Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 398 (2022) 115015
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

An automated mesh generation algorithm for simulating complex crack growth problems

Mohamad Mohmadsalehi[a], Soheil Soghrati[a,b,*]

[a] Department of Mechanical and Aerospace Engineering, The Ohio State University, USA
[b] Department of Materials Science and Engineering, The Ohio State University, USA

Received 13 January 2022; received in revised form 10 April 2022; accepted 10 April 2022
Available online 20 June 2022

Abstract

In this manuscript, we expand the conforming to interface structured adaptive mesh refinement (CISAMR) algorithm for modeling complex two-dimensional (2D) crack growth problems involving contact/friction along the crack surface and interaction between multiple cracks. The CISAMR algorithm transforms a structured mesh into a high-quality conforming mesh non-iteratively, which is an attractive feature for modeling the evolution of the crack geometry with minimal changes to the underlying mesh structure. To model such problems, the mesh structure is first adaptively refined and updated near the crack tip to form a spider-web pattern of elements for the accurate approximation of the energy release rate and thereby predicting the new crack path. In each step of the crack advance simulation, a small subset of elements in the vicinity of the crack tip is detected using a tree data structure and then deleted/regenerated to simulate the crack growth. The construction of a high-quality mesh with appropriate element aspect ratios in the algorithm allows the use of an explicit dynamic solver, which is essential to simulate the nonlinear response of the problem caused by contact forces along crack faces. Several benchmark fracture problems are presented to study the accuracy of the proposed algorithm, as well as two more complex problems to demonstrate its ability for modeling interaction of multiple growing cracks with one another and with embedded heterogeneities in the domain.

Keywords: CISAMR; Mesh generation; Crack growth; Finite element method; Multi-crack

1. Introduction

The reliability assessment and predicting the remaining service life of structural components/materials with embedded cracks is an important step towards their computational design. Many problems, such as predicting the fatigue life of a structure, require simulating the growth of these cracks, which is often set up as a linear elastic fracture problem at each load step. The finite element method (FEM) is one of the most versatile numerical techniques for modeling problems with both weak and strong discontinuities. However, the main challenge towards simulating crack growth problems using FEM is the spatial discretization of the domain (mesh generation), which

* Correspondence to: Departments of Mechanical and Aerospace Engineering & Materials Science and Engineering, The Ohio State University, Columbus, OH, USA.
E-mail address: soghrati.1@osu.edu (S. Soghrati).

https://doi.org/10.1016/j.cma.2022.115015
0045-7825/

must evolve at each load step to adapt to the changing topology of cracks [1]. To overcome this challenge, several numerical techniques are introduced to either facilitate the meshing process in the context of FEM or eliminate the need for generating conforming meshes throughout the simulation process. In the next few paragraphs, we briefly visit some of these algorithms.

A well-known strategy for modeling fracture problems without updating the mesh is the nodal release approach [2,3], which is a popular algorithm due to its relatively straightforward implementation. In this method, the crack is allowed to only propagate along element edges and thereby no remeshing is required, although adaptive refinement is often employed near crack tips to improve the accuracy of predicting stress intensity factors. However, simulation results are mesh dependent and the growing crack trajectory could considerably change with the refinement level and element type [4]. Note that forcing the crack trajectory to conform to the underlying mesh structure could lead to artificial local stress concentrations that reduce the accuracy of calculations required for determining the onset and angle of crack propagation [5].

The element free Galerkin (EFG) method [6] is another numerical technique that has shown promising outcomes for modeling progressive crack growth problems. This mesh free method does not require any element connectivity data and even in the case of irregular nodal arrangements discretizing the domain can maintain a good accuracy [7]. Moreover, it does not show any volumetric locking and its rate of convergence can exceed conventional FEM, while maintaining the continuity of both the field and its gradients in the entire domain [8]. On the other hand, one of the main limitations of the EFG method is the difficulty associated with imposing Dirichlet boundary conditions, which requires the use of Lagrange multipliers [9]. It should be noted that applying Lagrange multipliers for dynamic simulations is a major implementation issue. Numerical techniques such as the Reproducing Kernel Particle Method (RKPM) [10] and the EFG-Particle method [11] have been introduced to improve the accuracy or address some of the implementation issues associated with the EFG method.

The phase field method has also widely been used for simulating problems involving the initiation and propagation of damage, although the cracks are not explicitly represented in this approach [12,13]. Instead, a continuous field parameter (phase field variable) is employed to differentiate between fully damaged and intact material phases, with a sharp (but continuous) transition between them indicating the crack location. The evolution of crack geometry is then simulated by minimizing an energy functional in terms of the displacement field and internal fracture surface variable [14]. The most attractive feature of the phase field method is its ability to simulate crack initiation, propagation, merging and branching automatically, without any ad-hoc fracture criteria, on a fixed mesh [15]. Among drawbacks of this method, we can mention its high computational cost and parameter sensitivity of results [16], where accuracy is dependent on the solver type, length scale, mesh orientation, mesh refinement in transition zone, and conditioning coefficient [17].

Similarly, in the smeared crack approach [18], rather than an explicit representation of the crack geometry (strong discontinuity), it is modeled by modifying material constitutive relation of elements. While this method is easy to implement and does not need the remeshing process for each simulation step, the results are mesh sensitive [19] and suffer from spurious mesh bias [20]. Moreover, although the displacement field can be approximated accurately, the higher error associated with recovery of gradients near the crack tip undermines the fidelity of this method in predicting the crack path [21]. To reduce the influence of the mesh orientation, the constitutive models for the regularization of ill-posed strain softening problems such as nonlocal model [22] and gradient-enhanced approach [23] can be employed. However, these methods require a refined mesh in the localization zones to assure mesh objectivity. Therefore, for large scale problems, they are computationally expensive.

The eXtended/Generalized FEM (X/GFEM) [24–26] is one of the most successful methods for modeling fracture problems, which unlike standard FEM, does not require the construction of a conforming mesh for approximating the field [27]. This algorithm uses additional (generalized) degrees of freedom at the nodes of elements intersecting with the crack, as well as their neighboring (blending) elements and interpolates appropriate enrichment functions using the partition of unity method (PUM) at these nodes to reconstruct the discontinuous field [28]. Special enrichment functions are often used in the vicinity of a crack tip or at a branching point to accurately simulate the stress singularity, which can even yield a higher convergence rate than standard FEM [29,30]. Using X/GFEM, the background elements cut by the crack must still be subdivided into conforming sub-elements (children elements) to allow accurate numerical quadrature [31]. Additional considerations ($e.g.$, inverse mapping of quadrature points) are needed during the numerical integration, although the added computational cost is justified by the rise in the convergence rate [32]. Various treatment strategies are introduced in the literature to address other

challenges associated with the implementation of X/GFEM, such as enforcing Dirichlet boundary conditions and ill-conditioning of the stiffness matrix [24,33] Among other enriched methods used for modeling fracture problems we can mention the discontinuity enriched FEM (DEFEM) [34], which extends the Interface-enriched generalized FEM (IGFEM) [35,36] for approximating crack growth problems.

In the context of standard FEM approximation of fracture problems, the greatest challenge is reconstructing a high-quality mesh at each step of the simulation to accurately approximate the field and predict crack path in the next step [37]. Several algorithms have been introduced for remeshing the domain in such problems, among which we can mention the nodal relaxation and remeshing approach in [38,39], moving meshes relying on the Delaunay triangulation scheme [40], combining quad tree and advancing front technique [41], the Lepp–Delaunay based technique [42], adaptive Delaunay triangulation [43], adaptive mesh refinement algorithms relying on the super convergent patch recovery [44], and iterative h/r-adaptive schemes [45,45], edge rotation technique for efficient local refinement [46,47], and the universal mesh method relying on an iterative relaxation algorithm [48,49]. The overall remeshing procedure involves deleting a set of elements in the vicinity of the crack tip, evaluating/extending the crack tip to its new location, and finally creating new elements in this region to reconstruct the mesh [50]. It is critical that the remeshing algorithm is capable of generating high-quality elements at every step of the simulation, as the accuracy of stress intensity factors evaluated to predict the crack tip location at the following step is highly dependent on the mesh quality. Moreover, maintaining an acceptable computational efficiency while minimizing error requires generating much finer elements near the crack tip to accurately recover sharp gradients in this region [51].

There are three major criteria used for determining the crack growth orientation/length in an FE-based simulation: (i) energy-based methods; (ii) stress-based methods; and (iii) displacement-based methods. According to the minimum strain energy density criterion [52], the failure (crack growth) occurs when the strain energy density near the crack tip reaches a critical value. The J-integral [53] is a path-independent contour integral for evaluating the strain energy release rate near the crack front, although accurate calculation of this integral in FE implementation could be challenging [54]. By applying the divergence theorem, the J-integral can be converted into an area (2D) or volume (3D) integral [55], also known as interaction integral or M-integral, which not only facilitates its implementation over unstructured meshes but can also significantly increase the accuracy [56,57]. This approach relies on conservation integrals for two states of the material (actual and auxiliary) which has also been implemented in other numerical techniques such as BEM, XFEM, and EFG [58]. The virtual crack closure technique (VCCT) [59] is another approach for the evaluation of stress intensity factor in mixed-mode fracture problems, which is based on the assumption that released energy in the crack growth process is equal to the energy needed to close the crack to its original state. Another criterion originally introduced in [60] assumes when the maximum circumferential stress near the crack tip reaches a critical value, the crack propagates along the plane that is normal to the maximum hoop stress. Alternatively, the displacement extrapolation technique [61] utilizes relative opening and sliding displacements along opposing sides of the crack to approximate the stress intensity factor at the crack tip. In an FE simulation, this method yields considerably more accurate predictions when implemented together with quarter-point singular elements at the crack tip [62].

In this manuscript, we present an automated meshing/remeshing approach by expanding the Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) algorithm [63,64] for modeling crack growth problems. CISAMR is a non-iterative algorithm especially suited for meshing problems with complex geometries. The algorithm proposed here for modeling crack growth problems merely remeshes a small portion of the initial mesh in the vicinity of the crack tip, while the remainder of the mesh remains intact. Besides facilitating the implementation, this feature can considerably reduce the interpolation errors in non-linear fracture problems (not addressed in this work) by eliminating the need for element-to-element mapping to transfer the field and its gradients between old and new meshes. Further, the remeshing algorithm uses adaptive refinement and a spider-web pattern of elements at the crack tip to facilitate the use of interaction integral for the accurate approximation of stress intensity factors and subsequently the crack path. The geometric engine developed for CISAMR can automatically detect cracks that are about to intersect and the meshing algorithm readily accommodates merging them. Similarly, intersection between cracks and embedded heterogeneities can easily be handled using CISAMR. We have also integrated CIASMR with ABAQUS to use the explicit solver and ability to model contact/friction in this software to facilitate to simulate crack closure (contact forces) in mixed mode fracture problems and in particular after crack merging. It must be noted that CISAMR meshes are generated by converting an initial structured mesh into a conforming mesh. Therefore,

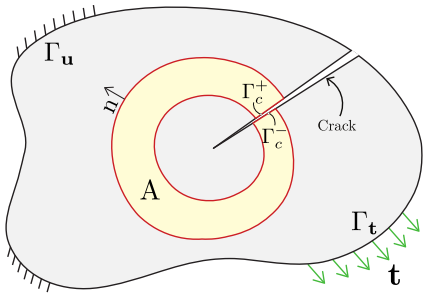


Fig. 1. An open domain with a pre-existing crack, which also shows a sub-domain that can be used for evaluating the M-integral.

the algorithm presented here is only capable of generating meshes composed of 4-node quadrilateral and 3-node triangular elements. It is worth mentioning that CISAMR has previously been extended to higher-order elements for interface problems [65]; thus expansion of the current remeshing algorithm to higher order for fracture problems would be feasible.

In the remainder of the article, we first present strong and weak forms of governing equations for modeling linear elastic fracture mechanics problems in Section 2, together with the equations used for the calculation of the interaction integral. After a brief review of original CISAMR algorithm in Section 3, the new features added to this algorithm for the automated modeling of crack growth problems are presented in Section 4. Several numerical examples are presented in Section 5 to verify the accuracy and demonstrate the versatility of CISAMR algorithm for modeling various fracture problems. Final concluding remarks are provided in Section 6.

2. Governing equations

2.1. Linear elastic fracture

Although in some examples of this manuscript we study multi-crack problems, without the loss of generality and to simplify the formulations that follow, we present the governing equations for a domain with a single crack. Consider the open domain $\Omega \subset \mathbb{R}^2$ shown in Fig. 1 with an outward unit normal vector $\mathbf{n}$ and the boundary $\partial\Omega$ subdivided into regions with applied displacement ($\Gamma_{\mathbf{u}}$) and traction ($\Gamma_{\mathbf{t}}$) boundary condition. The strong form of linear elasticity governing equations in this domain can be expressed as

$$\begin{cases} \nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} & \text{in } \Omega \\ \mathbf{u} = \bar{\mathbf{u}} & \text{on } \Gamma_{\mathbf{u}} \\ \boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} & \text{on } \Gamma_{\mathbf{t}}, \end{cases} \tag{1}$$

Where $\mathbf{u}$, $\mathbf{b}$, $\boldsymbol{\sigma}$, and $\mathbb{C}$ are the displacement field, body force, Cauchy stress tensor, and the 4th-order elasticity tensor, respectively. Also, $\bar{\mathbf{u}}$ and $\bar{\mathbf{t}}$ denote prescribed displacement and traction boundary conditions along $\Gamma_{\mathbf{u}}$ and $\Gamma_{\mathbf{t}}$, respectively, and $\nabla = \langle \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \rangle$. The infinitesimal strain tensor $\boldsymbol{\varepsilon}$ and Cauchy stress tensor $\boldsymbol{\sigma}$ are evaluated as

$$\begin{cases} \boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right), \\ \boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}. \end{cases} \tag{2}$$

To derive the weak (variational) form of (1) for an open domain with an embedded crack, consider the trial function $\mathbf{u}(\mathbf{x}) \in H^1(\Omega)$ and test function $\delta\mathbf{v}(\mathbf{x}) \in H^1(\Omega)$, where H^1 is a Hilbertian Sobolev space of order 1. The weak form of (1) can be written as

$$\int_{\Omega} \delta\mathbf{v} : \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_{\Omega} \delta\mathbf{v} \cdot \mathbf{b} \, \mathrm{d}\Omega - \int_{\Gamma_{\mathbf{t}}} \delta\mathbf{v} \cdot \bar{\mathbf{t}} \, \mathrm{d}\Gamma_t = \mathbf{0} \quad \forall \, \delta\mathbf{v} \in H^1(\Omega). \tag{3}$$

Using the Galerkin method and utilizing Lagrangian shape functions, (3) can be approximated by discretizing the domain into m finite elements that conform to the crack geometry such that $\Omega \approx \cup_{i=1}^{m} \bar{\Omega}_i$, where $\bar{\Omega}_i$ is the area of each element. Note that the accuracy of the FE approximation is highly dependent on the quality of the mesh (elements aspect ratio and size), specially near the crack tips to minimize the discretization and interpolation errors.

2.2. Interaction integral and crack path

The energy release rate associated with a crack tip can be evaluated using the J-integral, which can be written as

$$J(s) = \lim_{\Gamma_s \to 0} \int_{\Gamma_s} (W\delta_{1i} - \sigma_{ij} u_{j,1}) n_i \mathrm{d}\Gamma, \tag{4}$$

where Γ_s is a closed contour surrounding the crack tip (cf. Fig. 1), u_j is the displacement vector, and W is the strain energy density given by

$$W = \frac{1}{2}\sigma_{ij}\varepsilon_{ij} = \frac{1}{2}\mathbb{C}_{ijkl}\varepsilon_{kl}\varepsilon_{ij}. \tag{5}$$

Using the divergence theorem with a weighting function q, the domain integral equivalent to (4) can be written as

$$J(s) = \int_A (\sigma_{ij} u_{j,1} - W\delta_{1i}) q_{,i} \mathrm{d}A + \int_A (\sigma_{ij} u_{j,1} - W\delta_{1i})_{,i} q \mathrm{d}A - \int_{\Gamma_c^+ + \Gamma_c^-} t_j u_{j,1} q \mathrm{d}s, \tag{6}$$

where t_j is the traction vector along on the crack surface. By superposing William's solution [58] for the displacement field in the vicinity of the crack (auxiliary field) on the actual field, The combined J-integral, J^S, is given by

$$\begin{aligned} J^{\mathrm{S}}(s) = &\int_A \{(\sigma_{ij} + \sigma_{ij}^{\mathrm{aux}})(u_{j,1} + u_{j,1}^{\mathrm{aux}}) - W^{\mathrm{S}}\delta_{1i}\} q_{,i} \mathrm{d}A + \int_A \{(\sigma_{ij} + \sigma_{ij}^{\mathrm{aux}})(u_{j,1} + u_{j,1}^{\mathrm{aux}}) - W^{\mathrm{S}}\delta_{1i}\}_{,i} q \mathrm{d}A \\ &- \int_{\Gamma_c^+ + \Gamma_c^-} (t_j + t_j^{\mathrm{aux}})(u_{j,1} + u_{j,1}^{\mathrm{aux}}) q \mathrm{d}s, \end{aligned} \tag{7}$$

where superscripts "S" and "aux" denote superimposed and auxiliary states, respectively. For linear elastic problems, the strain energy density of the superimposed state, W^S, is expressed as

$$W^S = W + W^{\mathrm{aux}} + W^I = \frac{1}{2}(\sigma_{ij} + \sigma_{ij}^{\mathrm{aux}})(\varepsilon_{ij} + \varepsilon_{ij}^{\mathrm{aux}}), \tag{8}$$

where W^I is evaluated as

$$W^I = \frac{1}{2}(\sigma_{ij}\varepsilon_{ij}^{\mathrm{aux}})(\sigma_{ij}^{\mathrm{aux}}\varepsilon_{ij}). \tag{9}$$

Similarly, the superimposed J-integral in (7) can be decomposed into three terms as

$$J^{\mathrm{S}}(s) = J(s) + J^{\mathrm{aux}}(s) + I(s), \tag{10}$$

where J^{aux} and I are auxiliary and interaction domain integrals, respectively. Comparing (7) and (10), the interaction integral (also known as M-integral) can then be expressed as

$$\begin{aligned} I = &\int_A \{\sigma_{ij} u_{j,1}^{\mathrm{aux}} + \sigma_{ij}^{\mathrm{aux}} u_{j,1} - \frac{1}{2}(\sigma_{jk}\epsilon_{jk}^{\mathrm{aux}} + \sigma_{jk}^{\mathrm{aux}}\epsilon_{jk})\delta_{1i}\} q_{,i} \mathrm{d}A \\ &+ \int_A \{\sigma_{ij} u_{j,1}^{\mathrm{aux}} + \sigma_{ij}^{\mathrm{aux}} u_{j,1} - \frac{1}{2}(\sigma_{jk}\epsilon_{jk}^{\mathrm{aux}} + \sigma_{jk}^{\mathrm{aux}}\epsilon_{jk})\delta_{1i}\}_{,i} q \mathrm{d}A \\ &- \int_{\Gamma_c^+ + \Gamma_c^-} (t_j u_{j,1}^{\mathrm{aux}} + t_j^{\mathrm{aux}} u_{j,1}) q d\Gamma. \end{aligned} \tag{11}$$

Assuming the same elasticity tensor for both states, $\mathbb{C}_{ijkl}$ couples the auxiliary and actual stress and strain components. Also, using the equilibrium of actual stresses, (11) can be simplified as

$$\begin{aligned} I = &\int_A \{\sigma_{ij} u_{j,1}^{\mathrm{aux}} + \sigma_{ij}^{\mathrm{aux}} u_{j,1} - \sigma_{jk}\epsilon_{jk}^{\mathrm{aux}}\delta_{1i}\} q_{,i} \mathrm{d}A + \int_A \{\sigma_{ij}(u_{j,1i}^{\mathrm{aux}} - \varepsilon_{ij,1}^{\mathrm{aux}}) + \sigma_{ij,1}^{\mathrm{aux}} u_{j,1}\} q \mathrm{d}A \\ &- \int_{\Gamma_c^+ + \Gamma_c^-} t_j u_{j,1}^{\mathrm{aux}} q d\Gamma. \end{aligned} \tag{12}$$

After discretizing the domain in the FE approximation, (12) can be evaluated as

$$I = \sum_{e=1}^{e_A}\sum_{p=1}^{p_e}\{(\sigma_{ij}u^{\text{aux}}_{j,1} + \sigma^{\text{aux}}_{ij}u_{j,1} - \sigma_{jk}\varepsilon^{\text{aux}}_{jk}\delta_{1i})q_{,i}\}_p|\mathbf{J}|_p w_p + \sum_{e=1}^{e_A}\sum_{p=1}^{p_e}\{[\sigma_{ij}(u^{\text{aux}}_{j,1i} - \varepsilon^{\text{aux}}_{ij,1}) + \sigma^{\text{aux}}_{ij,1}u_{j,1}]q\}_p|\mathbf{J}|_p w_p - \sum_{f=1}^{e_f}\sum_{e=1}^{e_A}(t_j u^{\text{aux}}_{j,1}q)_p|\mathbf{J}|_p w_p, \tag{13}$$

where e_A and e_f are the number of elements in the contour integral and number of element faces, respectively, and p_e is number of integration points per element. Also, w_p is the weight factor at integration point p and $|\mathbf{J}|_p$ is the determinant of the element Jacobian matrix. Note that q is an arbitrary function in the contour integral domain that for plain stress/strain problems must satisfy the requirement of being $q = 1$ along the inner ring and $q = 0$ along the outer boundary of this contour. In [66], it has been shown that the integral value is insensitive to the function chosen for q, although the shape of the integration domain (including or excluding the crack tip) affects the accuracy. In Section 4, we discuss how the integration domain and q function are selected in the CISAMR implementation for simulation crack growth problems.

The relationship between the interaction integral and SIFs (K_I and K_{II}) is given by

$$I = \frac{2}{E'_{\text{tip}}}(K_I K^{\text{aux}}_I + K_{II}K^{\text{aux}}_{II}). \tag{14}$$

For plain stress problems, E'_{tip} is elastic modulus at the crack tip while for plain strain problems it is evaluated as and $E'_{\text{tip}} = \frac{E_{\text{tip}}}{1-\nu^2_{\text{tip}}}$ (ν_{tip}: Poisson's ratio at the crack tip). One could calculate the SIF for each fracture mode by setting $K^{\text{aux}}_I = 1$ and $K^{\text{aux}}_{II} = 0$ for mode I and $K^{\text{aux}}_{II} = 1$ and $K^{\text{aux}}_I = 0$ for mode II. The crack propagation angle, θ_p, can then be evaluated using maximum circumferential stress criterion as [67]

$$\theta_p = 2 \times \arctan\frac{1}{4} \times \left(\frac{K_I}{K_{II}} \pm \sqrt{\left(\frac{K_I}{K_{II}}\right)^2 + 8}\right). \tag{15}$$

2.3. Contact and friction along the crack face

Under mixed mode fracture loading and in particular after the merging of two nearby cracks, contact and relative sliding could occur along crack faces. The frictional contact is often applied in an FE model by constraining the nodes on one surface (labeled slave) to prohibit its penetration to the other surface (master). Using a balanced master–slave contact is crucial to minimizing the penetration of contacting bodies, as well as the accurate approximation of contact forces along crack faces. Coupling the software package ABAQUS with CISAMR for simulating crack growth problems, the general contact model [68] in this software was implemented to apply this constraint. We used the Coulomb friction model in conjunction with the finite sliding formulation and surface-to-surface contact model, together with the penalty method to approximate the pressure overclosure behavior. More details regarding how master/slave nodes are specified during the CISAMR meshing process and fed to ABAQUS solver are provided in Section 4.1.

3. CISAMR algorithm: Overview

Before discerning the modified CISAMR algorithm for modeling crack growth problems, it is worthwhile to review the original CISAMR algorithm for meshing 2D interface problems, as presented in [63]. This algorithm automatically transforms a structured background mesh composed of 4-node quadrilateral elements (Q4) into a high-quality conforming hybrid mesh composed of 3-node triangle (T3) and Q4 elements. A unique feature of CISAMR is that this transformation is carried out non-iteratively while ensuring resulting elements aspect ratios do not exceed three via h-adaptivity, r-adaptivity, and sub-triangulation of background elements. This process is schematically shown in Fig. 2 and each step of the algorithm is described in more detail below.

∘ ***h*-adaptive refinement:** To achieve the desired level of refinement along the interface (here, the crack body) and minimize the geometric discretization error, we implement a customized Structured Adaptive Mesh Refinement

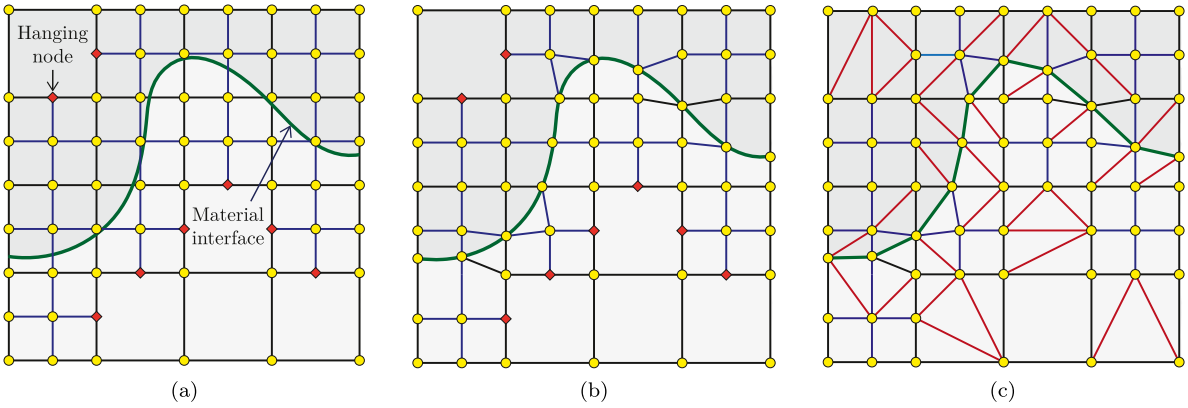


Fig. 2. Transforming a background mesh into a conforming mesh using CISAMR: (a) h-adaptive refinement of background elements; (b) r-adaptivity of elements cut by the interface; (c) sub-triangulating of elements with a node on the interface or with hanging nodes.

(SAMR) algorithm (cf. Fig. 2(a)). At each level of refinement, all elements cut by the interface/crack and their selected neighboring elements are subdivided into four sub-quadrangles. The neighboring elements selected for refinement must have at least one node whose distance with the intersection of crack/interface and the edges connected to that node is less than half of the element size. As discussed in Section 4, to accurately approximate SIFs and the crack kinking angle in fracture problems, additional level of refinement is applied to the background mesh near the crack tip grid along the whole crack body, near crack tip regions are refined further locally.

∘ ***r*-adaptivity:** A non-iterative r-adaptivity algorithm is then employed to relocate selected nodes of the adaptively refined background elements intersecting the interface/crack. These nodes are relocated to the interface/crack in the direction of element edges. Assume the length of element edges connected to node $\mathcal{N}$ are h_i and intersection points of these edges with the interface/crack are at distance(s) d_i form $\mathcal{N}$. The following algorithm is used to determine if and how this node is relocated (cf. Fig. 3(a)):

1. If only one of the edges connected to $\mathcal{N}$ intersects the interface/crack:
   (a) If $d \geq 0.5h$: no need to relocate the node.
   (b) If $d < 0.5h$: move the node to the edge–interface intersection point.
2. If two of the edges connected to $\mathcal{N}$ intersect the interface/crack:
   (a) If $d_1 \geq 0.5h_1$ and $d_2 \geq 0.5h_2$: do not relocate the node.
   (b) If $d_1 < 0.5h_1$ and $d_2 \geq 0.5h_2$: move the node to the closer intersection point at distance d_1 and ignore the second intersection point at distance d_2.
   (c) If $d_1 < 0.5h_1$ and $d_2 < 0.5h_2$ and $d_1 < d_2$: move the node to the closer intersection point at distance d_1 and delete the second intersection point at distance d_2.

Note that by imposing certain constraints on the mesh size during the SAMR phase (see [63] for details), no case scenario other than those outlined above could occur during the r-adaptivity phase for smooth interfaces. For interfaces with sharp corners (slope discontinuity), which also frequently occurs due to the initial crack kinking in crack growth problems, the r-adaptivity is carried out similarly to that discussed above with one additional step after the completion of this process. As shown in Fig. 3(b), after performing the regular r-adaptivity, the background element node with the closest distance to the sharp corner is snapped to this point to prepare the mesh for the next step (sub-triangulation). Similarly, two intersecting interfaces are handled using a hierarchical r-adaptivity algorithm in which regular r-adaptivity is carried out for each interface independently in a given order, followed by snapping the closest background element node to their intersection point.

∘ **Sub-triangulation:** Finally, all the elements intersecting the interface/crack or deformed during the r-adaptivity phase, and also elements with hanging nodes generated due to h-adaptive refinement process are subdivided into T3 elements to generate the final conforming mesh (cf. Fig. 2(c)). The sub-triangulation phase must also be extended

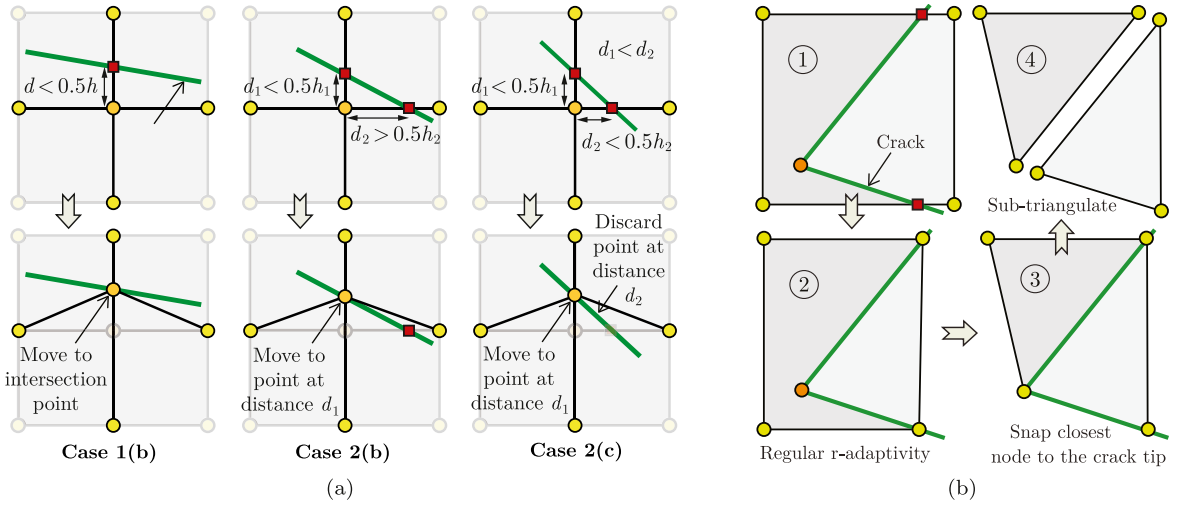


Fig. 3. (a) Three case scenarios for relocating background element nodes during the r-adaptivity process; (b) hierarchical r-adaptivity for handling a sharp corner of the interface.

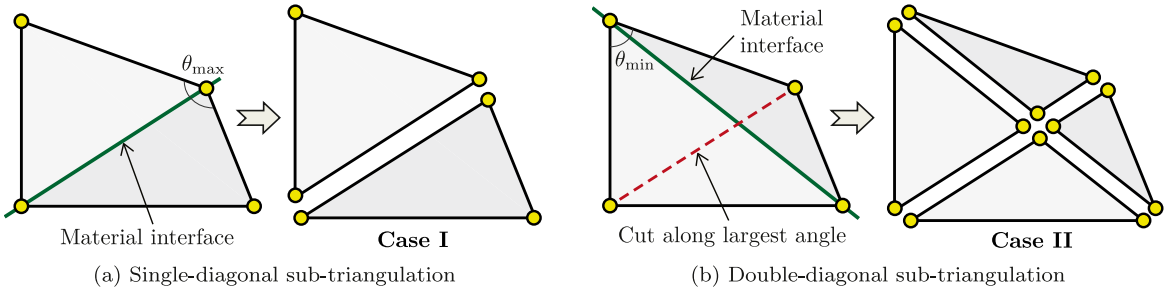


Fig. 4. Single- and double-diagonal rules for sub-triangulating a background element cut by the interface.

to elements with hanging nodes, which could emerge due to applying SAMR to their neighboring elements. To ensure that resulting element aspect ratios do not exceed three, the following rules must be followed for their sub-triangulation:

- **Case I:** If the Q4 element does not intersect the interface along the diagonal emanating from its smallest angle θ_{min}, sub-triangles are created by cutting that element along the diagonal corresponding to its largest angle (cf. Fig. 4(a)).
- **Case II:** Otherwise, two case scenarios could occur:
  - If $\theta_{\text{min}} > 60°$: subdivide by cutting along θ_{min}, as the aspect ratios of resulting sub-triangles would still be acceptable.
  - If $\theta_{\text{min}} < 60°$: subdivide by cutting along both diagonals, which results in creating four conforming sub-triangles (cf. Fig. 4(b)).

4. Modeling crack growth problems using CISAMR

4.1. Meshing the initial crack

In order to generate the conforming mesh corresponding to initial crack(s) embedded in the domain using CISAMR, we must interact each crack with the background mesh. In all examples presented here initial shape of cracks are modeled as line segments passing though two crack tips. To identify the background elements cut by

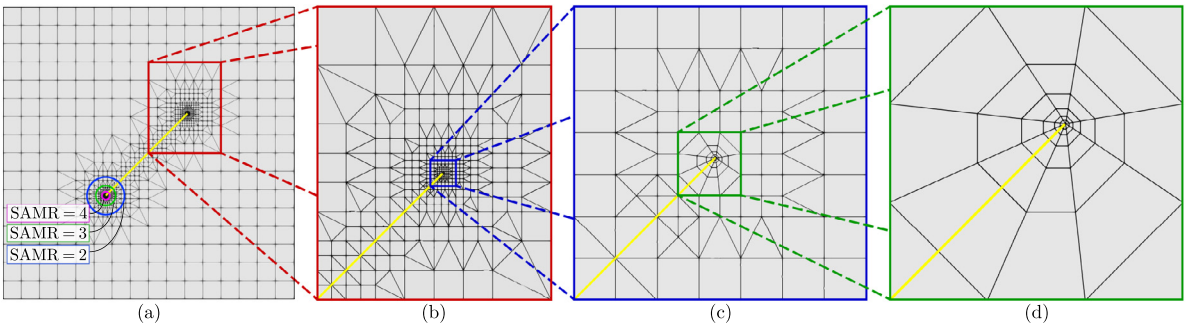


Fig. 5. (a) The conforming mesh generated using CISAMR for a domain with a 45° initial crack, showing three circular regions centered at the crack tip with higher SAMR levels; (b,c) enlarged views of the mesh near the crack tip; (d) spider web pattern of elements in the vicinity of the crack tip.

each crack, a quadtree search algorithm is first implemented to find a background element holding one of the crack tips. We then march from this element towards the second crack tip to locate the remainder of elements intersecting the crack body without searching the entire domain (see [63] for details).

The first step of the CISAMR algorithm is the SAMR phase, which is conducted similarly to that described in Section 3 along the crack length. However, because an accurate prediction of the crack growth path requires an accurate approximation of SIFs through calculating the interaction integral I, as outlined in (12)–(16), additional levels of refinement must be applied in the vicinity of crack tips. Given the singularity of the field at crack tips, creating refined elements with proper aspect ratios is crucial to minimize the error in the recovery of gradients (stresses and strains) used in the calculation of I. To achieve this goal, we adopt a topological refinement strategy, where all background elements with at least one node at distance R_{SAMR} from the crack tip are subjected to N_{SAMR} levels of refinement, *i.e.*, each Q4 background element is recursively refined into four Q4 sub-elements N_{SAMR} times. For the example problem shown in Fig. 5a, four additional levels of refinement are applied near the tip of the 45° crack embedded in the domain. As shown in this figure, since in CISAMR any element with more than one hanging node emerged from adaptive refinement of its neighboring element must be subjected to SAMR before sub-triangulation, the refined region further extends beyond R_{SAMR} and gradually fades to blend with the coarse background mesh. This automatically leads to a decreasing SAMR level as we move away from the crack tip, where field gradients sharply drop and highly refined elements are no longer required for their accurate approximation.

The next two steps of the CISAMR meshing process, *i.e.*, r-adaptivity and sub-triangulation, are conducted mostly similar to those described in Section 3 with only two adjustments: (i) identifying elements located along the crack face, separating degrees of freedom at nodes along the crack, and identifying master–slave element edges for the contact model; (ii) modifying the mesh structure near the crack tip to generate a spider-web pattern of elements (*cf*. Fig. 5d) for the efficient and accurate evaluation of the M-integral. The former is a rather straightforward task after the completion of the sub-triangulation phase. It begins by identifying sub-triangles with two nodes along the crack face marching from one crack tip to the other and separating the degrees of freedom (DOFs) at these nodes to model the strong discontinuity along the crack length. The corresponding element edges are then paired and grouped into slave and master edges on opposing sides of the crack, which will be written into the input file fed to ABAQUS for simulating the linear elastic response of the domain taking into account potential contact/friction forces along the crack face. Note that the element edges on opposing sides of the crack face, one being labeled as master and the other as slave, can easily be distinguished based on the counter clockwise node numbering of each element: If corresponding node numbers appear successively in this numbering they belong to one side of the crack and otherwise to the opposing side.

Generating a spider-web pattern of elements near the crack tip highly facilitates the selection of appropriate elements in this region for the accurate approximation of M-integral. Note that, as shown in [66], the calculation of M-integral over a domain in the vicinity of the crack tip that excludes the crack tip itself yields the most accurate approximation of the energy release rate and thereby the crack path. As shown in Fig. 5d, such domain can easily be identified as each ring of elements in a spider web pattern. Fig. 6 illustrates the non-iterative process of generating spider web elements using CISAMR, starting by deleting the background element holding the crack

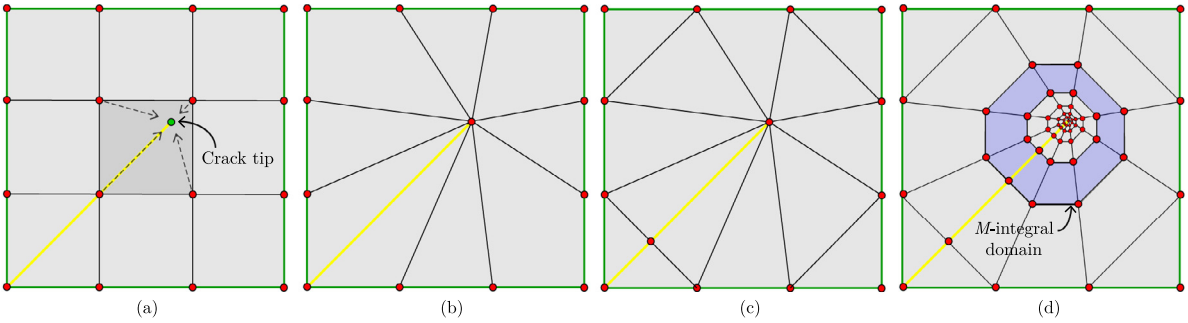


Fig. 6. Process of creating the spider web element pattern: (a) identify the background element holding the crack tip; (b) deleting this element after snapping all its nodes to the crack tip; sub-triangulating the remaining rectangular elements after performing this node snapping; (c) further sub-triangulating the triangular elements with a node at the crack tip in a radial pattern with respect to this node.

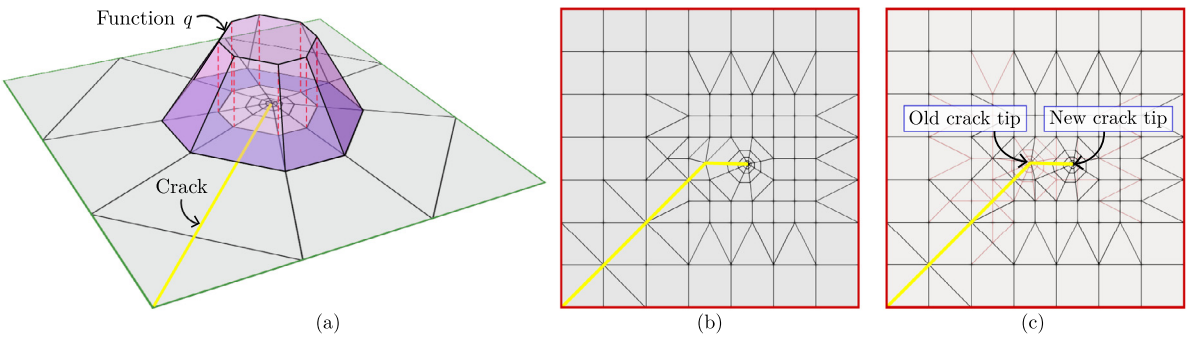


Fig. 7. (a) The function q corresponding to the M-integral domain depicted in Fig. 6d; (b) The crack growth in the second time step corresponding to the initial mesh view depicted in Fig. 5c; (c) comparison between the old and new meshes (old elements are shown in a lighter color).

tip (cf. Fig. 6a) followed by snapping all four nodes of its neighboring elements to the tip during the r-adaptivity process (cf. Fig. 6b). The standard sub-triangulation phase of CISAMR is carried out on this region identical to that described in Section 3, resulting in cutting the Q4 elements sharing the crack tip along their shorter diagonal to create conforming sub-triangles (cf. Fig. 6c). The spider web is then created by cutting the triangles sharing the crack tip in the middle of the edge sharing this node to create T3 and Q4 sub-elements, where the former is recursively subjected to the same process N_{web} times. As shown in Fig. 6d, the final spider web pattern has a transition from extremely fine elements very close to the crack tip (where they are most needed) to coarser elements moving away from this point.

After the construction of the spider web pattern of elements near the crack tip, any ring of Q4 elements similar to that shown in Fig. 6d can be used for approximating the M-integral. To minimize the numerical error, we use the average of the integral values obtained for all rings except for the first two rings (where field gradients and corresponding interpolation error are excessively high) to calculate SIFs and subsequently the growth angle θ_p. Note that the calculation of M-integral for each ring also requires adopting an arbitrary weighting function q, which in this work assumed to vary linearly from 1 along the inner boundary to 0 along the outer boundary of the ring, as shown in Fig. 7a. While dependent on the size of the background mesh, our numerical experiments showed that using $\max(N_i) = 3$ or 4 and $N_{\text{web}} = 4$ or 5 often yields an accurate approximation of the crack growth path with no oscillations during the simulation.

4.2. Remeshing to simulate the crack growth

After approximating θ_p over the spider web corresponding to each crack tip to determine its new location based on a user-defined growth length Δa, as shown in Fig. 7b, we must remesh the domain in this region to enable a similar

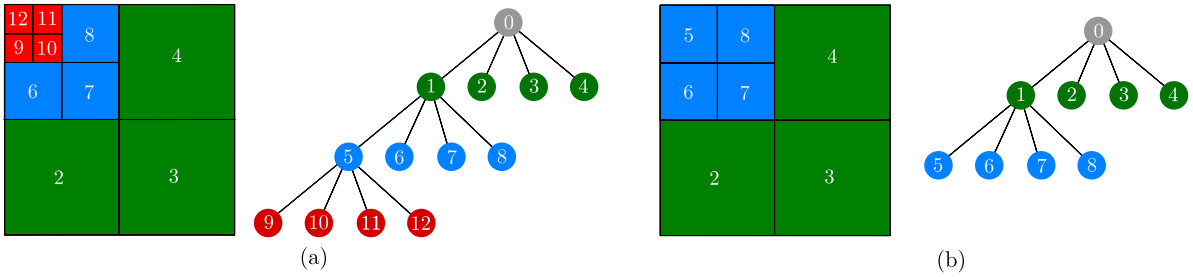


Fig. 8. The tree data structure corresponding to a portion of the mesh (a) before and (b) after the deletion of elements.

task in the next step. The fact that CISAMR non-iteratively transforms an initial structured mesh to a conforming mesh is a key advantage here, as it preserves the hierarchy of elements during the SAMR, r-adaptivity, and sub-triangulation phases, which in turn facilitates the deletion and reconstruction of sub-elements while remeshing the new crack geometry at each load step. As shown in Fig. 8, this can easily be achieved using a tree data structure to keep track of parent and children elements while generating the mesh, with roots representing original background elements and leaves the most refined (conforming) elements of the final mesh. Note that only leaf elements contribute to the assembly of the stiffness matrix for approximating the field and other elements are merely kept in this data structure to facilitate the deletion of unwanted elements and generating new elements during the remeshing process.

After determining the new crack tip at each load step, a line segment connecting new and old crack tips is appended to the crack geometry to simulate the crack growth. All children elements with at least one node located in an area with radius R_{del} centered at the old crack tip are then deleted and this region is remeshed to locally update the mesh, as well as the corresponding tree data structure. The radius of the deletion area is selected as

$$R_{\text{del}} = R_{\text{SAMR}} + N_{\text{SAMR}} h_0, \tag{16}$$

where h_0 is the size of the background element. Note that this value of R_{del} ensures that all children elements generated during the SAMR phase near the crack tip, as well as children of their neighboring elements with hanging nodes that are subjected to subtriangulation are deleted before reconstructing the mesh for the new crack tip. The local remeshing process near the new crack tip, involving applying additional SAMR and generating the spider web, is identical to that previously described in Section 4.1.

It is worth emphasizing that, without any major additional effort, CISAMR can handle remeshing the domain when two nearby cracks are merging (cf. Fig. 9). To simulate problems with multiple embedded cracks, we must keep track of the minimum distance d_{min} between every crack tip to other growing cracks, as well as angle θ_{min} of the corresponding line segment with the crack orientation. If $d_{\text{min}} < \Delta a \cos(\theta_{\text{min}})$, rather than generating a spider web element pattern at the crack tip, we imply snap the tip to the nearby crack, followed by using the hierarchical r-adaptivity scheme and standard sub-triangulation algorithms to remesh this region; cf. Fig. 9(b). Note that no additional SAMR is required in this region, as after the disappearance of crack tip corresponding stress concentrations are vanishing as well.

4.3. Coupling CISAMR code with ABAQUS

While in this work we only study linear elastic problems, the occurrence of contact and friction along crack faces under mixed mode loading and in particular after crack merging demands a robust solver to simulate these nonlinear phenomena. In fact, given the high nonlinearity of the problem in the presence of contact/friction forces, our numerical studies show that an implicit solver is often not capable of achieving convergence in domains with multiple growing cracks. This has motived using the robust explicit dynamic solver of ABAQUS, with the ability to model contact and friction along crack faces, to simulate crack growth problems in this work.

In order to integrate our in-house CISAMR C++ code with ABAQUS, at each load step, the updated mesh structure (nodes and their connectivity), master/slave element edges along crack faces, and applied BCs are written in a standard ABAQUS input file to perform the FE simulation. After simulating the field, nodal displacements and recovered stresses associated with each ring of the spider web elements of each crack tip are written in an ABAQUS

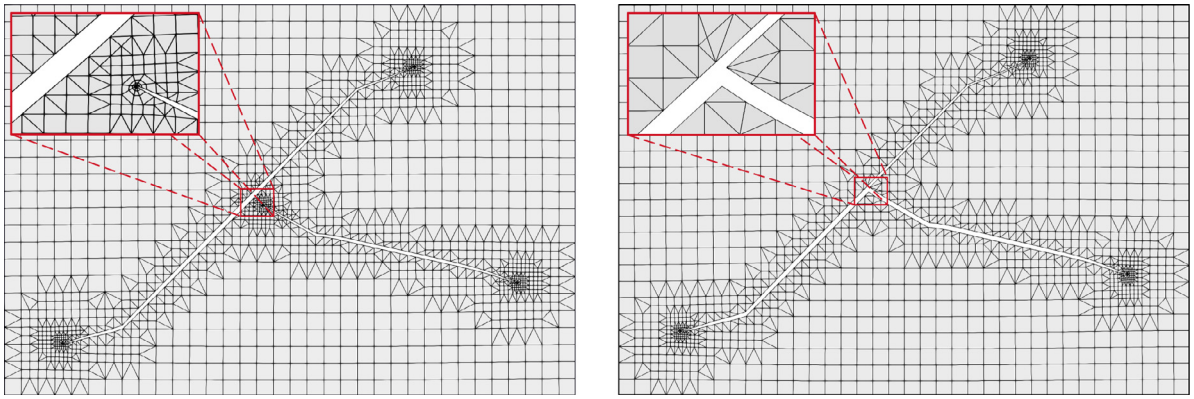


Fig. 9. CISAMR simulation of the interaction between two cracks (a) before their intersection and (b) after their merging.

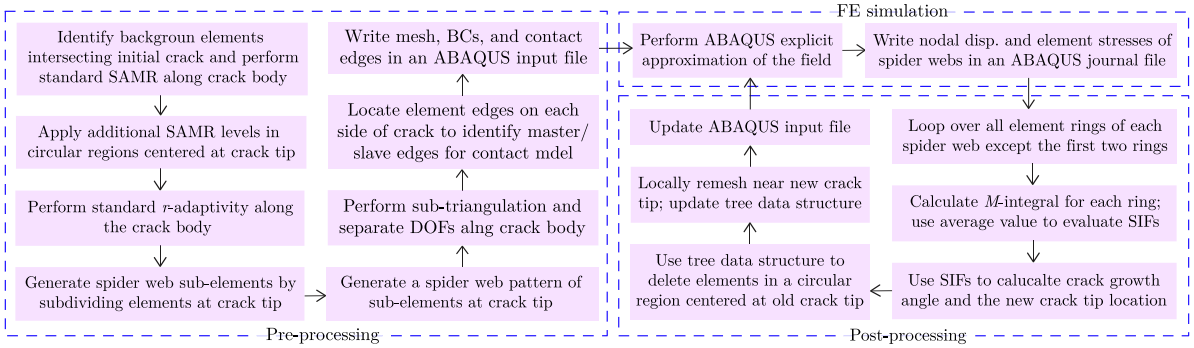


Fig. 10. Flow chart of the algorithm used for simulating crack growth problems by integrating CISAMR and ABAQUS.

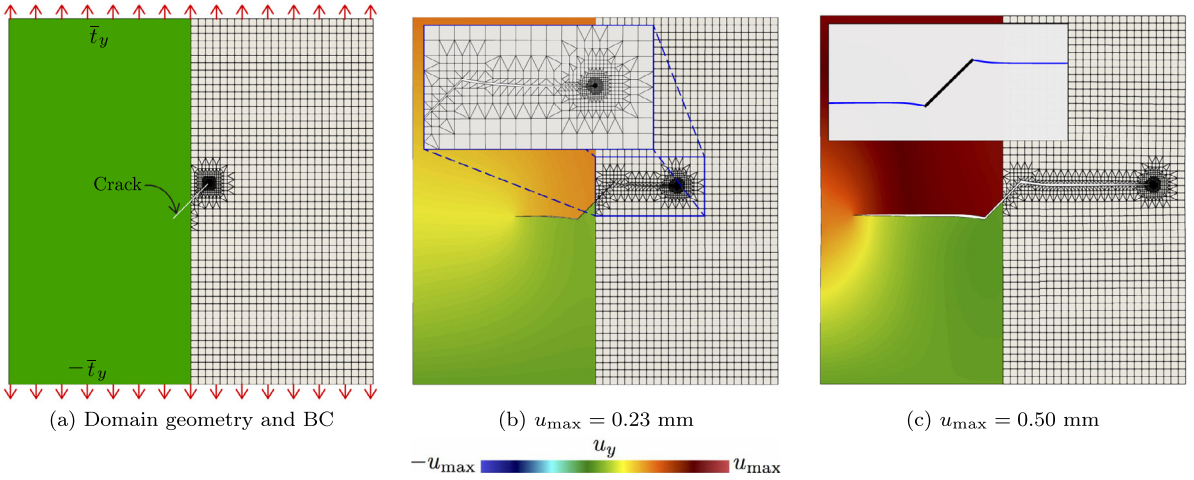
journal file. The key advantage of generating a spider web pattern of elements, where elements belonging to each ring and their connectivity are known a priori becomes evident at this stage, as it minimizes the post-processing phase and eliminates the need to write a complete ABAQUS output file (*i.e.*, an ODB file) and search for elements used in calculation of M-integral in the post-processing phase. The information written in the journal file is then used by our C++ code to evaluate the M-integral, SIFs, crack orientation, and subsequently the new location for all crack tips. The CISAMR algorithm is utilized to remesh the domain and write the results in a new ABAQUS input file to resume this recursive process until we reach either complete failure or a certain number of crack advance steps. A flow chart summarizing this algorithm is presented in Fig. 10.

5. Numerical examples

Six example problems are presented in this section, with the first four being benchmark problems aimed at verifying the accuracy of crack growth simulations relying on CISAMR through comparison with other numerical or experimental results. The last two examples are more geometrically elaborate fracture problems that demonstrate the ability of CISAMR to simulate complex multi-crack growth problems involving contact and friction along crack faces. In all the examples presented below, the spider web element pattern generated at the crack tip to evaluate the M-integral has five layers of Q4 elements and the crack growth length, Δa, is assumed to be 5% of the initial crack length.

5.1. Inclined crack under tension

In this example, we simulate the growth of an initial crack with length $2a = 0.14$ mm and orientation 45° embedded at the centroid of a 1 mm × 1 mm domain subjected to a tensile load in the y-direction, which is applied



(a) Domain geometry and BC (b) $u_{\max} = 0.23$ mm (c) $u_{\max} = 0.50$ mm

Fig. 11. First example problem: Simulated crack growth path and CISAMR mesh structure in a domain with an initial 45° crack subjected to a tensile load. The inset of Figure (c) shows the results approximated using the distributed dislocation method in [69].

as a traction BC, $\bar{t}_y$ along the top and bottom edges of the domain; *cf*. Fig. 11(a). Under plane strain condition, the elastic modulus and Poisson's ratio of the domain are assumed to be $E = 10^3$ and $\nu = 0.3$, respectively. A 50×50 structured background mesh is employed to generate the initial FE model and its subsequent local remeshing as the crack grows using CISAMR. One level of SAMR is applied along the crack length, while an area with radius $R_{\text{SAMR}} = 0.08$ mm centered at the crack tips is subjected to three additional levels of SAMR, as illustrated in the inset of Fig. 11(b).

Figs. 11(b) and 11(c) show two snapshots of the simulated crack growth path, corresponding mesh structures, and displacement fields in the domain approximated using the CISAMR-ABAQUS framework described previously. Due to the 45° orientation of the initial crack, at the first load step, we deal with a mixed mode fracture problem resulting in a kink in the crack direction, which can easily be handled by CISAMR during the remeshing process. Since the domain is subjected to a tensile load, mode I quickly becomes the dominant fracture mode while K_{II} goes to zero after a few load steps, meaning the crack grows perpendicular to the loading direction (*cf*. Fig. 11(c)). The inset of Fig. 11(c) also compared our simulated crack growth path with the results approximated using distributed dislocation method in [69], which shows a perfect agreement between the two simulations. It is worth noting that by applying additional SAMR levels and creating spider web elements at the crack tip, the crack grows with no oscillations due to the accurate approximation of the M-integral used in predicting the crack path. To demonstrate the mesh independency of the results presented above, we studied the effect of three background meshes of size 25×25, 50×50, and 100×100 on the predicted crack growth path. The resulting crack trajectories are compared in Fig. 12, showing the simulation relying on the coarsest mesh leads to only a slightly different crack path than the other two simulation, while the difference between results on 50×50 and 100×100 is negligible.

The analytical solution for mode I and II stress intensity factors in this problem are given by

$$K_I^{\text{exact}} = K_{I(0)}\cos^2(\beta), \qquad K_{II}^{\text{exact}} = K_{I(0)}\cos(\beta)\sin(\beta), \tag{17}$$

where β is the crack orientation angle and $K_{I(0)} = \bar{t}_y\sqrt{\pi a}$ is the SIF for mode I when the crack is horizontal ($\beta = 0$) [70]. The variation of relative errors in predicting normalized SIFs versus the background mesh size are illustrated in Fig. 13 for different background mesh sizes and SAMR levels. It is seen that increasing the refinement levels near the crack tip not only leads to a significant decrease in error but also improves the convergence rate. Moreover, even for the coarsest background mesh, the errors associated with predicting SIFs would be below 0.5% when $N_{\text{SAMR}} = 4$, which enables accurate prediction of the crack path for simulating the crack growth process.

5.2. Two parallel cracks

In this example, we use CISAMR to simulate the growth of two parallel cracks with the same length ($l = 2$ mm) and at distance $d_c = 1$ mm, which are embedded in a 10 mm $\times$ 10 mm domain as shown in Fig. 14. The domain

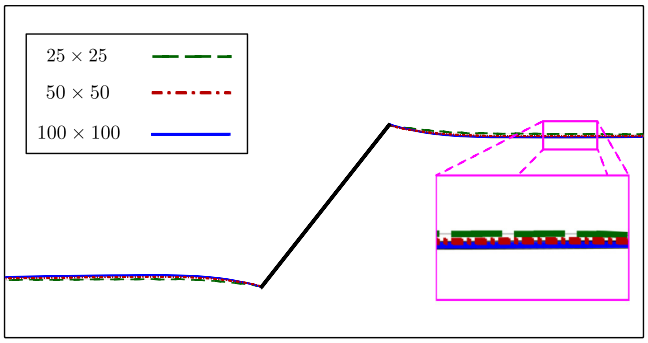


Fig. 12. First example problem: Crack propagation trajectories for different background element sizes.

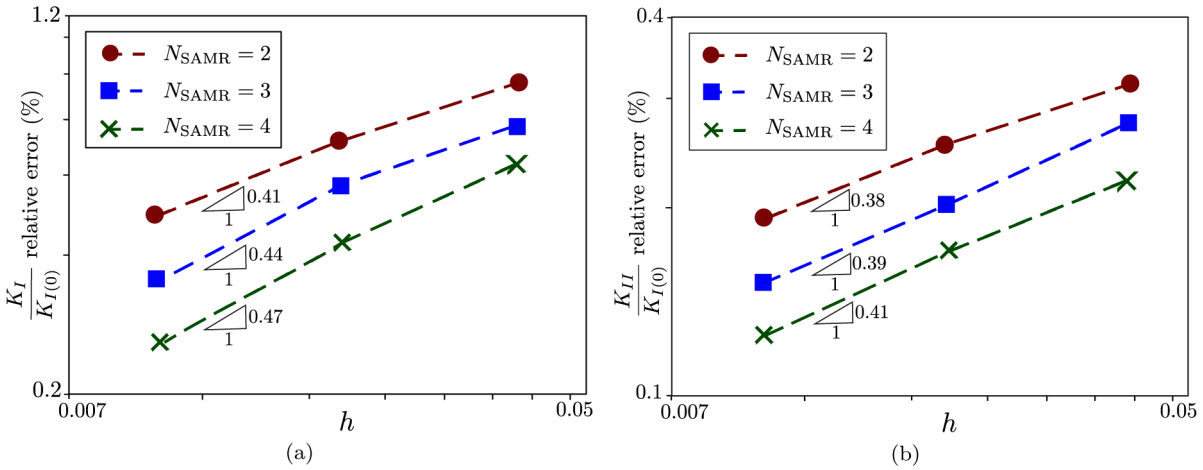
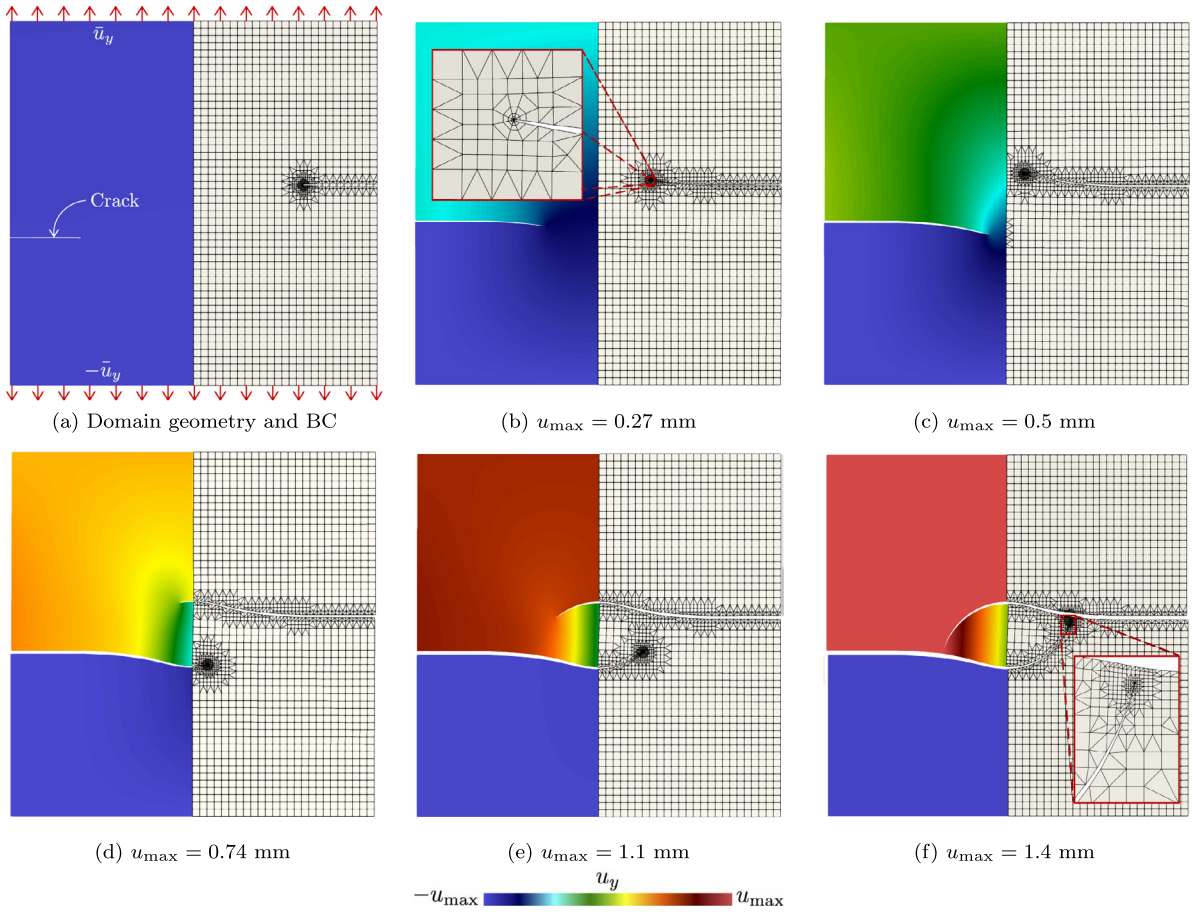


Fig. 13. First example problem: Normalized SIFs relative error versus the background mesh size for different refinement levels near the crack tip for (a) mode I and (b) mode II.

is subjected to a tensile displacement BC in the y direction, $\bar{u}_y$, and discretized using a 50 × 50 background mesh with one level of SAMR along the crack body. The crack tip is subjected to three additional levels of refinement within a distance of $R_{\text{SAMR}} = 0.4$ mm from the crack tip. The domain has elastic modulus $E = 70$ MPa and Poisson's ratio $\nu = 0.3$ and the simulation is conducted under the assumption of plane stress condition. Multiple snapshots of the simulated crack geometry, CISAMR mesh, and the displacement field in this domain are depicted in Fig. 14. Note how adaptively refined, high-quality meshes generated by CISAMR in this problem yield a smooth crack path with no oscillations during the crack growth simulation. Further, CISAMR maintains a perfect symmetry of the shape of both growing cracks due to accurate prediction of SIFs after applying additional SAMR levels near the crack tips. These simulations are similar to XFEM results for the same problem presented in [71].

5.3. Domain with two circular holes

In this example, we simulate the crack growth in the 20 cm × 10 cm rectangular domain shown in Fig. 15, which contains two circular holes with diameter $D = 2$ cm and two initial horizontal cracks with $l = 1$ cm. The domain is subjected to a tensile displacement BC in the y-direction along the top edge of the domain, while displacement along the bottom edge is constrained in the same direction. A 100 × 50 background mesh is used for generating the CISAMR mesh using one level of SAMR along the crack body and the circular holes. Three additional levels of refinement are applied in a circular region with $R = 0.25$ cm centered at the crack tip. The material properties used in this example are $E = 10^3$ and $\nu = 0.3$ and the simulation is carried out under the plane strain condition.



(a) Domain geometry and BC (b) $u_{\max} = 0.27$ mm (c) $u_{\max} = 0.5$ mm

(d) $u_{\max} = 0.74$ mm (e) $u_{\max} = 1.1$ mm (f) $u_{\max} = 1.4$ mm

Fig. 14. Second example problem: Simulated crack growth path and CISAMR mesh structure in a domain with two initial parallel cracks subjected to a tensile load.

Fig. 15 illustrates three snapshots of the simulated crack path, CISAMR mesh, and displacement field in this example problem, where Fig. 15(d) shows the CISAMR ability to capture cracks merging with circular holes in last step of the simulation. It is worth noting that the crack morphologies shown in this example are similar to simulation results obtained using adaptive mesh refinement based on modified super convergent patch recovery in [44], but with even less oscillations in predicting the crack path due to more accurate approximation of the M-integral.

5.4. Three-point bending of a beam with three circular holes

In this example, we further verify the accuracy of CISAMR for modeling crack growth problems by replicating a three-point bending test and comparing resulting fracture patterns with experimental results presented in [51]. Fig. 16 shows the domain geometry and applied BCs for this problem, which contains three circular holes with diameter $D = 5$ in and a vertical crack on the bottom edge with length a. The beam is made of polymethyl methacrylate (PMMA) with $E = 199.95$ GPa and $\nu = 0.3$, which is modeled as a plane strain problem. The CISAMR mesh is generated using a 200×80 background mesh with one level of SAMR along the crack body and circular holes. Also, an area with radius $R = 0.2$ in centered at the crack tip is subjected to 3 additional levels of SAMR during the meshing/remeshing process.

Following the three-point bending tests conducted in [51], two sets of simulations are conducted considering different initial crack lengths of $a = 1$ in and $a = 1.5$ in. An intermediate and the final simulated crack geometries for each case scenario, together with corresponding CISAMR mesh structures, and also the experimental results are

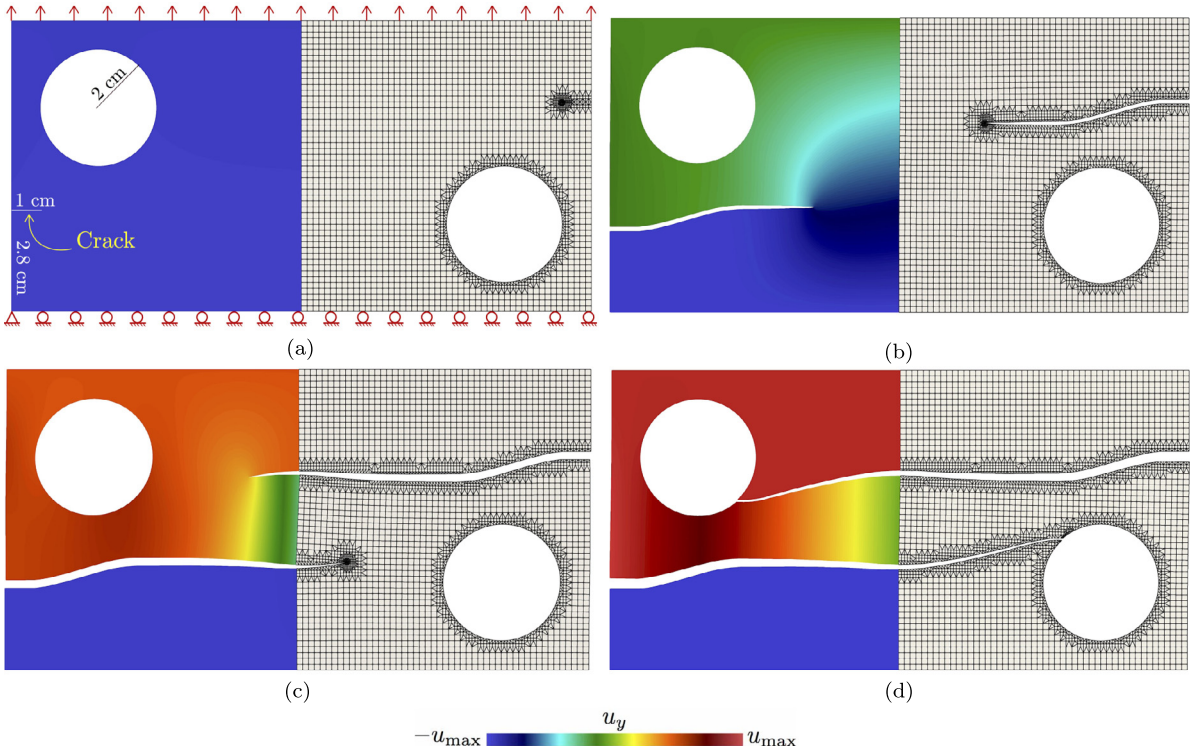


Fig. 15. Third example problem: Simulated crack growth path and CISAMR mesh structure in a domain with two circular holes and two initial parallel cracks subjected to a tensile load.

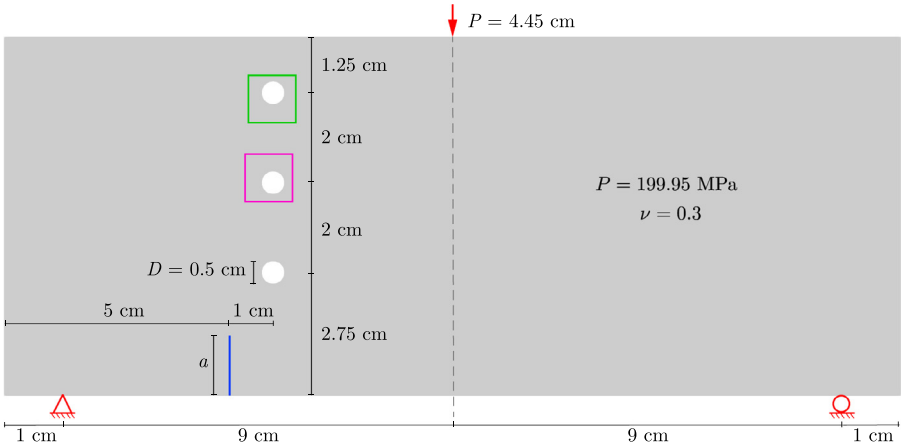
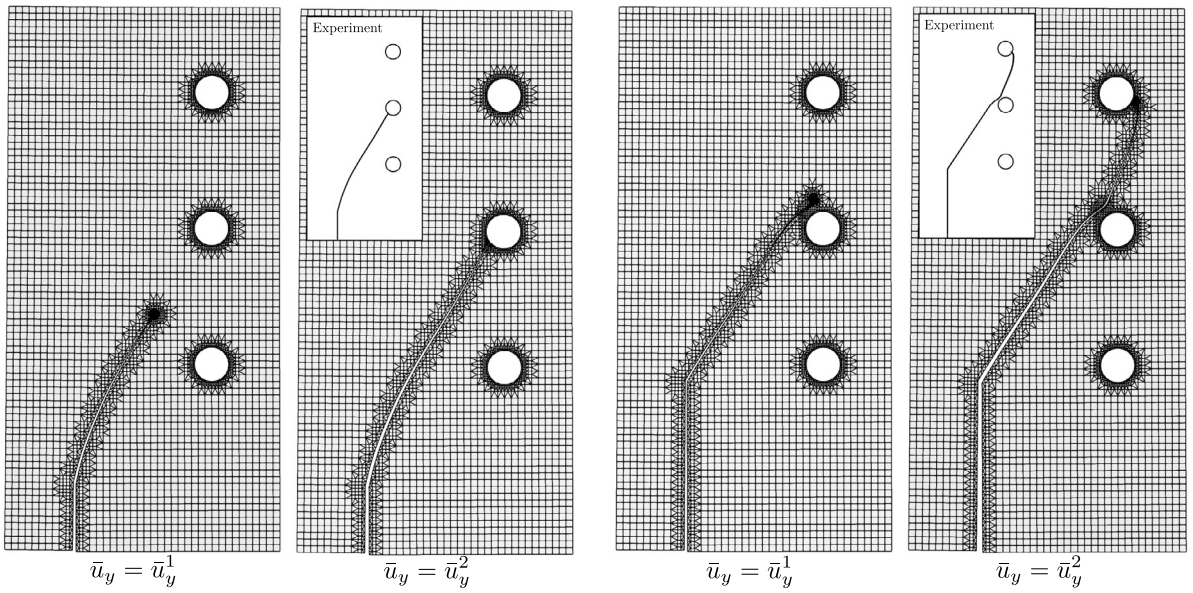


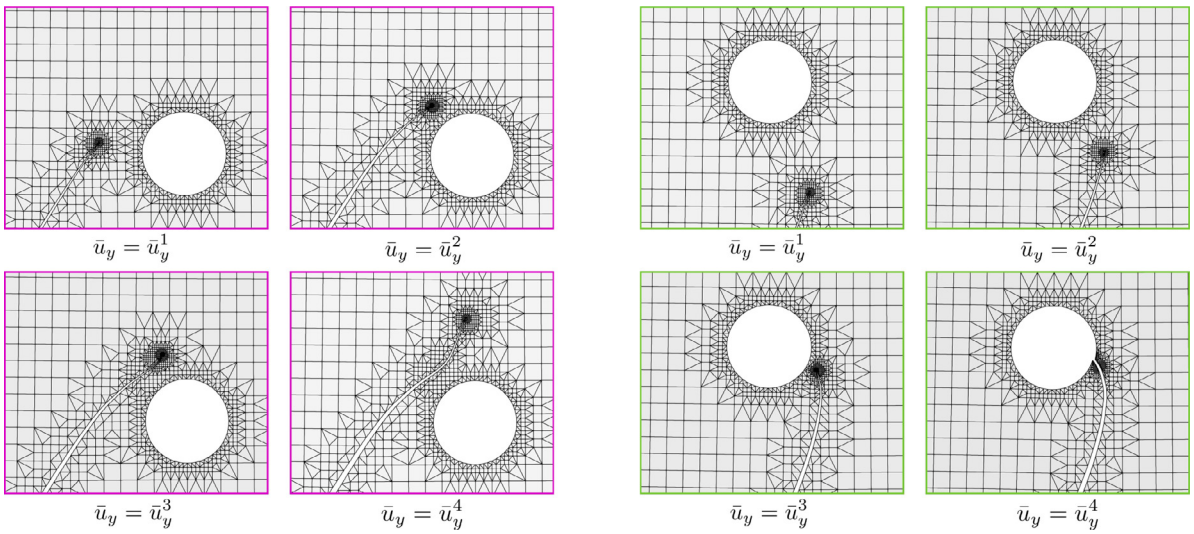
Fig. 16. Fourth example problem: domain geometry and boundary conditions.

illustrated in Fig. 17. Note that the difference between initial crack lengths (i.e., $a = 1$ in versus $a = 1.5$ in) leads to completely distinct crack trajectories, with the former merging with the middle hole while the latter narrowly passes by this hole and merges the top hole. Also, it is seen that the simulation results for both cases show an excellent agreement with crack shapes obtained from three-point bending experiments in [72]. Fig. 18 better shows the evolution of the mesh structure and the lack of oscillation in the predicted crack path near the middle and top holes for the simulation with $a = 1.5$ in.



(a) $a = 1$ cm (b) $a = 1.5$ cm

Fig. 17. Fourth example problem: crack geometry and mesh structure at intermediate and final stages of the simulation for different initial crack lengths a, as well as their comparison with the experimental results in [72].



(a) middle hole (b) top hole

Fig. 18. Fourth example problem: a closer view of the evolution of the mesh structure generated using CIASMR near the middle and top holes for an initial crack length of $a = 1.5$ in.

5.5. Multiple crack growth and merging

In this example, we demonstrate the CISAMR ability for modeling a more complex crack growth problem involving the growth and merging of 15 cracks with the initial geometry and spatial distribution shown in Fig. 19. As depicted in this figure, the 20 cm $\times$ 10 cm domain with material properties $E = 100$ GPa and $\nu = 0.3$ is subjected

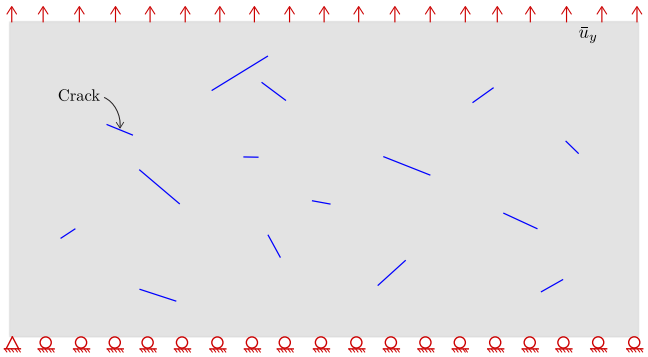


Fig. 19. Fifth example problem: Geometry and BCs of the domain with multiple randomly distributed initial cracks.

to a tensile displacement BC in the y direction. Also, a friction coefficient of $\mu_f = 0.01$ is considered along crack faces. To generate CISAMR meshes throughout the simulation, the domain is discretized using a 200×100 background mesh with one level of SAMR along the crack body and three additional refinement levels within a radius of $R_{\text{SAMR}} = 0.1$ cm from the crack tip.

Fig. 20 illustrates four snapshots of the evolving crack shapes simulated for this problem, which involves the merging of multiple cracks effortlessly handled by CISAMR. Note how the geometric complexity caused by the random distribution of cracks in the domain leads to mixed-mode growth of each crack during the simulation, even though the domain is subjected to a tensile load. The figure also shows that some crack faces collapse on one another after merging with a nearby crack, which gives rise to contact and friction forces approximated by the ABAQUS explicit solver.

5.6. Heterogeneous domain with multiple cracks

In this final example problem, we implement CISAMR to simulate the propagation of four initial cracks in a heterogeneous 1 cm $\times$ 1 cm domain subjected to a biaxial tensile load, as shown in Fig. 21(a). The tensile displacement BC along the right and top edges of the domain linearly ramp up from 0 to 0.1 mm in 50 steps. Mechanical properties of the domain are assumed to be $E_m = 10^4$ and $\nu_m = 0.3$ for the matrix and $E_p = 10^3$ and $\nu_p = 0.3$ for the soft particles embedded in the domain. The friction coefficient along crack faces is $\mu_f = 0.01$. To simulate the crack growth in this domain, CISAMR meshes are constructed on a 100×100 background mesh with one level of SAMR along particle–matrix interfaces and crack bodies. Three additional levels of refinement are applied near the crack tips. Fig. 21 illustrates three snapshots of predicted crack shapes at different stages of this simulation, which leads to the merging of three cracks and contact along one of the crack faces in the last few load steps. The CISAMR non-iterative algorithm is not only capable of handling such cases but can also maintain conforming elements when cracks intersect embedded particles during the remeshing process.

6. Summary and conclusion

A recently introduced meshing algorithm named Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) was expanded for modeling 2D crack growth problems with complex geometries. CISAMR non-iteratively transforms an initial (background) structured mesh into a high-quality conforming mesh with the desired level of adaptive refinement along the crack body, as well as additional refinement level in the vicinity of the crack tip. Moreover, a spider web pattern of elements centered at the crack tip was generated for the easy and accurate approximation of M-integral, which in turn reduces the error associated with evaluating SIFs and subsequently the crack growth orientation at each load step. In addition to describing new algorithmic aspects of CISAMR, we discussed implementation details such as coupling this algorithm with ABAQUS for modeling such problems. Since in multi-crack growth problems involving crack merging, contact between crack faces is often inevitable, using the robust explicit dynamic solver and contact model readily available in ABAQUS highly facilitated approximating the domain response. Several example problems were presented to verify the accuracy of simulations relying on

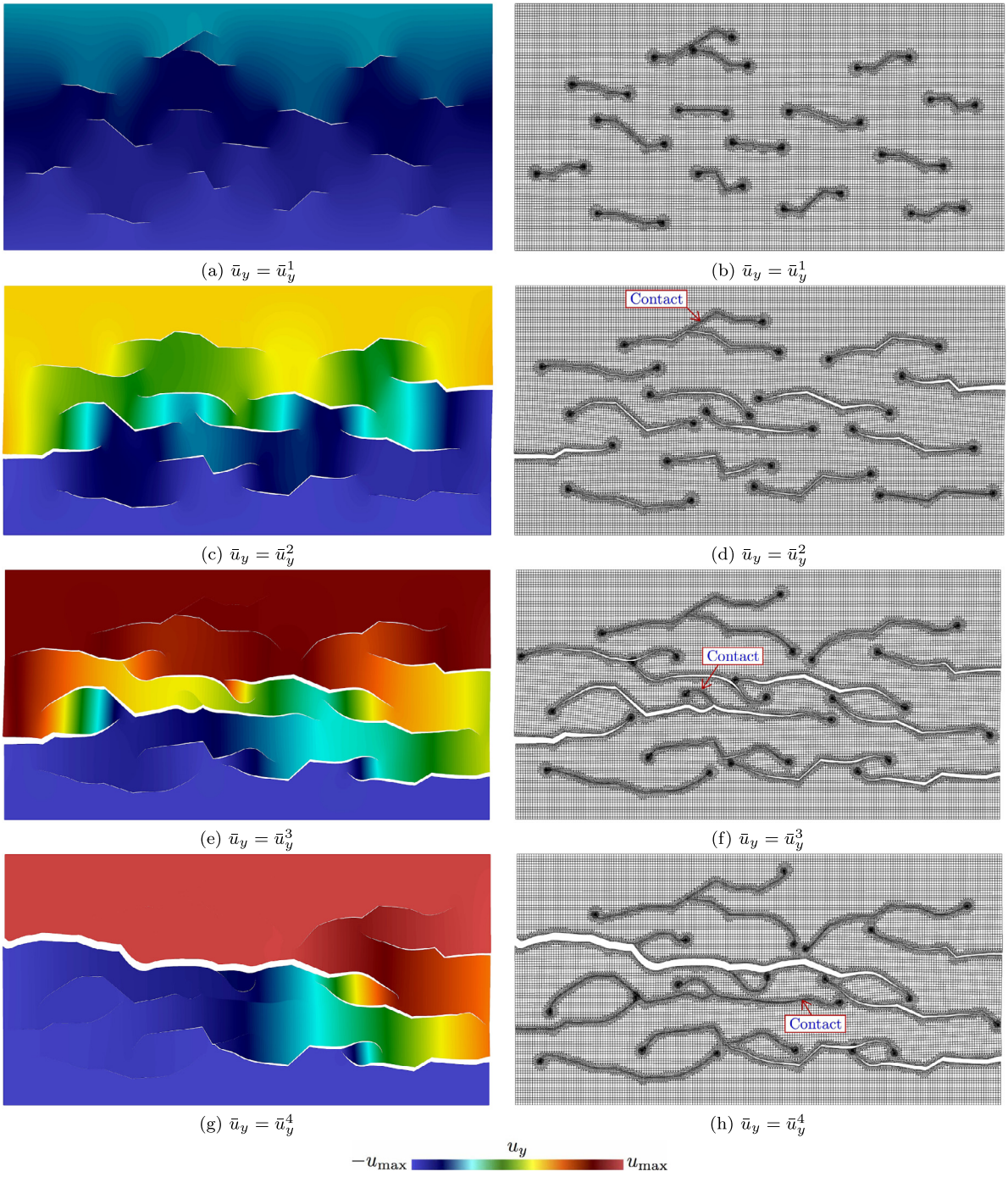


Fig. 20. Fifth example problem: Four snapshots of the displacement field and crack paths at different load steps, showing contact along some crack faces after merging with nearby cracks.

CISAMR algorithm for modeling crack growth problem through comparing results with predictions made by other numerical techniques or experimental data. Two additional problems were presented to demonstrate the coupled CISAMR-ABAQUS framework ability to capture crack merging and intersections between cracks and embedded heterogeneities in the domain in mixed-mode multi-crack growth problems.

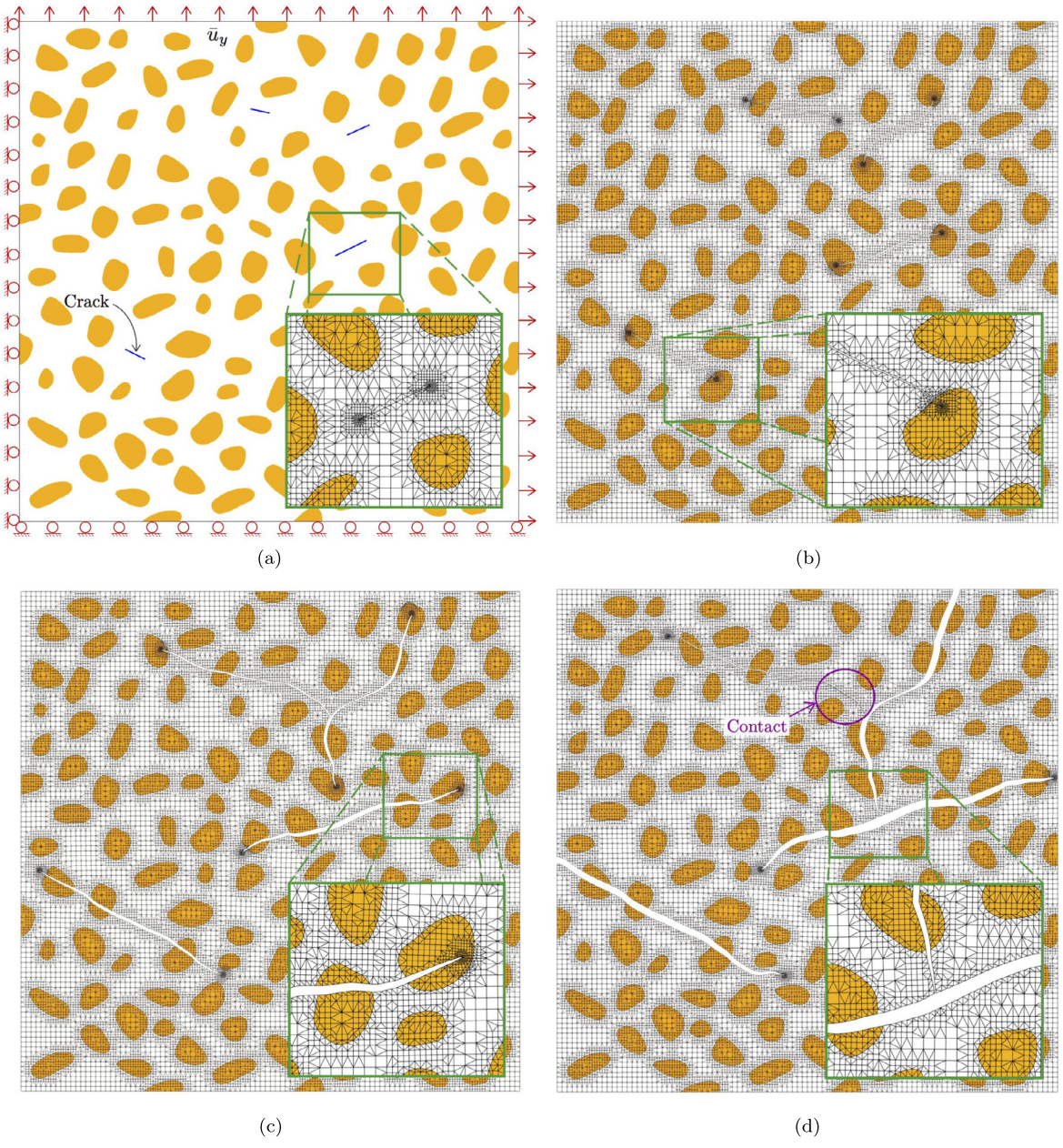


Fig. 21. Sixth example problem: (a) Domain geometry and BCs; (b–d) growing cracks shapes and evolving CISAMR mesh at different load steps throughout the simulation, showing cracks merging and crack-particle intersections.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been supported by the Air Force Office of Scientific Research (AFOSR) under grant number FA9550-21-1-0245. The authors also acknowledge the allocation of computing resources by the Ohio State University Simulation Innovation and Modeling Center (SIMCenter) and the Ohio Suercomputer Center (OSC).

References

[1] P. Areias, T. Rabczuk, D.D. da Costa, Element-wise fracture algorithm based on rotation of edges, Eng. Fract. Mech. 110 (2013) 113–137.

[2] G. Yagawa, Y. Sakai, Y. Ando, Analysis of a rapidly propagating crack using finite elements, in: Fast Fracture and Crack Arrest, ASTM International, 1977.

[3] A.O. Bolukbasi, D.H. Laananen, Analytical and experimental studies of crushing behavior in composite laminates, J. Compos. Mater. 29 (8) (1995) 1117–1139.

[4] S. Beissel, G. Johnson, C. Popelar, An element-failure algorithm for dynamic crack propagation in general directions, Eng. Fract. Mech. 61 (3–4) (1998) 407–425.

[5] N. Vieira De Carvalho, R. Krueger, G.E. Mabson, L. Deobald, Combining progressive nodal release with the virtual crack closure technique to model fatigue delamination growth without re-meshing, in: 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2018, p. 1468.

[6] T. Belytschko, L. Gu, Y. Lu, Fracture and crack growth by element free Galerkin methods, Modelling Simulation Mater. Sci. Eng. 2 (3A) (1994) 519.

[7] T. Belytschko, Y. Lu, L. Gu, M. Tabbara, Element-free Galerkin methods for static and dynamic fracture, Int. J. Solids Struct. 32 (17) (1995) 2547–2570.

[8] N. Sukumar, B. Moran, T. Black, T. Belytschko, An element-free Galerkin method for three-dimensional fracture mechanics, Comput. Mech. 20 (1) (1997) 170–175.

[9] T. Belytschko, Y. Lu, L. Gu, Element-free Galerkin methods, Internat. J. Numer. Methods Engrg. 37 (2) (1994) 229–256.

[10] W. Liu, S. Hao, T. Belytschko, S. Li, C. Chang, Multiple scale meshfree methods for damage fracture and localization, Comput. Mater. Sci. 16 (1) (1999) 197–205.

[11] T. Rabczuk, T. Belytschko, Cracking particles: a simplified meshfree method for arbitrary evolving cracks, Internat. J. Numer. Methods Engrg. 61 (13) (2004) 2316–2343.

[12] M. Borden, C. Verhoosel, M. Scott, T. Hughes, C. Landis, A phase-field description of dynamic brittle fracture, Comput. Methods Appl. Mech. Engrg. 217 (2012) 77–95.

[13] M. Ambati, T. Gerasimov, L. De Lorenzis, A review on phase-field models of brittle fracture and a new fast hybrid formulation, Comput. Mech. 55 (2) (2015) 383–405.

[14] I. Singer-Loginova, H. Singer, The phase field technique for modeling multiphase materials, Rep. Progr. Phys. 71 (10) (2008) 106501.

[15] E. Tanné, T. Li, B. Bourdin, J.-J. Marigo, C. Maurini, Crack nucleation in variational phase-field models of brittle fracture, J. Mech. Phys. Solids 110 (2018) 80–99.

[16] J.-Y. Wu, V.P. Nguyen, A length scale insensitive phase-field damage model for brittle fracture, J. Mech. Phys. Solids 119 (2018) 20–42.

[17] T.K. Mandal, V.P. Nguyen, J.-Y. Wu, Length scale and mesh bias sensitivity of phase-field models for brittle and cohesive fracture, Eng. Fract. Mech. 217 (2019) 106532.

[18] J.G. Rots, P. Nauta, G. Kuster, J. Blaauwendraad, Smeared crack approach and fracture localization in concrete, HERON 30 (1) (1985).

[19] R. de Borst, Fracture in quasi-brittle materials: a review of continuum damage-based approaches, Eng. Fract. Mech. 69 (2) (2002) 95–112.

[20] M. Cervera, M. Chiumenti, Smeared crack approach: back to the original track, Int. J. Numer. Anal. Methods Geomech. 30 (12) (2006) 1173–1199.

[21] J.G. Rots, Computational modeling of concrete fracture, 1988.

[22] G. Pijaudier-Cabot, Z.P. Bažant, Nonlocal damage theory, J. Eng. Mech. 113 (10) (1987) 1512–1533.

[23] R.H. Peerlings, R. de Borst, W.M. Brekelmans, J. de Vree, Gradient enhanced damage for quasi-brittle materials, Internat. J. Numer. Methods Engrg. 39 (19) (1996) 3391–3403.

[24] T. Belytschko, R. Gracie, G. Ventura, A review of extended/generalized finite element methods for material modeling, Modelling Simulation Mater. Sci. Eng. 17 (4) (2009) 043001.

[25] N. Moës, T. Belytschko, Extended finite element method for cohesive crack growth, Eng. Fract. Mech. 69 (7) (2002) 813–833.

[26] J. Dolbow, N. Moës, T. Belytschko, An extended finite element method for modeling crack growth with frictional contact, Comput. Methods Appl. Mech. Engrg. 190 (51) (2001) 6825–6846.

[27] J. Pereira, C.A. Duarte, X. Jiao, Three-dimensional crack growth with hp-generalized finite element and face offsetting methods, Comput. Mech. 46 (3) (2010) 431–453.

[28] T. Fries, A corrected XFEM approximation without problems in blending elements, Internat. J. Numer. Methods Engrg. 75 (5) (2008) 503–532.

[29] G. Zi, T. Belytschko, New crack-tip elements for XFEM and applications to cohesive cracks, Internat. J. Numer. Methods Engrg. 57 (15) (2003) 2221–2240.

[30] E. Chahine, P. Laborde, Y. Renard, Crack tip enrichment in the XFEM using a cutoff function, Internat. J. Numer. Methods Engrg. 75 (6) (2008) 629–646.

[31] N. Moës, T. Belytschko, Extended finite element method for cohesive crack growth, Eng. Fract. Mech. 69 (2002) 813–833.

[32] S. Nicaise, Y. Renard, E. Chahine, Optimal convergence analysis for the extended finite element method, Internat. J. Numer. Methods Engrg. 86 (4–5) (2011) 528–548.

[33] A. Simone, C. Duarte, E.V. der Giessen, A generalized finite element method for polycrystals with discontinuous grain boundaries, Internat. J. Numer. Methods Engrg. 67 (8) (2006) 1122–1145.

[34] A.M. Aragón, A. Simone, The discontinuity-enriched finite element method, Internat. J. Numer. Methods Engrg. 112 (11) (2017) 1589–1613.
[35] S. Soghrati, A.M. Aragón, C.A. Duarte, P.H. Geubelle, An interface-enriched generalized finite element method for problems with discontinuous gradient fields, Internat. J. Numer. Methods Engrg. 89 (8) (2012) 991–1008.
[36] S. S., Hierarchical interface-enriched finite element method: An automated technique for mesh-independent simulations, J. Comput. Phys. 275 (2014) 41–52.
[37] A.R. Ingraffea, V. Saouma, Numerical modeling of discrete crack propagation in reinforced and plain concrete, in: Fracture Mechanics of Concrete: Structural Application and Numerical Calculation, Springer, 1985, pp. 171–225.
[38] P. Bouchard, F. Bay, Y. Chastel, I. Tovena, Crack propagation modelling using an advanced remeshing technique, Comput. Methods Appl. Mech. Engrg. 189 (3) (2000) 723–742.
[39] D. Colombo, M. Giglio, A methodology for automatic crack propagation modelling in planar and shell FE models, Eng. Fract. Mech. 73 (4) (2006) 490–504.
[40] T. Nishioka, H. Tokudome, M. Kinoshita, Dynamic fracture-path prediction in impact fracture phenomena using moving finite element method based on delaunay automatic mesh generation, Int. J. Solids Struct. 38 (30) (2001) 5273–5301.
[41] J.C. Neto, P.A. Wawrzynek, M.T. Carvalho, L.F. Martha, A.R. Ingraffea, An algorithm for three-dimensional mesh generation for arbitrary regions with cracks, Eng. Comput. 17 (1) (2001) 75–91.
[42] D. Azócar, M. Elgueta, M. Rivara, Automatic LEFM crack propagation method based on local Lepp–Delaunay mesh refinement, Adv. Eng. Softw. 41 (2) (2010) 111–119.
[43] S. Phongthanapanich, P. Dechaumphai, Adaptive delaunay triangulation with object-oriented programming for crack propagation analysis, Finite Elem. Anal. Des. 40 (13–14) (2004) 1753–1771.
[44] A. Khoei, H. Azadi, H. Moslemi, Modeling of crack propagation via an automatic adaptive mesh refinement based on modified superconvergent patch recovery technique, Eng. Fract. Mech. 75 (10) (2008) 2921–2945.
[45] A. Meyer, F. Rabold, M. Scherzer, Efficient finite element simulation of crack propagation using adaptive iterative solvers, Commun. Numer. Methods. Eng. 22 (2) (2006) 93–108.
[46] P. Areias, T. Rabczuk, Finite strain fracture of plates and shells with configurational forces and edge rotations, Internat. J. Numer. Methods Engrg. 94 (12) (2013) 1099–1122.
[47] P. Areias, T. Rabczuk, P. Camanho, Initially rigid cohesive laws and fracture based on edge rotations, Comput. Mech. 52 (4) (2013) 931–947.
[48] R. Rangarajan, M. Chiaramonte, M. Hunsweck, Y. Shen, A. Lew, Simulating curvilinear crack propagation in two dimensions with universal meshes, Internat. J. Numer. Methods Engrg. 102 (3–4) (2015) 632–670.
[49] R. Rangarajan, H. Kabaria, A. Lew, An algorithm for triangulating smooth three-dimensional domains immersed in universal meshes, Internat. J. Numer. Methods Engrg. 117 (1) (2019) 84–117.
[50] A. Miranda, J.C. Neto, L.F. Martha, An algorithm for two-dimensional mesh generation for arbitrary regions with cracks, in: XII Brazilian Symposium on Computer Graphics and Image Processing (Cat. No. PR00481), IEEE, 1999, pp. 29–38.
[51] P.A. Wawrzynek, A.R. Ingraffea, An interactive approach to local remeshing around a propagating crack, Finite Elem. Anal. Des. 5 (1) (1989) 87–96.
[52] G. Sih, E. Madenci, Fracture initiation under gross yielding: strain energy density criterion, Eng. Fract. Mech. 18 (3) (1983) 667–677.
[53] J. Rice, P. Paris, J. Merkle, Some further results of J-integral analysis and estimates, in: Progress in Flaw Growth and Fracture Toughness Testing, ASTM International, 1973.
[54] H. Okada, S. Ohata, Three-dimensional J-integral evaluation for cracks with arbitrary curvatures and kinks based on domain integral method for quadratic tetrahedral finite element, Eng. Fract. Mech. 109 (2013) 58–77.
[55] J.-H. Kim, G.H. Paulino, Mixed-mode J-integral formulation and implementation using graded elements for fracture analysis of nonhomogeneous orthotropic materials, Mech. Mater. 35 (1–2) (2003) 107–128.
[56] H. Ozer, C. Duarte, I. Al-Qadi, Formulation and implementation of a high-order 3-D domain integral method for the extraction of energy release rates, Comput. Mech. 49 (4) (2012) 459–476.
[57] J.-H. Kim, G.H. Paulino, T-stress, mixed-mode stress intensity factors, and crack initiation angles in functionally graded materials: a unified approach using the interaction integral method, Comput. Methods Appl. Mech. Engrg. 192 (11–12) (2003) 1463–1494.
[58] M.C. Walters, G.H. Paulino, R.H. Dodds Jr., Interaction integral procedures for 3-D curved cracks including surface tractions, Eng. Fract. Mech. 72 (11) (2005) 1635–1663.
[59] E.F. Rybicki, M.F. Kanninen, A finite element calculation of stress intensity factors by a modified crack closure integral, Eng. Fract. Mech. 9 (4) (1977) 931–938.
[60] F. Erdogan, G. Sih, On the crack extension in plates under plane loading and transverse shear, 1963.
[61] G.V. Guinea, J. Planas, M. Elices, KI evaluation by the displacement extrapolation technique, Eng. Fract. Mech. 66 (3) (2000) 243–255.
[62] W. Zhu, D. Smith, On the use of displacement extrapolation to obtain crack tip singular stresses and stress intensity factors, Eng. Fract. Mech. 51 (3) (1995) 391–400.
[63] S. Soghrati, A. Nagarajan, B. Liang, Conforming to interface structured adaptive mesh refinement: New technique for the automated modeling of materials with complex microstructures, Finite Elem. Anal. Des. 125 (2017) 24–40.
[64] A. Nagarajan, S. Soghrati, Conforming to interface structured adaptive mesh refinement: 3D algorithm and implementation, Comput. Mech. 62 (5) (2018) 1213–1238.
[65] M. Mohmadsalehi, A. Nagarajan, S. Soghrati, Higher-order mesh generation using CISAMR: A case study on bias in presentation and interpretation of results, Comput. Methods Appl. Mech. Engrg. 372 (2020) 113360.
[66] S.H. Song, G.H. Paulino, Dynamic stress intensity factors for homogeneous and smoothly heterogeneous materials using the interaction integral method, Int. J. Solids Struct. 43 (16) (2006) 4830–4866.

[67] P.-O. Bouchard, F. Bay, Y. Chastel, Numerical modelling of crack propagation: automatic remeshing and comparison of different criteria, Comput. Methods Appl. Mech. Engrg. 192 (35–36) (2003) 3887–3908.
[68] S. King, T. Richards, DS UK Ltd, Coventry March, 2013.
[69] J. Zhang, Z. Qu, W. Liu, L. Wang, Automated numerical simulation of the propagation of multiple cracks in a finite plane using the distributed dislocation method, C. R. MÉC. 347 (3) (2019) 191–206.
[70] T.L. Anderson, Fracture Mechanics: Fundamentals and Applications, CRC Press, 2017.
[71] D. Sutula, Energy minimising multi-crack growth in linear-elastic materials using the extended finite element method with application to smart-cuttm silicon wafer splitting, Cardiff Univ. 347 (1) (2016) 1–256.
[72] A.R. Ingraffea, M. Grigoriu, Probabilistic Fracture Mechanics: a Validation of Predictive Capability, Tech. rep., Cornell Univ Ithaca Ny Dept OF Structural Engineering, 1990.