Applied Mathematical Modelling 39 (2015) 7291–7302

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

An efficient and parallel level set reinitialization method – Application to micromechanics and microstructural evolutions

Modesar Shakoor *, Benjamin Scholtes, Pierre-Olivier Bouchard, Marc Bernacki

Mines-ParisTech, PSL-Research University, CEMEF – Centre de mise en forme des matériaux, CNRS UMR 7635, CS 10207 rue Claude Daunesse, 06904 Sophia Antipolis Cedex, France

ARTICLE INFO

Article history:
Received 3 October 2014
Received in revised form 13 February 2015
Accepted 4 March 2015
Available online 19 March 2015

Keywords:
Level set reinitialization
k-d tree
Finite element method

ABSTRACT

The paper introduces a new parallel and efficient algorithm for the reinitialization of level set functions on unstructured finite element (FE) meshes in two and three dimensions. The originality of this implementation lies in the use of a direct method enhanced by a *k-d tree* space partitioning technique. Different test cases illustrate the potential of the method for typical metallurgical and micromechanical problems with isotropic and anisotropic meshes. Comparison with other classical reinitialization methods, such as Hamilton–Jacobi formulations, proves that the proposed method guarantees optimal accuracy together with importantly reduced computational costs.

1. Introduction

Most problems in materials science at the microscale involve multiphase formulation and tracking of dynamic interfaces, e.g. recrystallization, grain growth, fracture mechanics, ductile damage, etc. Thanks to the explosion of large scale parallel computations, these phenomena can now be simulated at a representative volume element (RVE) scale. Among existing approaches, the level set (LS) method introduced in 1988 [1] receives a growing attention. In the LS method each phase of the simulation is represented by a LS function evaluated at mesh nodes. This function can simply be defined as a binary function equal to 1 inside the phase and 0 elsewhere, as proposed in [2]. This approach does nevertheless not provide a precision greater than the mesh size. A well-known alternative consists in defining the LS function $\psi(x,t)$ on the domain Ω as a signed distance function to the interface Γ:

$$\forall t \begin{cases} \psi(x,t) = \pm d(x,\Gamma(t)), \ x \in \Omega, \\ \Gamma(t) = \{x \in \Omega, \psi(x,t) = 0\}, \end{cases} \tag{1}$$

where $d(.,.)$ is the Euclidean distance and the sign depends on whether the node is located inside or outside of the represented object. In an Eulerian context, the evolution of this function is classically computed by solving a transport equation involving the velocity field $\vec{v}$. This formulation has been employed to simulate a wide variety of mechanical and metallurgical phenomena [3–6].

* Corresponding author.
E-mail address: modesar.shakoor@mines-paristech.fr (M. Shakoor).

http://dx.doi.org/10.1016/j.apm.2015.03.014
0307-904X/

A major drawback of this formulation lies in the fact that after transport, in general, the function is no longer a distance function. This is particularly problematic when a specific remeshing technique depending on the distance property is used at the interface [3]. In addition, some phenomena, such as curvature-driven interface motions, require a distance function at least in a thin layer around the interface in order to compute properly the corresponding velocity field [7]. Finally, the conditioning of the transport problem also depends on the regularity of the LS function [8].

For these reasons, the distance function needs to be reinitialized. Indeed, restoring the metric property is equivalent to solving the following eikonal equation:

$$\forall t \begin{cases} \|\nabla\psi(x,t)\| = 1, \ x \in \Omega, \\ \psi(x,t) = 0, \ x \in \Gamma(t). \end{cases} \tag{2}$$

There exist different approaches to solve this equation including the well-known Fast Marching Method introduced by Sethian [9] which propagates a front from the interface and ensures directly a gradient equal to unity. Though this approach has been later extended to unstructured meshes [10], its implementation becomes extremely complicated when it comes to consider anisotropic (*i.e.* obtuse) triangulations [11]. The latter relies on the insertion of numerical supports for obtuse triangle and is mentioned as "cumbersome" in [11]. To our knowledge, this variant is not used in the recent literature. Another major drawback of the Fast Marching Method lies in the parallel implementation. More specifically, the algorithm has to be performed several time on each partition to synchronize the values between the processors, which requires significant implementation effort and poor parallel efficiency.

In [12], a Hamilton–Jacobi (H–J) formulation equivalent to (Eq. 2) was proposed in order to correct iteratively the level set values around the interface by solving a partial differential equation (PDE). This method thus requires the definition of a purely numerical parameter known as the fictive time step for reinitialization $\Delta\tau$. This quantity is generally of order of the mesh size h in the direction normal to the interface. For convenience we assume in the following that $\Delta\tau = h$. By noting ε the reinitialized thickness, $\varepsilon/\Delta\tau$ increments are then needed to reinitialize completely the layer $\psi \in [-\varepsilon, \varepsilon]$.

More recently, coupled convection-reinitialization (CR) methods emerged wherein the LS function is automatically reinitialized during the resolution of the transport equation [8]. Their main advantage lies in the fact that only one solver is needed for the simulation instead of two for the classical H–J technique. The signed distance function can also be replaced by any smooth function which satisfies the metric property, at least in a thin layer around the interface. In the following these two variants will be mentioned: the former using a classical distance function (CR–DF) and the latter working with a hyperbolic tangent distance function $\tilde{\psi} = E \cdot \tanh(\psi/E)$ (CR–HTDF). Since the hyperbolic tangent function has a gradient close to one only in the neighborhood of zero, the truncation thickness E has to be chosen big enough to verify the metric property at least in a thin layer around the interface.

Finally, a natural way to reinitialize LS functions consists in using a brute force algorithm to perform a complete reconstruction of the distance function. This technique works in two steps: discretize the interface (zero-isovalue of the LS function) into a collection of simple elements and, for every node, compute the distance to all elements of the collection and store the smallest one which becomes the updated value of the distance function. Though it guarantees optimal accuracy, this Direct Reinitialization (DR) technique is generally mentioned as extremely greedy in terms of computational requirements in the literature [12,13]. Hence it is carefully avoided in most implementations, with the exception of [6]. In [14], a review of various improvements to this method proposed in literature to overcome this difficulty can be found. These works generally address only regular grids or hierarchical meshes [15].

In the following, the DR method is investigated and a new parallel and efficient implementation is proposed for unstructured and possibly anisotropic meshes. It is then compared to other approaches in terms of accuracy and numerical performances. Applications addressed here cover full field grain growth simulations and ductile damage modeling at the microscale.

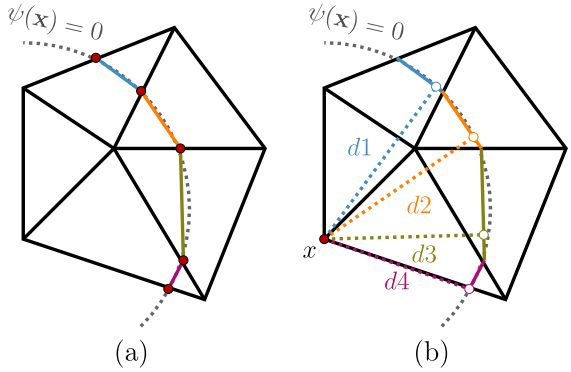

Fig. 1. Direct Reinitialization method on a P1 mesh: (a) collection construction; (b) distance computation.