Numerical implementation of the eXtended Finite Element Method for dynamic crack analysis

Ionel Nistor, Olivier Pantalé, Serge Caperaa

Numerical implementation of the eXtended Finite Element Method for dynamic crack analysis

Ionel Nistor, Olivier Pantalé *, Serge Caperaa

L.G.P C.M.A.O – E.N.I.T, 47 Av d'Azereix BP 1629, 65016 Tarbes Cedex, France

Abstract

A numerical implementation of the eXtended Finite Element Method (X-FEM) to analyze crack propagation in a structure under dynamic loading is presented in this paper. The arbitrary crack is treated by the X-FEM method without re-meshing but using an enrichment of the classical displacement-based finite element approximation in the framework of the partition of unity method. Several algorithms have been implemented, within an oriented object framework in C++, in the home made explicit FEM code. The new module, called DynaCrack, included in the dynamic FEM code DynELA, evaluates the crack geometry, the propagation of the crack and allow the post-processing of the numerical results. The module solves the system of discrete equations using an explicit integration scheme. Some numerical examples illustrating the main features and the computational efficiency of the DynaCrack module for dynamic crack propagation are presented in the last section of the paper.

Keywords: Partition of unity; eXtended Finite Element Method; Finite element programming; Dynamic crack propagation; Dynamic energy release rate

1. Introduction

The development of computational techniques for the analysis of dynamic fracture and their implementation in numerical codes are becoming more and more important in recent years. Such interest is motivated by the desire to predict both the initiation of a crack and its propagation through the structure under dynamic loading. This is a typical case concerning impact applications where severe dynamic loading induces damage and fracture of the material. Several numerical approaches have been proposed in the last decades for analyzing some discontinuous phenomena, such as cracks and shear bands, occurring in structures under quasi-static or dynamic loading.

The first category concerns the re-meshing methods that are usually used for modeling cracks or other strong discontinuities in structures. Based on classical finite element method (FEM), the geometry is usually re-meshed at each time step during the discontinuity propagation. In the most recent developments, the re-meshing area has been limited to the immediate vicinity of the discontinuity to save computational time. Because of its simplicity (a standard FEM program and a re-meshing algorithm are sufficient to evaluate crack initiation and propagation), different versions of this technique have been implemented in commercial codes, especially for quasi-static analysis. Nevertheless, several important drawbacks remain. The mesh dependence of the crack is one of the main. The user must have "a priori" knowledge of the response of the model in order to generate an accurate initial mesh in the crack-tip region; beside that, the direction of the crack propagation is usually very sensitive with nodes alignment. Another important difficulty is the remapping of the data attached to physical points situated around the crack between the old mesh and the new one. For dynamic fracture problems, this approach remains quite difficult to apply.

Discontinuity methods appeared as an innovating technique to model crack growth using cohesive segments at

* Corresponding author.
E-mail address: Olivier.Pantale@enit.fr (O. Pantalé).
URL: http://www.enit.fr (O. Pantalé).

doi:10.1016/j.advengsoft.2007.06.003

the element's interface without the necessity to introduce supplementary nodes. This approach was used by Xu and Needleman [1] and Camacho and Ortiz [2]. Its numerical implementation for an explicit time integration scheme was presented by Remmers and co-workers [3]. Based on the use of the partition of unity method (PUM) developed by Babuska and Melenk [4], a crack is represented by a number of overlapping cohesive segments which are inserted as discontinuities in the displacement field of the elements cut by the crack. In this approach, the crack direction is limited to the element edges and thus the crack paths are limited to specified directions [5]. Even based on substantial theoretical foundations, this approach is not still completely well-contained. For instance, the link between the parameters of the cohesive law used in the models with measurable material properties is not well known.

The embedded discontinuity methods represent another class of fracture methods that consider the crack at the element level as a band of high strain. This approach was proposed by Belytschko, Fish and Engelmann [6]. A good description of these methods is given by Dvorkin et al. [7], Simo et al. [8] and references quoted. Jirasek [9] published a comparative study of these methods and showed that there is three major classes of models with embedded discontinuities. One of these classes is known as the statically and kinematically optimal non-symmetric (SKON) formulation [9] which allow to represent both the kinematic and the static aspects properly and leads to an improved numerical performance. It is able to effectively represent the complete separation at later stages of the fracturing process without the transfer of spurious stress. Recently Oliver and Huespe [10] presented a numerical implementation of this approach based on the use of finite elements with embedded discontinuities where both nodal and elemental enrichments are taken into account.

Finally we present in this literature survey the eXtended Finite Element Method (X-FEM). The general idea of this method is to enrich the displacement approximation space spanned by standard finite element shape functions with some specific discontinuous functions. It is about an approach based on PUM. To our knowledge, Belytschko and Black [11] have been the first one to model a crack using this approach. The name X-FEM was given by Moes et al. [12] and Dolbow et al. [13] after the introduction of a step function enrichment in the displacement field for the elements entirely cut by the crack. This method we chosen for implementation in the home made DynaCrack code in order to carry out dynamic analysis of structures containing discontinuities.

Concerning the restricted application field relative to dynamic crack analysis, a few contributions based on the X-FEM formulation have been published. The major contribution has been proposed by Belytschko et al. [5] with an adaptation of the X-FEM approach for dynamic crack analysis and the development of a new discontinuous enrichment. They proposed a crack evolution model based on the loss of hyperbolicity criterion. Réthoré et al. [14] investigated some instability problems occurring in X-FEM dynamic crack analysis. They proposed a technique called "balance recovery method", that provide the ability to evaluate both numerical stability and accuracy for any type of projection used with varying meshes. Recently, Menouillard et al. [15] proposed a new technique to evaluate the lumped mass matrix for the enriched elements in X-FEM. This is a non-trivial result because of the additional degrees of freedom linked to the enriched nodes. This later allows to use an explicit integration scheme where the critical time step does not tend to zero when the crack is in the immediate vicinity of a node.

The present paper is organized as follows. A description of the X-FEM with a focus on the explicit time integration scheme is presented in Section 2. The main features of the crack evolution model and its numerical implementation are presented in Section 3. The entire procedure used for the numerical integration in the DynELA FEM code [16] is presented in Section 4. Numerical simulations illustrating the robustness and the effectiveness of the implemented algorithms are presented in Section 5. A brief review of the problems encountered during this work and some conclusions and future works are reported in the last section of this paper.

2. eXtended Finite Element Method for dynamic crack analysis

The crack representation in X-FEM is based on the enrichment of the classical displacement-based finite element approximation through the framework of the partition of unity method. Therefore, a crack is modeled by introducing a set of additional degrees of freedom to the nodes whose nodal shape function space intersects this crack. Within the approach proposed by Moes et al. [12], two types of nodal enrichments are considered in order to model the crack. When an element is splitted in two parts by the crack, all the nodes of this element are enriched by the Heaviside step function, while, the Westergaard asymptotic function is used to enrich the nodes of the elements containing the crack tips. Therefore, in a 2-D analysis for example, the nodes of fully-cut elements have two classical degrees of freedom and two enriched degrees of freedom modeling the strong displacement jump. The nodes of the elements containing the crack-tip have two classical degrees of freedom and four enriched degrees of freedom based on the radial and angular behavior of the asymptotic displacement field. As the crack propagates and the crack-tip crosses the edges of the elements, the enrichment status of some nodes of the structure changes. This type of enrichment was implemented by Sukumar et al. [17] for a quasi-static crack. Fig. 1a shows a finite element mesh where the circled nodes are enriched with the Heaviside step function, while the squared nodes are enriched with the Westergaard functions.

As discussed by Belytschko et al. [5], this mixed enrichment is not easy to incorporate in methods with time-dependent solutions. Therefore, they proposed recently

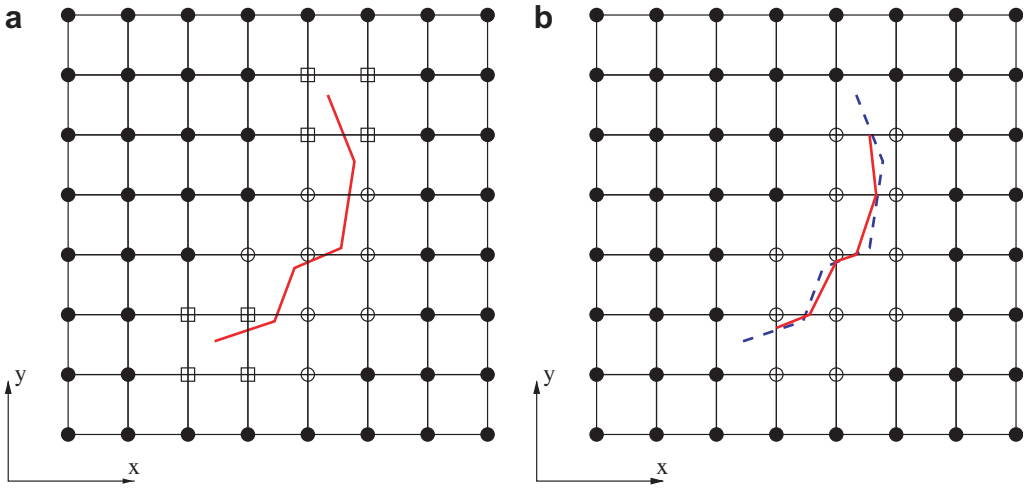

Fig. 1. Crack modeling using X-FEM approach.

another approach for the enrichment in dynamic applications. The basic idea is to avoid the near-tip enrichment by imposing the crack-tip to cross one element at a time. Essentially the crack-tip goes from edge to edge and the enrichment for such a situation can be treated using only Heaviside step functions. In Fig. 1b, we present for the same mesh and arbitrary crack the new enrichment and the new path for the crack resulting from this approach. The major consequence of limiting the crack-tip position at the element's edge is that the modification of the direction will only occur at the element edges. The comparison of Fig. 1a and b shows that the mesh size plays a more important role, since a finer mesh helps to minimize the errors due to the crack path approximation and the fact that the crack tips are not taken into account. In the same time, being a simpler procedure for implementation in dynamic crack analysis, we have chosen to use it. This involves a special choice in the crack evolution models (as presented further in Section 3), in order to avoid that the propagation models require quantities not accurately computed within this framework.

2.1. Crack modeling in 2-D

To model a given crack geometry within the X-FEM approach, a criterion for the selection of the enriched nodes is necessary. In the mainly used approach in the literature [12,17], the support of nodal shape functions is defined as the union of the elements connected to the node I. If this support of nodal shape functions is intersected by the crack, then the node I is enriched with a discontinuous function based on the Heaviside step function $H(\vec{X})$. The use of the generalized Heaviside step function $H(\vec{X})$ allows to represent the "jump" in the displacement field across the crack. In the proposed approach, $H(\vec{X})$ takes the value $+1$ above the crack and -1 below the crack for a given direction of the crack:

$$H(\vec{X}) = \begin{cases} +1, & \text{if } (\vec{X} - \vec{X}^*) \cdot \vec{n} \geqslant 0 \\ -1, & \text{otherwise} \end{cases} \tag{1}$$

where $\vec{X}$ is the considered point in the initial configuration, $\vec{X}^*$ is the projection of $\vec{X}$ onto the crack and $\vec{n}$ is the unit outward normal to the crack at $\vec{X}^*$. Each node I of the mesh is associated to a shape function $\phi_I(\vec{X})$. The new discontinuous displacement field $\vec{u}^h(\vec{X})$ for a N nodes mesh, including N_Γ enriched nodes, is therefore approximated by

$$\vec{u}^h(\vec{X}) = \sum_{I \in N} \phi_I(\vec{X})\vec{u}_I + \sum_{I \in N_\Gamma} \phi_I(\vec{X})H(\vec{X})\vec{a}_I \tag{2}$$

where $\vec{u}_I$ denotes the classical degrees of freedom and $\vec{a}_I$ are the enriched degrees of freedom relative to node I. This approach is quite the same as the one proposed by Moes et al. [12] except that the Westergaard contribution has been removed.

The numerical implementation of X-FEM presented in this paper was achieved for a 2-D analysis and a single type of element: quadrilateral four nodes element and an example illustrating the enrichment used is presented in Fig. 1b.

2.2. Discrete equations

The proposed approach is based on an elastodynamics behavior for the X-FEM analysis of a cracked homogeneous domain Ω, in the current configuration, as presented in Fig. 2. The crack is represented by the boundary Γ_c used to represent the two lips. A traction force vector $\vec{t}$ is applied on the Neumann boundary Γ_t and $\vec{u}$ is the applied displacement vector on the Dirichlet boundary Γ_u. It can be noted that $\Gamma_u \cup \Gamma_t = \Gamma$ and $\Gamma_u \cap \Gamma_t = \emptyset$. Crack lips are considered traction-free. Thus, we can write the strong form of the momentum conservation law in terms of the Cauchy stress tensor, for the current configuration described in Fig. 2, as follows:

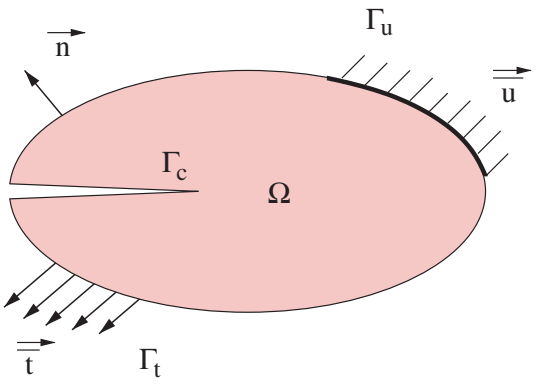

Fig. 2. Notations used for a 2-D domain.

$$\frac{\partial \sigma_{ij}}{\partial x_j} + \rho b_i - \rho \ddot{u}_i = 0 \in \Omega \tag{3}$$

$$\sigma_{ij} n_j = \bar{t}_i \in \Gamma_t \tag{4}$$

$$u_i = \bar{u}_i \in \Gamma_u \tag{5}$$

where ρ is the current density, $\vec{b}$ is the body force vector per unit mass, σ is the Cauchy stress tensor, $\vec{n}$ is the external unit vector to Γ and $(\ddot{})$ is the second time derivative of $()$. The weak form of the momentum equation in the current configuration is then given by

$$\int_\Omega \delta u_i \rho \ddot{u}_i \mathrm{d}\Omega = \int_\Omega \delta u_i \rho b_i \mathrm{d}\Omega + \int_{\Gamma_t} \delta u_i \bar{t}_i \mathrm{d}\Gamma_t - \int_{\Omega \setminus \Gamma_c} \frac{\partial(\delta u_i)}{\partial x_j} \sigma_{ij} \mathrm{d}\Omega \tag{6}$$

where u_i is the trial displacement field (see Eq. (2) for the definition of u_i) and δu_i is the test displacement field. The equilibrium discrete system of equations for dynamic analysis with X-FEM is obtained from Eq. (6) using the standard Bubnov–Galerkin procedure. Substituting trial and test displacement fields and their derivatives yield to the following system:

$$\begin{bmatrix} M_{IJ}^{uu} & M_{IJ}^{ua} \\ M_{IJ}^{au} & M_{IJ}^{aa} \end{bmatrix} \begin{bmatrix} \ddot{u}_J \\ \ddot{a}_J \end{bmatrix} + \begin{bmatrix} F_{iI}^{int} \\ Q_{iI}^{int} \end{bmatrix} - \begin{bmatrix} F_{iI}^{ext} \\ Q_{iI}^{ext} \end{bmatrix} = 0 \tag{7}$$

where:

$$F_{iI}^{int} = \int_{\Omega \setminus \Gamma_c} \frac{\partial(\phi_I)}{\partial x_j} \sigma_{ij} \mathrm{d}\Omega \tag{8}$$

$$F_{iI}^{ext} = \int_{\Gamma_t} \phi_I \bar{t}_i \mathrm{d}\Gamma + \int_\Omega \phi_I \rho b_i \mathrm{d}\Omega \tag{9}$$

$$Q_{iI}^{int} = \int_{\Omega \setminus \Gamma_c} \frac{\partial(\phi_I H)}{\partial x_j} \sigma_{ij} \mathrm{d}\Omega \tag{10}$$

$$Q_{iI}^{ext} = \int_{\Gamma_t} (\phi_I H) \bar{t}_i d\Gamma + \int_\Omega (\phi_I H) \rho b_i \mathrm{d}\Omega \tag{11}$$

$$M_{IJ}^{uu} = \int_\Omega \rho \phi_I \phi_J \mathrm{d}\Omega \tag{12}$$

$$M_{IJ}^{ua} = \int_\Omega \rho \phi_I (\phi_J H) \mathrm{d}\Omega \tag{13}$$

$$M_{IJ}^{aa} = \int_\Omega \rho (\phi_I H)(\phi_J H) \mathrm{d}\Omega \tag{14}$$

The details concerning the development of the terms in Eqs. 8–14 for the quadrilateral four nodes finite element are presented further in Section 4.3 and the assembling procedure in Section 4.4.

In this work the consistent mass matrix is used because the enriched degrees of freedom obstruct its direct lumping as reported by Belytschko et al. [5]. In fact, neglecting these terms leads to suppress one of the essential information concerning the coupling of the regular and enriched degrees of freedom as de Borst et al. [18] observed. Even the lumping technique proposed by Menouillard et al. [15], for the case of Heaviside step function, does not takes into account the coupling terms. The use of the entire mass matrix increases the CPU time and requires a more powerful processor, but the analyzed models presented here usually contains a quite small number of degrees of freedom, therefore, this choice has been adopted in order to preserve the informations concerning the enrichment.

2.3. Explicit integration scheme

The explicit integration procedure used in the X-FEM module DynaCrack uses the Chung–Hulbert [19] explicit time integration scheme already implemented in the DynELA code [20]. The time integration scheme is given by

$$\ddot{\vec{u}}_{n+1} = \frac{M^{-1}\left(F_n^{ext} - F_n^{int}\right) - \alpha_M \ddot{\vec{u}}_n}{1 - \alpha_M} \tag{15}$$

$$\dot{\vec{u}}_{n+1} = \dot{\vec{u}}_n + \Delta t_{n+1}\left[(1-\gamma)\ddot{\vec{u}}_n + \gamma \ddot{\vec{u}}_{n+1}\right] \tag{16}$$

$$\vec{u}_{n+1} = \vec{u}_n + \Delta t_{n+1} \dot{\vec{u}}_n + \Delta t_{n+1}^2 \left[\left(\frac{1}{2} - \beta\right)\ddot{\vec{u}}_n + \beta \ddot{\vec{u}}_{n+1}\right] \tag{17}$$

The main feature of this algorithm is the presence of a numerical dissipation through its characteristic parameters α_M, β and γ. The values of these parameters are given by the following relations [20]:

$$\alpha_M = \frac{2\rho_b - 1}{1 + \rho_b}; \qquad \beta = \frac{5 - 3\rho_b}{(1+\rho_b)^2(2-\rho_b)}; \qquad \gamma = \frac{3}{2} - \alpha_M \tag{18}$$

where $\rho_b \in [0,1]$ defines the numerical dissipative character of the algorithm. Setting $\rho_b = 1.0$ leads to a conservative algorithm while $\rho_b < 1.0$ introduces numerical dissipation in the scheme. In this work, the conservative algorithm is considered. The integration time step is computed using the following relation:

$$\Delta t = f \frac{l_e}{w_d} \tag{19}$$

where f is a safety factor (the value of $f = 0.82$ has been used here) that accounts for numerical instabilities, l_e is the characteristic length of the smallest element of the structure and w_d is the dilatational elastic wave speed of the material. Note that the elements intersected by the

crack are not taken into account for this computation in order to avoid that the time step tend to zero when the crack path is very close to an enriched node.

2.4. Cut elements partition

For the elements cut by the crack, a special procedure is applied in X-FEM in order to integrate the discrete system of equations. The main idea is not to perform a re-meshing of these elements by adding supplementary nodes on the intersecting points with the crack because this would be contrary to the main principle of X-FEM (modeling of the discontinuities without any dependence with the mesh size and orientation). The main accepted concept in X-FEM is the *partitioning* of these elements. A clear analysis of "element partitioning versus re-meshing" is provided by Sukumar and Prévost [17]. The partitioning procedure in X-FEM is done for numerical integration purpose only, and no additional degrees of freedom are introduced into the discrete space during this operation. Subdividing these elements into triangles in 2-D was proposed by many authors [12,17,21], and several numerical integration options were also presented. On the other way, an original method to perform this integration without any subdivision of the cut elements has been recently proposed by Ventura [22]. Special (higher-order) quadrature rules are usually used for the numerical integration of the elements that are partitioned in this way [12] like a six-point integration rule for triangular elements.

In this work, we subdivide the two zones on both sides of the crack into sub-quadrilaterals, as shown in Fig. 3. The main reason for this partitioning solution is related to the numerical integration accuracy. The numerical computation of the requested quantities such as stiffness matrix, on the sub-quadrilaterals is achieved using the same integration scheme as for all other element of the mesh. In the same time, bilinear shape functions are used to interpolate the fields in order to integrate them. This approach has given more accurate results than the classic sub-triangulation associated to a three or six-point integration rule. The entire geometrical procedure for partitioning the cut elements and the numerical integration algorithm are presented in Section 4.

3. Crack evolution model

For the complete characterization of the dynamic crack propagation, beside the strong form given by Eqs. 3–5, a crack evolution model providing the crack advancing criteria (its direction and its velocity) is necessary. As mentioned earlier in the opening part in Section 2, the choice of the crack evolution model implemented in DynaCrack was strongly influenced by the enrichment. Since only the Heaviside function is taken into account for the crack modeling, the propagation is restricted from edge to edge of the elements. The limitations related to this approach are a quite inaccurate stress–strain evaluation in the immediate vicinity of the crack tips and a loss of smoothness for some fields because the crack propagates in fits and starts. On the other hand, in the context of adapting classical models of crack propagation to this enrichment type, this approach is very attractive for numerical integration and good results can be obtained as presented further in Section 5.

Several propagation criteria were studied and implemented for dynamic crack propagation. The maximum circumferential stress criterion, also called the maximum principal stress criterion by Erdogan et al. [23] was implemented by Moes and Sukumar [12,17] in X-FEM for quasi-static propagation and by Belytschko et al. [5] for dynamic propagation. It sets that a crack will propagate from its actual tip in the direction θ_c where the circumferential stress $\sigma_{\theta\theta}$ is maximum. Physical models used for computing crack propagation based on the energy release rate calculation represent an other important class of criteria for quasi-static and dynamic cracks. Freund [24] has developed a criterion giving an analytical solution for the DSIF (dynamic stress intensity factor) and the connection to the energy release rate for a dynamic stationary crack.

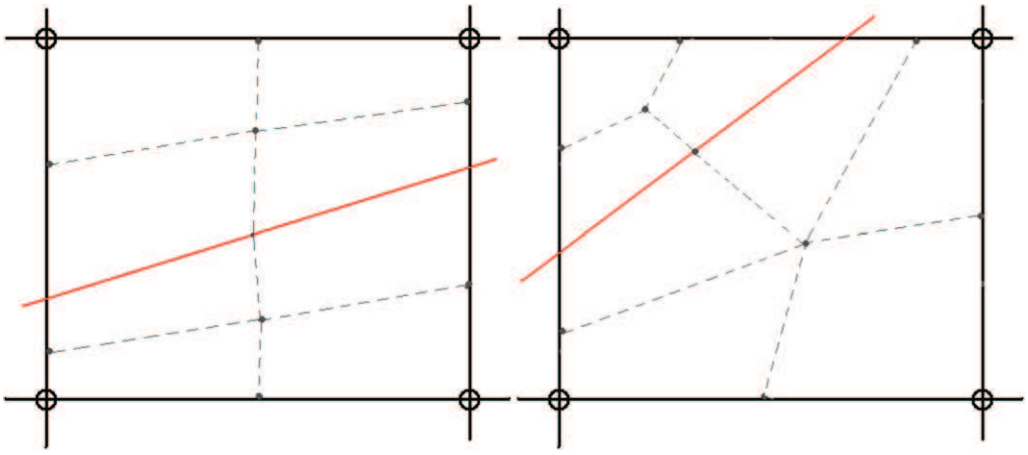

Fig. 3. Partitioning of a four nodes element.

Those criteria have been used in other numerical methods than the finite element method. Krysl and Belytschko [25] and Duarte et al. [26] implemented these criteria in EFGM (element free Galerkin method) and GFEM (generalized finite element method), respectively, for a 3D dynamic crack propagation.

In our X-FEM code, we have chosen to introduce a physical crack evolution model based on Nishioka [27] and Freund [24] approaches and have adapted it to our considered enrichment. The definition of a physical crack evolution model enhances some particular problems since the modeling of the field in the immediate vicinity of the tip is not very accurate in our approach. Therefore it was difficult to obtain an accurate numerical solution around the crack-tip. One of the possible solutions is then to use the path-independent dynamic J-integral characterized by the following features [28]:

- it has the physical meaning of a dynamic energy release rate,
- it gives a unique value for an arbitrary path surrounding the crack-tip,
- it can be related to the dynamic stress intensity factors (DSIF).

The second feature mentioned above is the most interesting one since it allows to use a contour-path quite far from the crack-tip to evaluate the dynamic J-integral. This allow to avoid the inaccurately computed asymptotic field zone around the crack-tip. The analytical form of the J'-integral developed by Nishioka and Alturi [29] for moving dynamic cracks is considered here:

$$J'_k = \int_{\Gamma+\Gamma_c} \left[(W+U)n_k - \sigma_{ij}u_{i,k}n_j\right]\mathrm{d}\Gamma + \int_S (\rho\ddot{u}_i u_{i,k} - \rho\dot{u}_i\dot{u}_{i,k})\mathrm{d}S \tag{20}$$

where, as presented in Fig. 4, W and U are the strain and kinetic energy densities, respectively, S is an area inside of Γ and $\Gamma_c = \Gamma_c^+ + \Gamma_c^-$ represents the crack edge inside of the considered contour Γ. Considering a plane strain approach, the J'-integral components denoted by k in Eq. (20) can be related to the DSIF by

$$J_1'^0 = \frac{1}{2\mu}\left\{A_{\mathrm{I}}(c_s)K_{\mathrm{I}}^2 + A_{\mathrm{II}}(c_s)K_{\mathrm{II}}^2\right\}; \qquad J_2'^0 = -\frac{1}{2\mu}A_{\mathrm{IV}}(c_s)K_{\mathrm{I}}K_{\mathrm{II}} \tag{21}$$

where $A_{\mathrm{I,II,IV}}(c_s)$ are coefficients depending on the propagation crack speed c_s (see [24] for more details) and μ is the shear modulus of the material. The numerical value of the dynamic energy release rate is given by

$$G = J'_1\cos\theta_0 + J'_2\sin\theta_0 \tag{22}$$

DSIF numerical values are extracted using the separation components method proposed by Nishioka and Atluri [29]. This method has been retained because it avoids the need to compute the J'_2 value since this one is known to be sensitive to the near crack-tip singular stress solution. In the separation components method, only the J'_1 (non-affected by the near crack-tip singular solution) and the normal and tangential crack-tip opening displacement components (δ_n and δ_t) contributes to the DSIF evaluation:

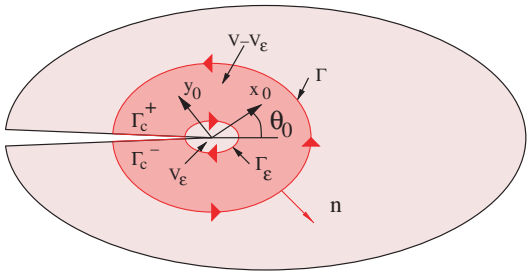

Fig. 4. Evaluation path scheme for the J'-integral.

$$K_{\mathrm{I}} = \delta_n\sqrt{\frac{2\mu J_1'^0\alpha_s}{A_{\mathrm{I}}(c_s)\left(\delta_n^2\alpha_s + \delta_t^2\alpha_d\right)}}; \qquad K_{\mathrm{II}} = \delta_t\sqrt{\frac{2\mu J_1'^0\alpha_s}{A_{\mathrm{II}}(c_s)\left(\delta_n^2\alpha_s + \delta_t^2\alpha_d\right)}} \tag{23}$$

where α_s and α_d are functions depending on the crack propagation speed c_s, the shear wave speed w_s and dilatational wave speed w_d by

$$\alpha_s = \sqrt{1 - \frac{c_s^2}{w_s^2}}; \qquad \alpha_d = \sqrt{1 - \frac{c_s^2}{w_d^2}} \tag{24}$$

The crack propagation model uses the DSIF to answer for the essential questions: is the crack propagates, and if it's true, in what direction and how quickly? Hence, the crack will propagate if the value of the energy release rate G computed from Eq. (22) is greater or equal to a critical limit G_{crit} given by

$$G_{\mathrm{crit}} = K_{\mathrm{ID}}(c_s)\frac{1-\nu^2}{E} \tag{25}$$

where $K_{\mathrm{ID}}(c_s)$ is the dynamic fracture toughness, assumed here equal to $\widehat{K}_{\mathrm{ID}}$ a constant value considered as an intrinsic material property, E and ν are the Young coefficient and the Poisson ratio, respectively. The direction of the crack propagation is then given by

$$\theta_c = 2\arctan\left\{\frac{1}{4}\left(\frac{K_{\mathrm{I}}}{K_{\mathrm{II}}} - \mathrm{sign}(K_{\mathrm{II}})\sqrt{\left(\frac{K_{\mathrm{I}}}{K_{\mathrm{II}}}\right)^2 + 8}\right)\right\}; \quad \text{if} \quad K_{\mathrm{II}} \neq 0 \tag{26}$$

and by $\theta_c = 0$ if $K_{\mathrm{II}} = 0$. The term $\mathrm{sign}(K_{\mathrm{II}})$ inside of the previous equation leads to a positive stress intensity factor along the direction given by θ_c. The crack speed is provided by the numerical propagation algorithm, since the crack-tip advances one edge at a time. As mentioned in Ref. [5], this edge-to-edge approach achieves closure by the discretiza-

tion but the coarse meshes must be avoid. The numerical algorithms implemented for computing the parameters of this crack evolution model will be presented in Section 4.

4. Numerical implementation of the DynaCrack module

The DynELA explicit finite element program [16] has been used for the implementation of the X-FEM crack propagation module. This FEM program is written in C++ and developed within an object-oriented programming (OOP) approach. This feature presents a very well defined mechanism for modular design and re-use of code: in our case this has allowed to develop the new module DynaCrack, re-using several objects already implemented. Some new classes have been implemented and other have been specialized for this application using the inheritance mechanism. In the following sub-sections, we present the major steps of this implementation, pointing out the characteristic keys of X-FEM. Concerning the notation, all keywords related to the DynaCrack code such as the class names, method names, object names ... are written using a `typewriter font`.

4.1. Crack representation and implementation

The crack is represented by the class `CrackFunction` containing a list of $p+1$ points $X_c = \{x_i\}_{i\in[0,p]}$. Within this representation, the two points x_0 and x_p are the crack tips. The initial crack at time $t=0$ is defined through the input data file according to the initial geometry of the crack. The `CrackFunction` class contains a dynamic list updated when adding new points as the crack propagates.

One of the most useful method of this class returns the sign of the Heaviside function for any given point. The so called method `IsOnPositiveSide()` returns a Boolean value: true for positive side (i.e. the Heaviside function $H=1$) and false for negative side (i.e. the Heaviside function $H=-1$) based on geometric predicate evaluation. This is done by computing the projection of this point onto the closest crack segment.

In the presented X-FEM approach, the mesh contains both classical and enriched nodes, therefore, the class `XNode`, inherited from the class `Node` is used for handling the supplementary degrees of freedom (*dof*) corresponding to the enriched nodes. A specific flag `XType` is then used to identify the status of the nodes. During the computation, the status of some nodes changes from normal to enriched because of the crack propagation. The method `getXRType()` is called at each integration time step and is used to update the status of the elements. A method based on an efficient BoundingBox algorithm is used to find the nodes around the crack. For all the elements of the structure three flags are considered according to theirs status:

x elements
- TypeX – enriched elements (from 1 to 3 enriched nodes)
- TypeR – completely cut elements (4 enriched nodes)
- TypeN – normal elements (no enriched nodes)

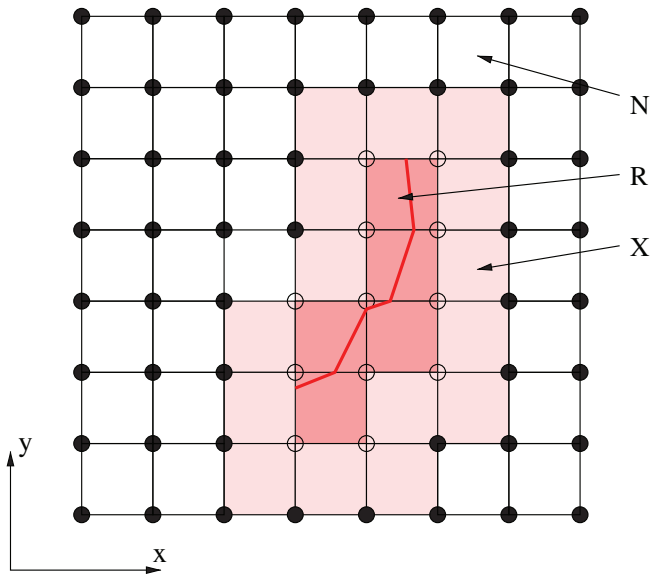

Fig. 5. Enriched element types definition.

Fig. 5 illustrates this aspect for an X-FEM mesh; the enriched nodes are the encircled ones. The two nodes at both end of the edge where the crack-tip is situated are enriched with the Heaviside step function. In the approach proposed by Moes et al. [12] those two nodes are enriched using the Westergaard functions leading to the step function for $\theta = \pi$.

4.2. Partition algorithm

Once the elements status have been updated for the current time step, the partitioning of the so-called `TypeR` elements is done using the methods of the `XPartition` class. An arbitrary crack geometry in a structure discretized with 4-nodes quadrilateral finite elements leads to one of the two situations for cut elements: the crack intersects two opposite or two adjoining edges (see Fig. 3 for more details). The algorithms implemented in the `XPartition` class compute the partition of the elements. As an illustration, the algorithm for the partition of the element presented in Fig. 6 is reported in Box 1.

Box 1 Partitioning algorithm

1. Compute the intersection points, x_{c1} and x_{c2};
2. Build the $\{p_i\}_{i=1\ldots4}$ `xpoints` using nodes location;
3. Compute the positive and negative centroids, C^+ and C^-;
4. Get the median points, m_j, for all sub-domain segments;
5. Build `pointsPositifs` and `pointsNegatifs` lists;
6. Build `Surf2D` surfaces.

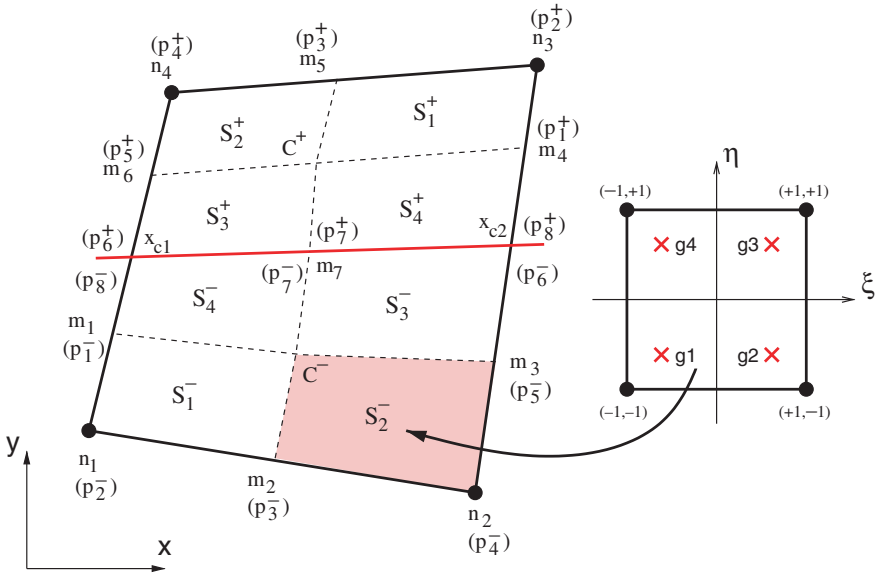

Fig. 6. Integration of a partitioned element.

Two more classes are used to hold the informations for points and surfaces created during the partition, `XPoint` and `Surf2D`, respectively. As the element is sub-divided into two sub-domains, a "positive" one and a "negative" one (according to enriched nodes sign), two lists of `XPoint` objects are created, `pointsPositifs` $\{p_i^+\}$ and `pointsNegatifs` $\{p_j^-\}$, as reported in Fig. 6. The `XPoint` list contains the intersection points between the crack and the element edges, x_{c1} and x_{c2}, the node locations and all middle points m_l of the edges arranged in counter-clockwise order. From the computation of the sub-domain centroids (C^+ and C^-), two lists of `Surf2D` quadrilateral sub-surfaces are generated, $\{S_i^+\}$ and $\{S_i^-\}$. Those two sub-surfaces are used for the numerical integration of the conservative laws and their respective contributions are summed.

4.3. Setting-up of the matrices

One of the main consequence of the additionals *dof* from a numerical point of view is the variable size of the elementary matrices, according to the element status: cut, enriched or normal. In this work, we adopted a block formalism for the setting-up of the matrices as described here after.

The numerical implementation of the DynaCrack module has been done for a 4-node quadrilateral element. For a standard element (`TypeN`), the elementary *dof* vector contains eight terms (two *dof* for each node) and the displacement field is approximated by standard shape functions. For the enriched or cut elements, the size of the elementary *dof* vector is larger (10, 12 or 14 *dof* for `TypeX` and 16 *dof* for `TypeR`). A choice concerning the *dof* placement is necessary: do we put all enriched *dof* in the second part of the vector after the classical ones as proposed by Sukumar et al. [17] or do we keep in order all *dof* (both classical and enriched) for each node? As this choice has no real impact on the final solution, we adopted the later and called this one the "block approach" as each node is stored with his "block" of *dof*. For a `TypeR` element, the elementary *dof* vector is therefore given by

$$u^{e^T} = \{u_{x1},\ u_{y1},\ a_{x1},\ a_{y1},\ \ldots,\ u_{x4},\ u_{y4},\ a_{x4},\ a_{y4}\} \tag{27}$$

where u_{xi}, u_{yi} are the standard *dof* and a_{xi}, a_{yi} are the enriched *dof* for the node i. Starting from the displacement approximation Eq. (2), the shape functions matrix N and the derivatives of the shape functions matrix B, for the `TypeR` element, are given by

$$N = [Ns_1 \quad Nh_1 \quad Ns_2 \quad Nh_2 \quad Ns_3 \quad Nh_3 \quad Ns_4 \quad Nh_4] \tag{28}$$

$$B = [Bs_1 \quad Bh_1 \quad Bs_2 \quad Bh_2 \quad Bs_3 \quad Bh_3 \quad Bs_4 \quad Bh_4] \tag{29}$$

where the block matrices Ns, Nh, Bs and Bh are

$$Bs_i = \begin{bmatrix} \phi_{i,x} & 0 \\ 0 & \phi_{i,y} \\ \phi_{i,y} & \phi_{i,x} \end{bmatrix}; \quad Ns_i = \begin{bmatrix} \phi_i & 0 \\ 0 & \phi_i \end{bmatrix} \tag{30}$$

$$Bh_i = \begin{bmatrix} H\phi_{i,x} & 0 \\ 0 & H\phi_{i,y} \\ H\phi_{i,y} & H\phi_{i,x} \end{bmatrix}; \quad Nh_i = \begin{bmatrix} H\phi_i & 0 \\ 0 & H\phi_i \end{bmatrix} \tag{31}$$

For `TypeX` elements, both N and B matrices contains Nh and Bh blocks only for the enriched nodes. A detailed flowchart of the algorithm for the computing of the elementary stiffness matrix for a `TypeR` element is reported in Fig. 7. This one returns the stiffness matrix resulting from the numerical integration over both positive and negative domains `Surf2D` by a Gaussian quadrature (see Fig. 6). The algorithm for the mass matrix computation is obtained straightforward.

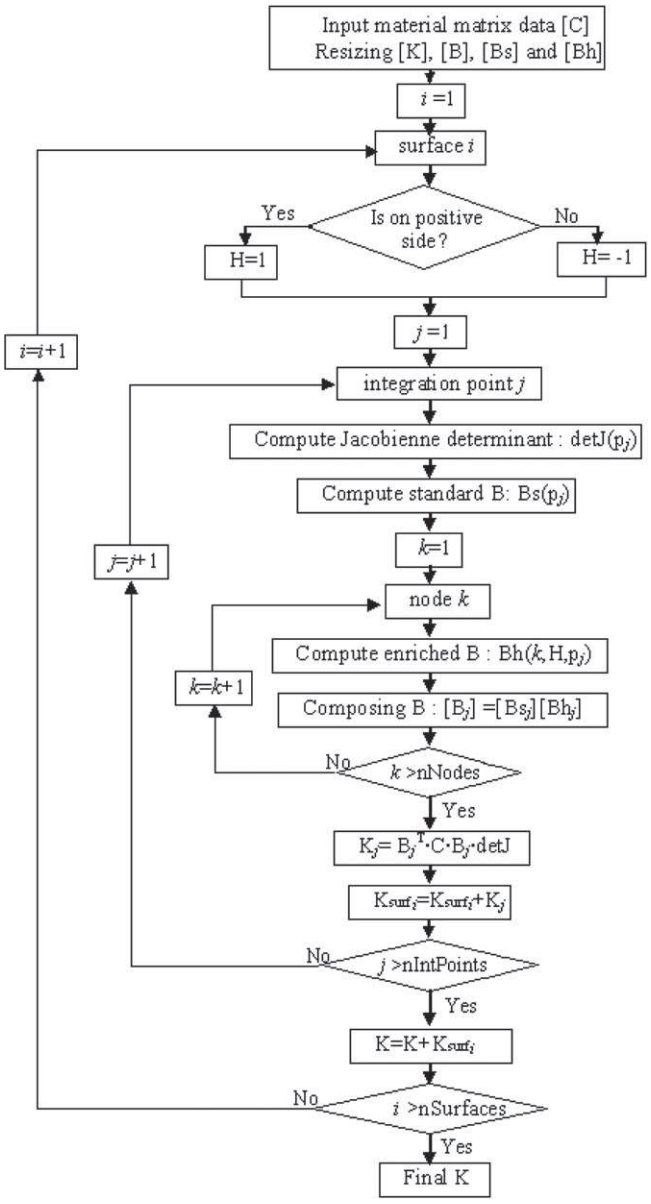

Fig. 7. Flowchart for stiffness matrix computation of a partitioned element.

4.4. Assembly procedure

The global mass matrix, stiffness matrix and external force vector assembling procedure is specific since the elementary corresponding matrices sizes differ from element to element depending on the status. To get round this, the `XAllocation` class has been specifically developed. The main feature of this class is to allow a dynamic mapping between both local and global positions of a *dof* as illustrated in Fig. 8. This bi-directional link is performed by the methods `loc2glob` and `glob2loc`. The former returns the global position of a *dof* depending on the node Id and the local *dof* position while the later gives the reversed mapping. The assembly of the mass matrix, stiffness matrix and external force vector are achieved in a like manner, looping over all elements of the mesh and taking into account the status of each node.

4.5. Implementation of the crack evolution model

Numerical algorithms implemented for the crack evolution model refer mainly to the energy release rate computation and the DSIF extraction presented earlier. The components of the J'-integral are numerically evaluated in DynaCrack considering a symmetrical rectangular path centered at the crack-tip as presented in Fig. 9. As the edges of the path are set parallel to the system axis and the terms inside Eq. (20) are computed with respect to the normal unit vector orientation, the first term in Eq. (20), noted J_k^Γ, is given by

$$J_k^\Gamma = J_k^{AD} + J_k^{DE} + J_k^{EF} + J_k^{FA} + J_k^{B^-C} + J_k^{CB^+} \tag{32}$$

where the six components in the right hand side are computed along the corresponding segments with respect to notations reported in Fig. 9. The first component of the J'-integral (for $k = 1$) is given by

$$J_1^{AD} = \int_A^D \left[-(W+U) + \sigma_{11}\frac{\partial u}{\partial x_1} + \sigma_{12}\frac{\partial v}{\partial x_1}\right] \mathrm{d}x_2 \tag{33}$$

$$J_1^{DE} = \int_D^E \left[\sigma_{22}\frac{\partial v}{\partial x_1} + \sigma_{12}\frac{\partial u}{\partial x_1}\right] \mathrm{d}x_1 \tag{34}$$

$$J_1^{EF} = \int_E^F \left[(W+U) - \sigma_{11}\frac{\partial u}{\partial x_1} - \sigma_{12}\frac{\partial v}{\partial x_1}\right] \mathrm{d}x_2 \tag{35}$$

$$J_1^{FA} = \int_F^A \left[-\sigma_{22}\frac{\partial v}{\partial x_1} - \sigma_{12}\frac{\partial u}{\partial x_1}\right] \mathrm{d}x_1 \tag{36}$$

$$J_1^{B^-C} = \int_{B^-}^C \left[\sigma_{22}\frac{\partial v}{\partial x_1} + \sigma_{12}\frac{\partial u}{\partial x_1}\right] \mathrm{d}x_1 \tag{37}$$

$$J_1^{CB^+} = \int_C^{B^+} \left[-\sigma_{22}\frac{\partial v}{\partial x_1} - \sigma_{12}\frac{\partial u}{\partial x_1}\right] \mathrm{d}x_1 \tag{38}$$

The second component for $k = 2$ are obtained in a straightforward manner. In the above equations, the strain and the kinetic energies are given by

$$W = \frac{1}{2}[\sigma_{11}\varepsilon_{11} + \sigma_{22}\varepsilon_{22} + 2\sigma_{12}\varepsilon_{12}] \tag{39}$$

$$U = \frac{1}{2}\rho[\dot{u}_1\dot{u}_1 + \dot{u}_2\dot{u}_2] \tag{40}$$

The second term, noted J_1^S, of Eq. (20) is given by

$$J_1^S = \int_S \rho\left[\left(\ddot{u}\frac{\partial u}{\partial x_1} + \ddot{v}\frac{\partial v}{\partial x_1}\right) - \left(\dot{u}\frac{\partial \dot{u}}{\partial x_1} + \dot{v}\frac{\partial \dot{v}}{\partial x_1}\right)\right] \mathrm{d}S \tag{41}$$

The computation of this integral is numerically done by evaluating the integrand terms for all Gauss points inside of the considered path. Concerning the elements intersected by the integral path, in light gray in Fig. 9, only theirs integration points inside of the path are considered for the above computation. This is done by the method `computeJ` of the `XExplicitSolver` class dedicated to the explicit time integration and the crack propagation processing. A partitioning algorithm is used to split the path's

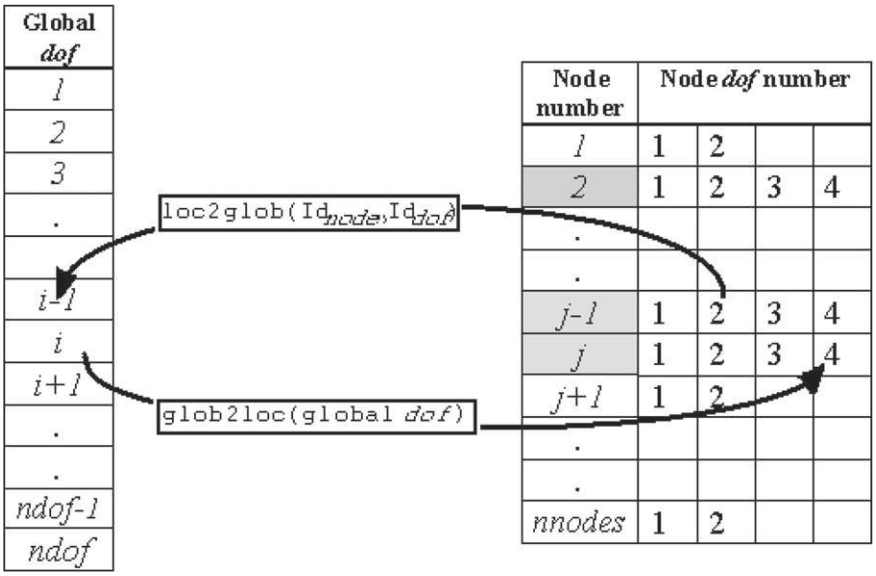

Fig. 8. Allocation procedure illustration.

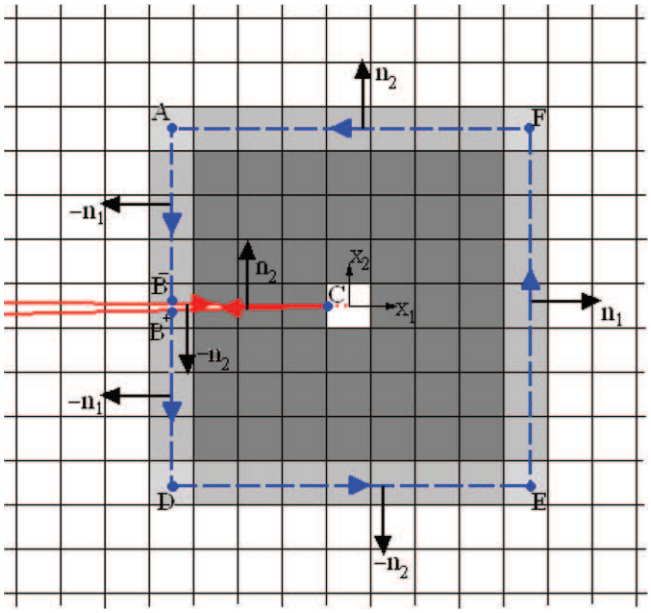

Fig. 9. J-integral rectangular path.

segments into a set of equal sub-segments Δx, and to generate a list of geometric points corresponding to the middle of those sub-segments. The numerical values for stresses, strains, $\frac{\partial u}{\partial x_i}$ and $\frac{\partial v}{\partial x_i}$ are interpolated to the above mentioned points. The numerical integration is based on a classic Gauss quadrature. As observed in Fig. 9, this evaluation is done over a $8L \times 8L$ quadrilateral domain (L being the largest edge of the element containing the crack-tip). Once the J'-integral is computed, the dynamic energy release rate is determined by Eq. (22) and the DSIF components are extracted using Eq. (23). Chessa et al. [30] have shown that the numerical evaluation of the crack-tip opening displacements (δ_n and δ_t), contributing to K_{I} and K_{II}, depends only on the enrichment *dof* by

$$\delta = 2\sum_{j=1}^{4} N_j a_j \tag{42}$$

Some numerical results proving the effectiveness of the numerical evaluation of the J'-integral are presented in Section 5.

5. Numerical examples

In this section, we present some numerical results concerning the propagation of a crack in Mode I for a rectangular plate subjected to an impact load and the analysis of a finite plate with an inclined crack subjected to a mixed mode fracture.

The first problem, illustrated in Fig. 10, was proposed by Lu et al. [31], Krysl et al. [25] and Belytschko et al.

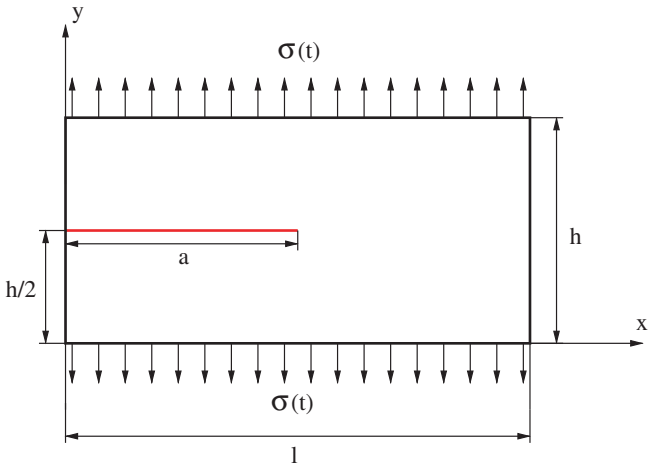

Fig. 10. Model used for the crack propagation.

[32] using EFGM, by Duarte et al. [26] using GFEM, and by Belytschko et al. [5] using X-FEM among many others. Two uniform quadrilateral meshes are presented in this paper to illustrate the independence of the crack propagation with the mesh size: a coarse mesh (51×21 elements), and a finer mesh (65×25 elements). The crack-tip is located at the center of the plate, as shown in Fig. 10, and the crack is horizontal.

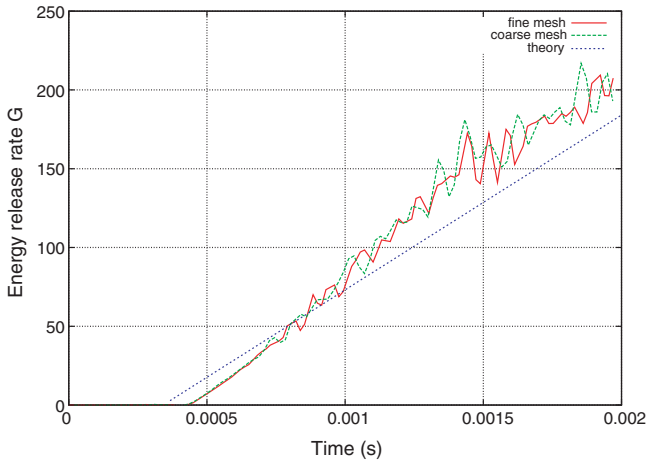

Fig. 11. Energy release rate time-history plot for the stationary crack.

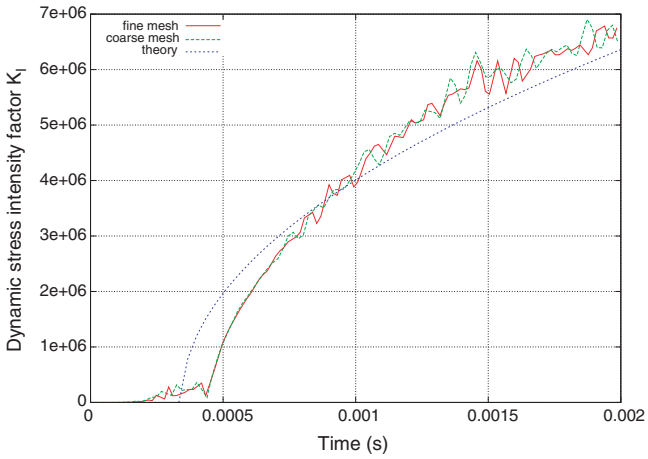

Fig. 12. Mode I DSIF time-history plot for the stationary crack.

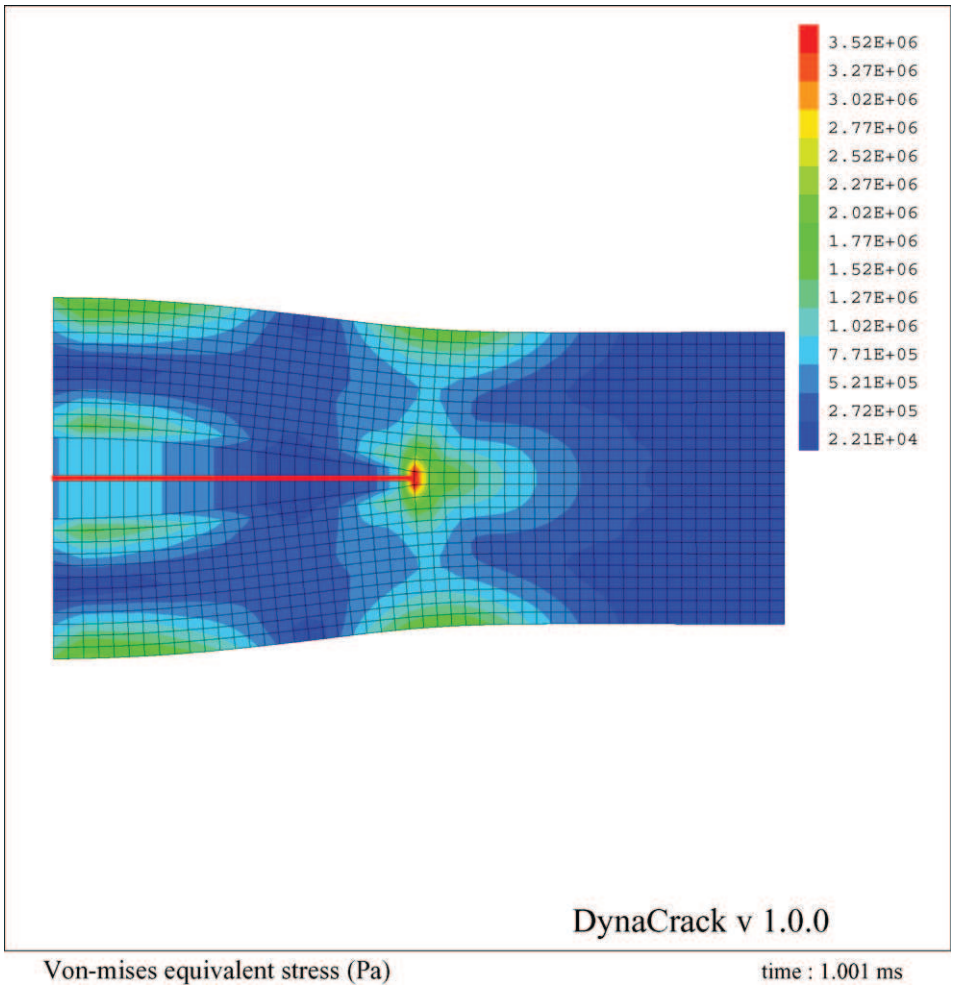

Fig. 13. von Mises stress field for the stationary crack.

A linear elastic material behavior with $E = 200$ GPa, $v = 0.3$ and $\rho = 7833\ \mathrm{kg/m^3}$ is considered and a critical value for fracture toughness $K_{\mathrm{ID_{crit}}} = 1.5\ \mathrm{MPa}\sqrt{\mathrm{m}}$ is used in the dynamic crack evolution model. Only a stationary crack is considered in the first part and the dimensions are $l = 10$ m and $h = 4$m. Only a boundary traction force with a magnitude of 63750 Pa is applied in accordance with the model proposed by Duarte et al. [26]. The computed and the analytical mode-I DSIF and the dynamic energy release rate reported in Figs. 11 and 12. Both figures show a quite good agreement and the non-dependence of the solution with the mesh size. On both figures, one can observe, after $t = 7.5 \times 10^{-4}$ s, quite strong oscillations disturbing the numerical solutions. Actually, as mentioned in [26], for the same test, it occurs because of the finite dimensions of the plate whose boundary reflect the elastic waves. This elastic waves perturbing the displacements, strains and stresses fields used for the J'-integral evaluation. The explicit integration algorithm favour and maintain also these oscillations in absence of plasticity behavior.

Fig. 13 is a contour-plot of the von Mises stresses field at the end of the computation for the coarse mesh. In this later the expected form of the stress field gradient near the crack-tip can be observed. Fig. 14 reports the vertical displacement field contour-plot and shows the jump in the vertical displacement on both sides of the crack. It must be mentioned that because of the finite dimensions of the plate, some elastic waves are reflected by the boundaries when the simulation time increases. Those reflected elastic waves are perturbing the numerical solution and this one is no longer in accordance with the analytic solution.

For the analysis of the crack propagation, the dimensions of the rectangular plate are $l = 0.1$ m and $h = 0.04$ m and the applied traction force magnitude is 1 MPa. The same conditions were adopted by Belytschko et al. [5]. Fig. 15 reports the evolution of the mode I DSIF for both fine and coarse meshes. The crack propagation initiates quite at the same instant for the two considered meshes, $t \simeq 14$ μs for the fine mesh and $t \simeq 15$ μs for the coarse mesh. The crack-tip advances through 8 elements for the fine mesh and through seven elements for the coarse mesh. The exact final crack-tip propagations are $\mathrm{d}x = 6.23 \times 10^{-2}$ m and $\mathrm{d}x = 6.37 \times 10^{-2}$ m for the fine and coarse meshes, respectively. The crack propagation speed is in the range $[1000, 2200]$ m/s. This is below the Rayleigh wave speed. Those results agree quite well with

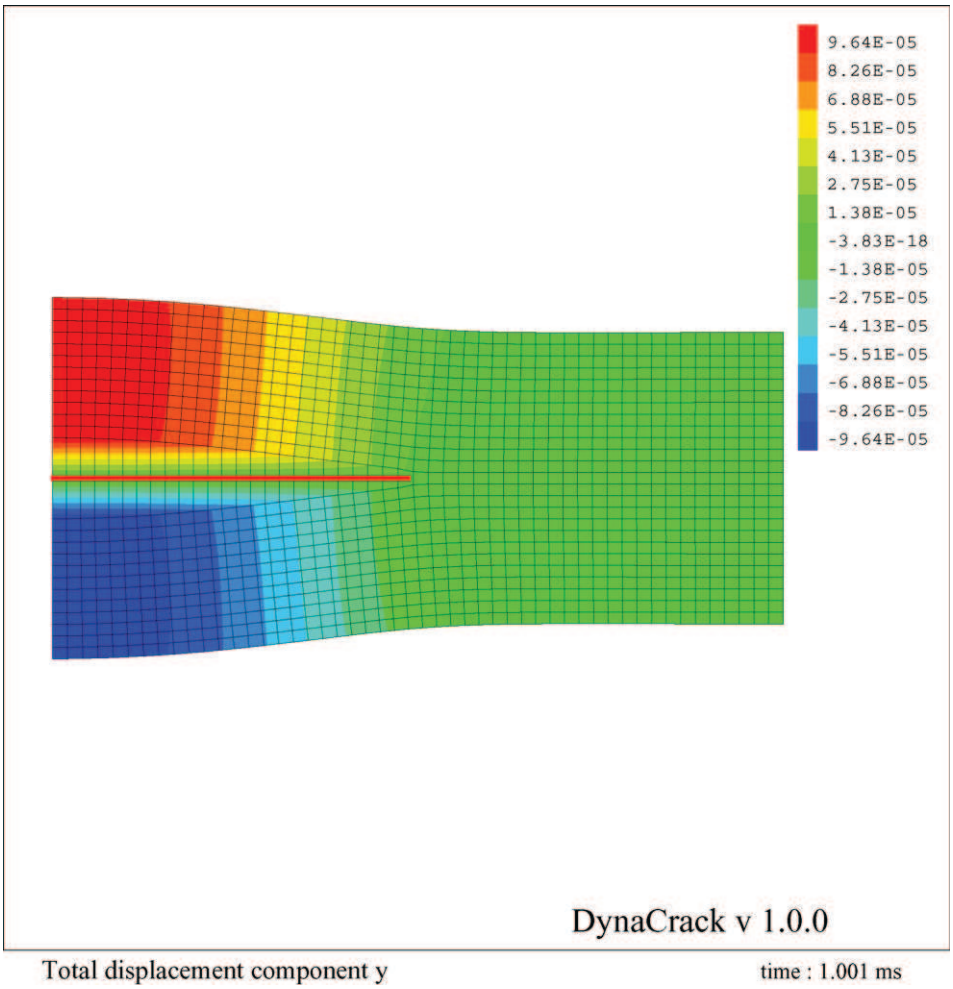

Fig. 14. Vertical displacement component for the stationary crack.

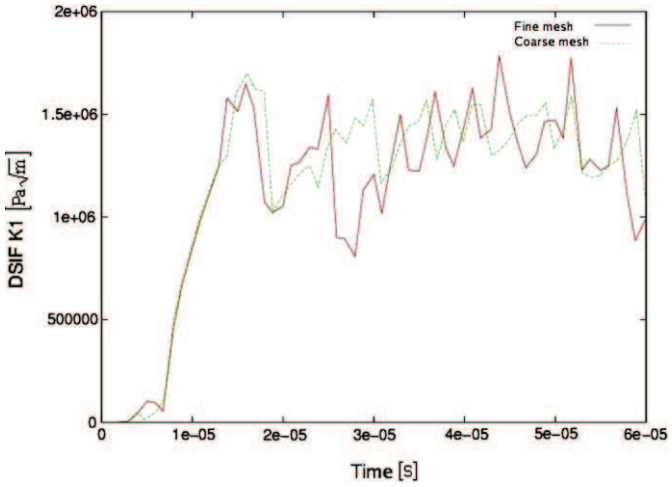

Fig. 15. Mode I DSIF time-history plot for the moving crack.

the one obtained by Belytschko et al. [5] and allow to validate the proposed approach.

The second problem presented in Fig. 16a is the analysis of an inclined crack in a finite plate. The purpose here is to compare the numerical results obtained by DynaCrack for the DSIF in a mixed mode fracture with the results obtained by the Abaqus FEM code for the same simulation. The reference length of the plate is $L = 1$ m and a boundary traction force with a magnitude $\sigma = 1 \times 10^3$ Pa is applied via a step function. The material considered is the same as the one used in the previous example and three different uniform quadrilateral meshes are considered: (i) 20×35, (ii) 25×44 and (iii) 30×53 elements. The equivalent model simulated with the Abaqus software is composed of 911 non-structured elements and the mesh conforms with the crack geometry.

Stress distribution fields given by the X-FEM analysis at the end of the computation time ($t = 0.1$ s) are illustrated in Fig. 16b. Here again, the stress field has the expected shape around the crack-tip. Table 1 reports the numerical values for the fracture parameters (DSIF and dynamic energy release rate) obtained at the end of the analysis by both FEM codes. The results obtained by the path-independent integral technique with DynaCrack agree quite well with the ones obtained using Abaqus. The differences observed for the K_{II} values are related to the inaccurate field representation around the crack-tip because we used a quite simplistic X-FEM enrichment used in this work. More explicitly, these differences are related to the different manner used in Abaqus for computing K_{II} since the second equation for extracting K_I and K_{II} (the first one being the one of the energy release rate) is related to the stress field around crack-tip, obviously more accurately represented in this case.

Table 1
Numerical results obtained for the inclined crack problem

Parameter	Fine mesh	Middle mesh	Coarse mesh	Abaqus
Elements number	1590	1100	700	911
G [J/m^2]	1.18×10^{-3}	1.26×10^{-3}	1.36×10^{-3}	1.23×10^{-3}
K_I [Pa$\sqrt{\text{m}}$]	1521.7	1617.9	1624.2	1480.4
K_{II} [Pa$\sqrt{\text{m}}$]	−404.4	−460.9	−493.8	−722.4

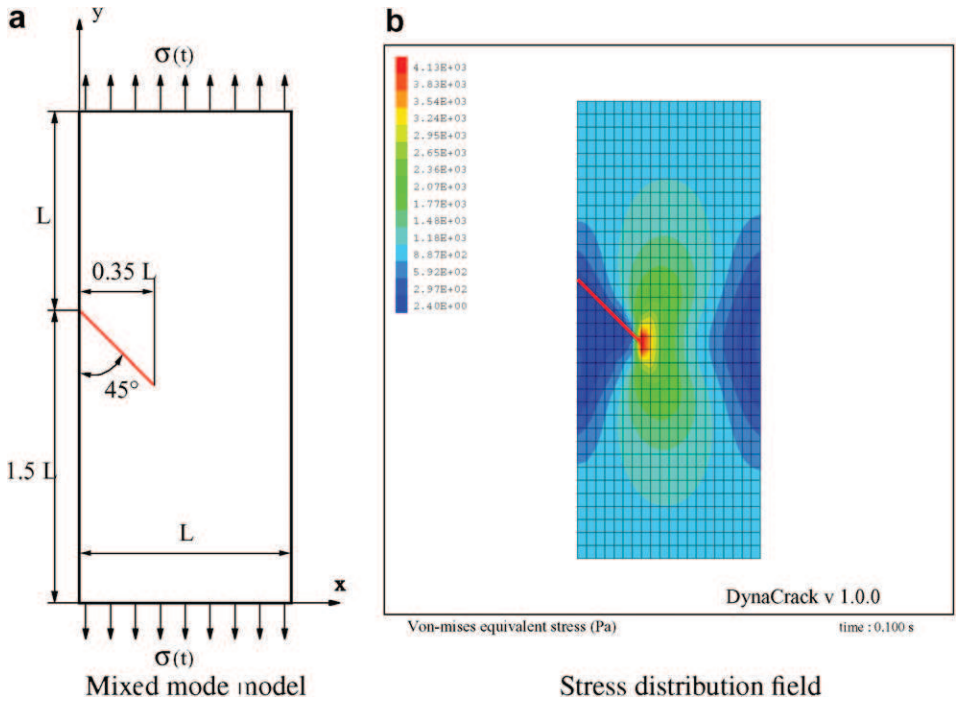

Fig. 16. Inclined crack problem in a finite plate.

6. Conclusions

The development of a dynamic crack propagation module based on the X-FEM approach and its implementation in an explicit finite element code has been presented in this paper. Some specific algorithms have been developed and programmed using the C++ language in order to implement the main features of the X-FEM. The main challenges of this work concerns the treatment of the dynamic nodes enrichments with additional degrees of freedom for the representation of the crack while this one propagates through the structure. The approach used for the numerical integration of the mass and stiffness matrices and the vector force have also been presented with the retained approach to manage of a variable size for the global degree of freedom vector. The implementation of a crack propagation criterion adapted to our specific crack-tip enrichment has also been presented. Finally two numerical examples have also been presented to show the effectiveness of the proposed algorithms.

Concerning the results obtained for these numerical examples, it was clearly pointed out that some supplementary works on our code are needed in order to improve them. First of all a new type of finite element (3-node triangle) will be tested for searching a better conditioning of the elementary matrix. The impact of the Westergaard crack-tip enrichment must be studied also to evaluate the potential gain in accuracy. Several contributions on the dynamic analysis using X-FEM were published since our works began, especially concerning the numerical integration schemes, and next we will try to apply some of such concepts recently developed in order to improve the time integration algorithm.

Future works and further developments of this code concerns the initiation of the crack, i.e. the evolution from a continuous structure to a fissured one, the computation of the crack propagation within a plastic deformable body and the introduction of the contact between the lips of the crack. The most recent developments concern a new formalism for the treatment of the crack based on a mixed extended element free Galerkin (X-EFG) and a FEM model. The main advantage of this approach is to propose an implicit enrichment, i.e. no additional *dof* are used. This may allow to get round of some problems encountered in this work, for example the mass matrix diagonalization.

References

[1] Xu XP, Needleman A. Numerical simulation of fast crack growth in brittle solids. J Mech Phys Solids 1994;42:1397–434.

[2] Camacho GT, Ortiz M. Computational modeling of impact damage in brittle materials. Int J Solids Struct 1996;33:2899–938.

[3] Remmers, JC, Borst, R, Needleman, A. Simulation of fast crack growth using cohesive segments. In: VII international conference on computational plasticity, COMPLAS; 2003.

[4] Babuska I, Melenk JM. The partition of unity method. Int J Numer Meth Eng 1997;40:727–58.

[5] Belytschko T, Chen H, Xu J, Zi G. Dynamic crack propagation based on loss of hyperbolicity and a new discontinuous enrichment. Int J Numer Meth Eng 2003;58:1873–905.

[6] Belytschko T, Fish J, Engelmann BE. A finite element with embedded localisation zones. Comput Meth Appl Mech Eng 1988;70:59–89.

[7] Dvorkin EN. Finite elements with displacements interpolated embedded localization lines insensitive to mesh size and distorsions. Int J Numer Meth Eng 1990;30:541–64.

[8] Simo JC, Oliver J, Armero F. An analysis of strong discontinuities induced by strain softening in rate-independent inelastic solids. Comput Mech 1993;12:277–96.

[9] Jirasek M. Comparative study on finite elements with embedded discontinuities. Comput Meth Appl Mech Eng 2000;188:307–30.

[10] Oliver J, Huespe AE. Theoretical and computational issues in modelling material failure in strong discontinuity scenarios. Comput Meth Appl Mech Eng 2004;193:3195–220.

[11] Belytschko T, Black T. Elastic crack growth in finite elements with minimal remeshing. Int J Numer Meth Eng 1999;45:601–20.

[12] Moes N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. Int J Numer Meth Eng 1999;46:131–50.

[13] Dolbow J, Moes N, Belytschko T. Discontinuous enrichment in finite elements with a partition of unity method. Finite Elem Anal Des 2000;36:235–60.

[14] Réthoré J, Gravouil A, Combescure A. A stable numerical scheme for the finite element simulation of dynamic crack propagation with remeshing. Comput Meth Appl Mech Eng 2004;193:4493–510.

[15] Menouillard T, Réthoré J, Combescure A, Bung H. Efficient explicit time stepping of the extended finite element method (X-FEM). Int J Numer Meth Eng 2006;68(9):911–39.

[16] Pantalé O. An object-oriented programming of an explicit dynamics code: application to impact simulation. Adv Eng Softw 2002;33(5):275–84.

[17] Sukumar N, Prévost JH. Modelling quasi-static crack growth with the extended finite element method. Part I: Computer implementation. Int J Solids Struct 2003;40:7513–37.

[18] Borst R, Remmers JJC, Needleman A. Mesh-independent discrete numerical representations of cohesive-zone models. Eng Fract Mech 2006;73:160–77.

[19] Hulbert GM, Chung J. Explicit time integration for structural dynamics with optimal numerical dissipation. Comput Meth Appl Mech Eng 1996;137:175–88.

[20] Pantalé O. Parallelization of an object-oriented FEM dynamics code: influence of the strategies on the speedup. Adv Eng Softw 2005;36(6):361–73.

[21] Dolbow JE, Devan A. Enrichment of enhanced assumed strain approximations for representing strong discontinuities: addressing volumetric incompressibility and the discontinuous patch test. Int J Numer Meth Eng 2004;59:47–67.

[22] Ventura G. On the elimination of quadrature subcells for discontinuous functions in the extended finite-element method. Int J Numer Meth Eng 2006;66:761–95.

[23] Erdogan F, Sih G. On the crack extention in plates under plane loading and traverse shear. J Basic Eng 1963;85:519–27.

[24] Freund LB. Dynamic fracture mechanics. Cambridge University Press; 1998.

[25] Krysl P, Belytschko T. The element free Galerkin method for dynamic propagation of arbitrary 3-D cracks. Int J Nume Meth Eng 1999;44:767–800.

[26] Duarte CA, Hamzeh ON, Liszka TJ, Tworzydlo WW. A generalized finite element method for the simulation of three-dimensional dynamic crack propagation. Comput Meth Appl Mech Eng 2001;190:2227–62.

[27] Nishioka T. Computational aspects of dynamic fracture in comprehensive structural integrity. Elsevier; 2003.

[28] Nishioka T. Computational dynamic fracture mechanics. Int J Frac 1997;86:127–59.

[29] Nishioka T, Atluri SN. On the computation of mixed-mode k-factors for a dynamically propagating crack, using path-independent integrals *j*'. Eng Frac Mech 1984;20:193–208.
[30] Chessa C, Belytschko T. A local space-time discontinuous finite element method. Comput Meth Appl Mech Eng 2006;195:1325–43.
[31] Lu YY, Belytschko T, Tabbara M. Element-free Galerkin method for wave propagation and dynamic fracture. Comput Meth Appl Mech Eng 1995;126:131–53.
[32] Belytschko T, Tabbara M. Dynamic fracture using element-free Galerkin methods. Int J Numer Meth Eng 1996;39:923–38.