NUMERICAL MODEL WITH EXPLICIT TIME INTEGRATION SCHEME FOR TRACKING INTERFACES

Ewa Węgrzyn-Skrzypczak[1], Tomasz Skrzypczak[2]

[1] Institute of Mathematics, Czestochowa University of Technology
Częstochowa, Poland
[2] Institute of Mechanics and Machine Design, Czestochowa University of Technology
Częstochowa, Poland
[1] ewa.skrzypczak@im.pcz.pl, [2] skrzyp@imipkm.pcz.pl

Abstract. In this paper a simple and effective method for tracking interfaces in two--dimensional area is described. The presented approach is very attractive in solving Stefan problems where moving internal boundaries occur. It is based on the level set method (LSM) and uses the so-called distance function. A numerical model based on the finite element method (FEM) is proposed.

Keywords: *moving interface, level set method, finite element method*

Introduction

There is a wide-ranging account of the mathematical formulation of problems involving moving boundaries as they occur in such areas as hydrology, metallurgy, chemical engineering, soil science, molecular biology, materials science, etc. One of the most popular numerical techniques for tracking interfaces and shapes is LSM. The advantage of LSM is that one can perform numerical computations involving curves and surfaces on a fixed grid. Also, LSM makes it very easy to follow boundaries when they split, develop holes, etc. Level set method was invented by J.A. Sethian and S. Osher in 1988 [1] and extensively developed for the next 20 years [2-7].

The quantity that determines the temporary position of the interface is called distance function and indicated by φ [m]. It measures current distance from any point to the moving interface. Thus the position of the interface is determined by $\varphi = 0$. There are two kinds of the level set method - global and local LSM. The first one requires calculating φ for the entire grid, so it is a time-consuming method. The second allows one to calculate φ only in a narrow band at the interface saving a lot of time.

Shape and location of the interface can vary over time due to physical factors such as velocity of the solidification front or degree of curvature of the expanding or collapsing surface. In each case mathematical descriptions of such phenomena

are very similar. In particular, the local LSM is a very useful and popular tool used for tracking interfaces due to its speed, efficiency and simplicity.

1. Mathematical model of the interface tracking

Tracked interface is a curve in 2D or a surface in 3D space. Let's consider a two-dimensional area where distribution of the distance function φ is known (Fig. 1). A temporary position of expanding or collapsing interface corresponds to $\varphi_0 = 0$. It is evident that φ is a function of position $\mathbf{x}$ [m] and time t [s]. The area enclosed inside boundary Γ is marked as Ω_{in}, while the outer area is marked as Ω_{out}.

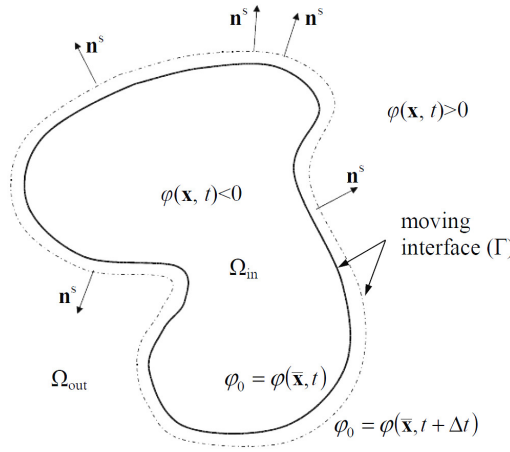

Fig. 1. Moving boundary Γ is described using the distance function φ

The quantity φ satisfies the following condition derived from the LSM [1]:

$$\varphi(\mathbf{x},t) = \min_{\bar{\mathbf{x}}\in\Gamma} \|\mathbf{x}-\bar{\mathbf{x}}\| \operatorname{sgn}\left[\mathbf{n}^s \cdot (\mathbf{x}-\bar{\mathbf{x}})\right] \tag{1}$$

where $\bar{\mathbf{x}}$ [m] is a position of a moving interface Γ and $\mathbf{n}^s$ is the normal vector pointing towards Ω_{out}.

The distance function φ takes the following values

$$\begin{aligned} \varphi(\mathbf{x},t) &> 0, \mathbf{x} \in \Omega_{out} \\ \varphi(\mathbf{x},t) &= 0, \mathbf{x} \in \Gamma \\ \varphi(\mathbf{x},t) &< 0, \mathbf{x} \in \Omega_{in} \end{aligned} \tag{2}$$

A hyperbolic first order differential equation governing the movement of Γ is shown below

$$u_n |\nabla \varphi| = -\frac{\partial \varphi}{\partial t} \tag{3}$$

where u_n [m s^{-1}] is the velocity of the interface Γ in the normal direction computed according to the following formula

$$u_n = \mathbf{n}^s \cdot \mathbf{u} = \frac{\nabla\varphi}{|\nabla\varphi|} \cdot \mathbf{u} \tag{4}$$

Equation (3) describes the so-called initial problem. In the case of such problems only an initial condition must be defined. The distance function φ i.e. position of the interface at $t = 0$ s must be known.

2. Numerical scheme

Starting from the criterion of the method of weighted residuals (3) is multiplied by the weighting function w and integrated over the entire region $\Omega = \Omega_{in} \cup \Omega_{out}$:

$$\int_\Omega w\left(u_n|\nabla\varphi| + \frac{\partial\varphi}{\partial t}\right)dV = 0 \tag{5}$$

Further the weak form of (3) can be written as a sum of integral terms

$$\int_\Omega w u_n|\varphi_{,i}|dV + \int_\Omega w\frac{\partial\varphi}{\partial t}dV = 0 \tag{6}$$

Ω is subjected to the spatial discretization by dividing it into a set of triangular finite elements. They form the mesh that approximates the shape of the area. Approximation accuracy increases with the number of elements. Requested values of φ are determined at the nodes of elements.

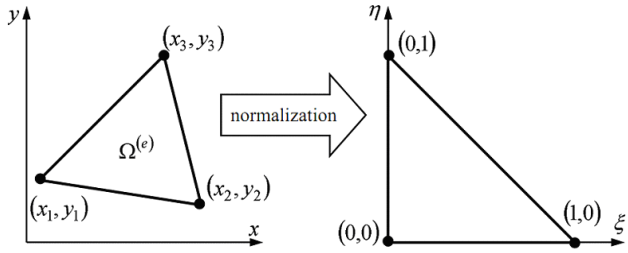

Fig. 2. Normalization of the triangular finite element

Each finite element is subjected to the normalization. This is done by transforming the coordinates of nodes as shown in Figure 2. Normalization simplifies the form of shape functions as well as integrating operations. Linear shape functions in a normalized triangle N_k, $k = 1...3$, take the following form

$$N_1 = 1 - \xi - \eta, \quad N_2 = \xi, \quad N_3 = \eta \tag{7}$$

Figure 3 indicates three areas which may include finite elements. The distance function may be calculated in the whole mesh (global LSM) but to determine temporary position of the interface it is enough to use elements that contain it, together with its closest neighbors (local LSM).

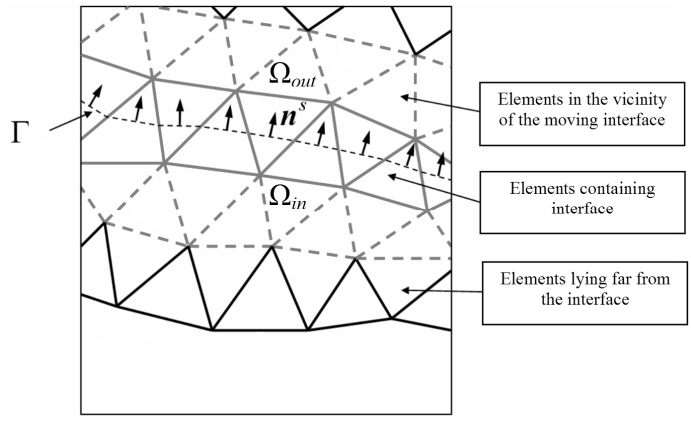

Fig. 3. The narrow band of triangular elements at the interface Γ

The approximation of a distance φ using linear shape functions for the triangle (7) is as follows:

$$\varphi = N_k \varphi_k, \quad \varphi_{,i} = N_{k,i} \varphi_k, \quad \frac{\partial \varphi}{\partial t} = N_k \frac{\partial \varphi_k}{\partial t}, \quad k = 1 \ldots 3 \tag{8}$$

In accordance with the standard Galerkin formulation the weighting functions are the same as the shape functions. Using the substitution (8) in (6) the formula for a single triangle is

$$u_n^{(e)} \int_{\Omega^{(e)}} N_l \left| N_{k,i} \varphi_k \right| dV + \int_{\Omega^{(e)}} N_k N_l dV \frac{\partial \varphi_k}{\partial t} = 0 \tag{9}$$

The matrix forms of the integral terms of equation (9) are as follows:

$$\mathbf{B}_{\varphi}^{(e)} = u_n^{(e)} \int_{\Omega^{(e)}} \begin{bmatrix} N_1 \left| N_{k,i} \varphi_k \right| \\ N_2 \left| N_{k,i} \varphi_k \right| \\ N_3 \left| N_{k,i} \varphi_k \right| \end{bmatrix} dV \tag{10}$$

$$\mathbf{M}_{\varphi}^{(e)} = \int_{\Omega^{(e)}} \begin{bmatrix} N_1 N_1 & N_1 N_2 & N_1 N_3 \\ N_2 N_1 & N_2 N_2 & N_2 N_3 \\ N_3 N_1 & N_3 N_2 & N_3 N_3 \end{bmatrix} dV \tag{11}$$

As a result of substitutions (10) and (11) in (9) the following expression is obtained

$$\mathbf{M}_{\varphi}^{(e)}\dot{\mathbf{\Phi}} + \mathbf{B}_{\varphi}^{(e)} = 0 \tag{12}$$

After the introduction of time grid

$$t^0, t^1, \ldots, t^f, t^{f+1}, \ldots, t^n \tag{13}$$

and the following scheme

$$t \in \left[t^f, t^{f+1}\right] : \dot{\mathbf{\Phi}} = \frac{\mathbf{\Phi}^{f+1} - \mathbf{\Phi}^f}{\Delta t} \tag{14}$$

equation (12) takes the following form

$$\mathbf{M}_{\varphi}^{(e)} \frac{\mathbf{\Phi}^{f+1} - \mathbf{\Phi}^f}{\Delta t} + \mathbf{B}_{\varphi}^{(e)} = 0 \tag{15}$$

After elementary transformations and aggregation a global system of equations according to an explicit time integration scheme is obtained

$$\mathbf{\Phi}^{f+1} = \mathbf{\Phi}^f - \Delta t \mathbf{M}_{\varphi}^{-1} \mathbf{B}_{\varphi} \tag{16}$$

This system of equations is built only for selected finite elements containing the front or lying in its nearest neighborhood (Fig. 3). This approach helps to speed up calculating the position of the front.

The values of vector $\mathbf{\Phi}^{f+1}$ require re-initialization [8, 9], to satisfy the following condition

$$\left|\nabla \varphi\right| = 1 \tag{17}$$

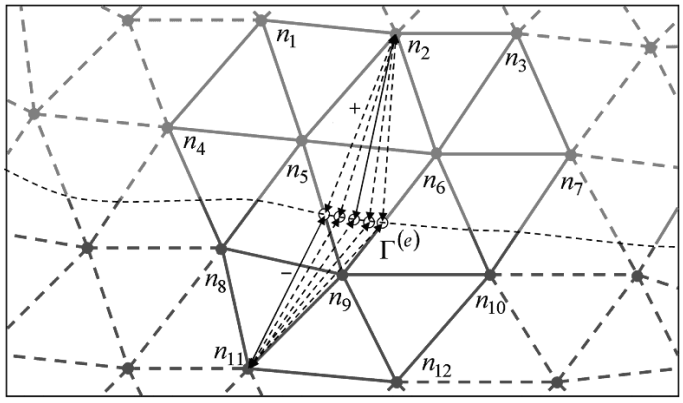

Fig. 4. The idea of re-initialization of the interface position

The idea of the method of re-initialization is shown in Figure 4. It involves the introduction of control points on the section $\Gamma^{(e)}$. Then the distance between them and the nodes lying in the neighborhood are calculated and the smallest value is remembered. The operation is repeated for the whole front. The method is simple and fast because of its locality.

References

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

[2] Adalsteinsson D., Sethian J.A., A fast level set method for propagating interfaces, Journal of Computational Physics 1995, 118, 269-277.

[3] Barth T., Sethian J., Numerical schemes for the Hamilton-Jacobi and level set equations on triangulated domains, Journal of Computational Physics 1998, 145, 1-40.

[4] Sethian J.A., Fast marching methods, SIAM Review 1999, 41, 199-235.

[5] Sethian J.A., Curvature and the evolution of fronts, Communications in Mathematical Physics 1985, 101, 487-499.

[6] Sethian J.A., Numerical methods for propagating fronts, [in:] Variational Methods for Free Surface Interfaces, New York 1987, 155-164.

[7] Sethian J.A., Fast marching methods and level set methods for propagating interfaces, [in:] von Karman Institute Lecture Series, Computational Fluid Mechanics 1998.

[8] Peng D., Merriman B., Osher S., Zhao H., A PDE-based fast local level set method, Journal of Computational Physics 1999, 155, 410-438.

[9] Sussman M., Fatemi E., An efficient interface preserving level set re-distancing algorithm and its applications to interfacial incompressible fluid flow, Journal of Scientific Computing 1999, 20, 1165-1191.