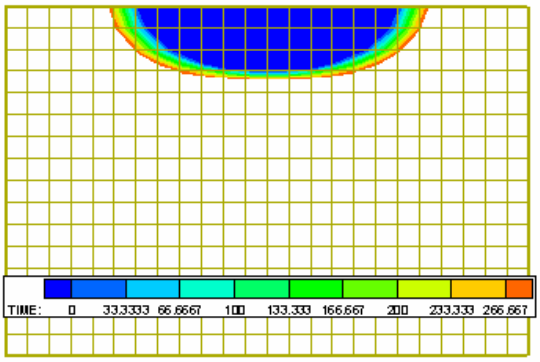


Figure 9. The front configuration semi-elliptical edge crack until time step $T = 300$

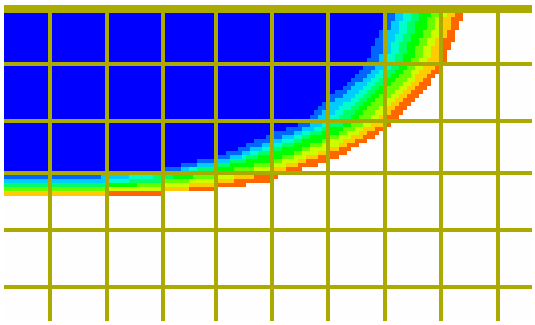


Figure 10. Semi-elliptical edge planar crack front at the time step $T = 0$ and the step $T = 300$

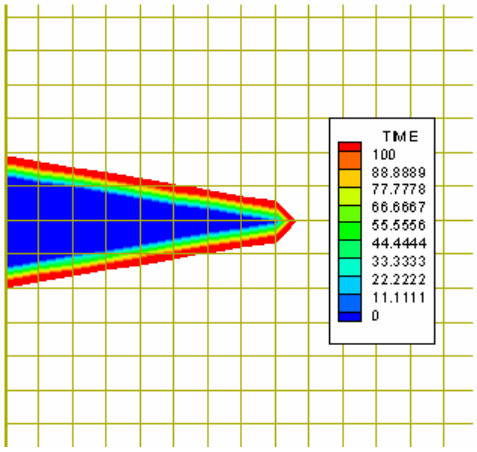


Figure 11. The arrival time of the planar edge polygonal crack

In the Fig. 11 crack front propagation of the planar edge polygonal crack is shown, when front velocities at intersection points are independent of the front position.

In the Fig. 12, crack front propagation of the planar edge polygonal crack is shown, when the front velocities at the intersection points only depend on the front position.

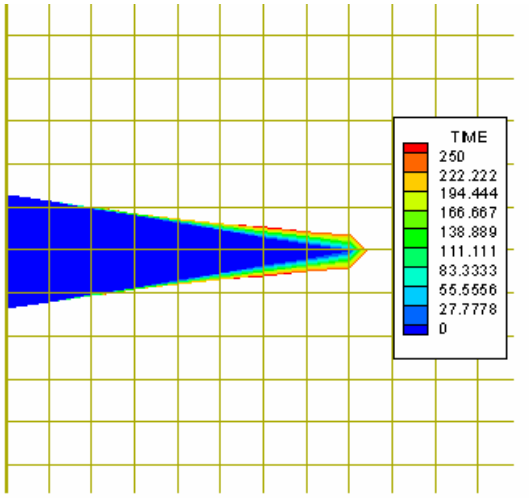
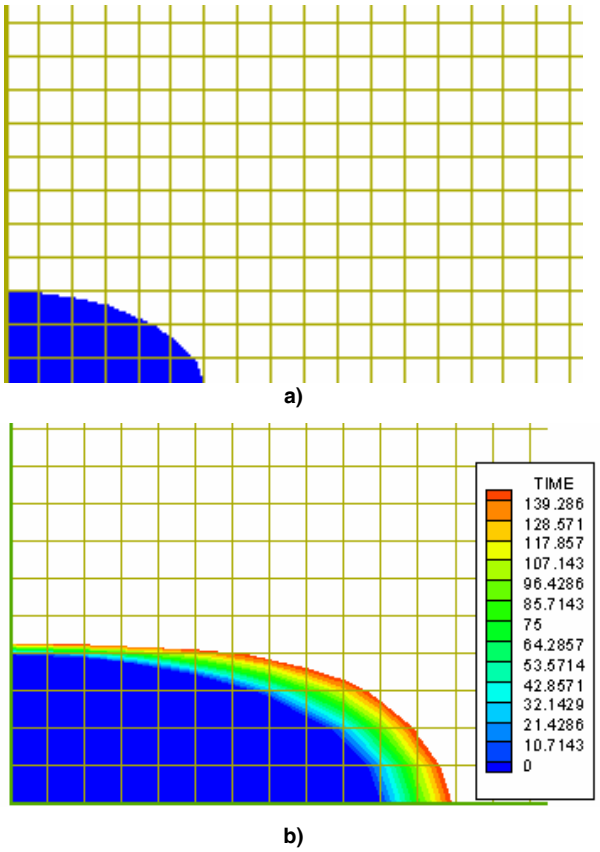


Figure 12. The front configuration of the planar edge polygonal crack until time step $T = 250$



a)

b)

Figure 13. a) Quarter – elliptical corner crack b) Arrival time for quarter-elliptical corner crack until time step $T = 150$

Figure 13.a presents a quarter–elliptical corner crack with $a/b = 0,85/0,5$.

The enlarged part of the computational domain with quarter-elliptical corner crack front is shown in Fig. 13.b. Also, Fig. 13.b shows arrival time (crossing time mapping) for a quarter–elliptical corner crack front.

7. CONCLUSION

The LS method is a numerical technique for tracking moving interfaces [11], [12], [13]. The related FM method is computationally attractive alternative for strictly monotonically advancing fronts. In both methods, the evolving interface is represented as a level contour of a function of one higher dimension (i.e., $\psi(\mathbf{x},t) = C$). In the FM method, the motion of the interface is embedded in solution of an elliptic equation in terms of $\psi(\mathbf{x},t)$. A first-order upwind finite difference scheme [14], was adopted in the FM method. The performance of the new technique for planar arbitrary static crack was studied. We demonstrated that an initial penny-shaped crack remained circular in shape [15], [16]. In addition, we also studied the growth of an edge elliptical and edge polygonal planar crack.

By using the present technique, a fully automated crack growth simulation is carried out without the need to remesh the crack during its evolution. This is in contrast to finite element method based on re-meshing, which engenders significant complexity in maintaining and describing the crack geometry during crack growth analysis [16], [17].

REFERENCES

[1] Sethian J.A.: *Fast marching methods and level set methods for propagating interfaces*,1995

[2] Sethian J.A.: *Level Set Method and Fast Marching Methods - Evolving interfaces in Computational geometry, Fluid mechanics, Computer vision and Material science*, Dept. of Mathematics, University of California, Barkeley, Cambridge University Press,1999.

[3] Ji H., Chopp D.L., Dolbow J.E.: *A hybrid extended finite element / level set method for modeling phase transformations*, 2001.

[4] Smith K.A., Solis F.J., Chopp D.L.: A projection method for motion of triple junctions by level sets, Interfaces and Free Boundaries, 4(3):263-276. 2002.

[5] Sethian J.A.: Fast Marching Methods, SIAM Review, 41(2):199-235.1998.

[6] Sethian J.A.: A fast marching level set method for monotonically advancing fronts, Proc. Nat. Acad. Sci., 93(4): 1591-1595, 1996

[7] Adalsteinsson D., Sethian J.A.: A Fast Level Set Method for propagating Interfaces, J.Comp.Phys.,118,2. pp. 269-277, 1995.

[8] Osher S., Sethian J.A.: Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations, Journal of Computational, 79, pp. 12-49, 1988.

[9] Adalsteinsson D., Sethian J.A.: The Fast construction of extension velocities in level set methods, in press. Journal of Computational Physics, 1999.

[10] Chopp D.L., Sukumur N.: Fatigue crack propagation of multiple coplanar cracks the coupled extended finite element / Fast Marching Method, SIAM J. Sci. COMPUT., Vol. 23, No. 1, pp. 1-39, 2001.
[11] Sethian J.A., Chopp D.L.: Physics and geometry of flow under curvature: Singularity formation, Minimal surfaces, Geodesics, and Surface tension, Comm. Math. Phys., 101:487-499, 1995.
[12] Chopp D.L., Sethian J.A.: Motion by intrinsic Laplacian of curvature, Interfaces and Free Boundaries 1., 1-18, 1999.
[13] Sukumur N., Chopp D. L., Moran B.: Extended Finite Element Method and Fast Marching Method for Three-Dimensional Fatigue Crack Propagation, Engineering Fracture Mechanics, 2002.
[14] Jovicic N., Jovicic G., Babic M., Milovanovic D.: Computational Efficiency of Artificial Dissipation Models for Three-Dimensional Incompressible Flow Solutions, XXIII Yugoslav congress of theoretical and applied mechanics, Belgrade, Yugoslavia, 2001.
[15] Stolarska M., Chopp D.L., Moës N., Belytschko T.: Modelling crack growth by level sets in the extended finite element method, International Journal for Numerical Methods in Engineering, 51(8):943-960, 2000.
[16] Sukumur N., Moës N., Moran B., Belytschko T., Extended finite element method for three-dimensional crack modeling, Int. J. Numer. Meth. Enging, 48:1549-1570, 2000.
[17] Sukumur N., Chopp D.L., Moës N., Belytschko T.: Modeling holes and inclusions by level sets in the extended finite-element method, Comput. Methods Appl. Mech. Engin., 190, 6183-6200, 2001.

НУМЕРИЧКО МОДЕЛИРАЊЕ РАСТА ПРСЛИНЕ УПОТРЕБОМ *LEVEL SET FAST MARCHING* МЕТОДА

Гордана Јовичић, Мирослав Живковић, Небојша Јовичић

Fast Marching је нумеричка техника за математичко моделирање прслина произвољног облика, отвора, интерфејса два материјала, матријалних укључака, без потребе за меширањем интерних граница. Ова техника је прорачунски атрактивна за решавање проблема монотоно расућих фронтова. Развој интерфејса се репрезентује функцијом контурног нивоа која је у простору увек за ред виша од реда разматраног проблема. Ова техника се базира на коришћењу методе коначних разлика за хиперболичке конзервативне законе која омогућава прецизну и стабилну симулацију развоја оштрих ивица и врхова на интерфејсу.

Fast Marching метод омогућава потпуно аутоматизовану симулацију раста прслине. У овом раду приказана је методологија, алгоритам и имплементација развијеног софтвера за симулацију раста раванске прслине у 3Д простору.