Received: 14 July 2021 | Revised: 18 November 2021 | Accepted: 25 November 2021

DOI: 10.1002/nme.6911

RESEARCH ARTICLE

WILEY

Adaptive configurational force-based propagation for brittle and fatigue crack analysis

Robert Bird[1,2] | **William M. Coombs**[2] | **Stefano Giani**[2]

[1]Department of Earth Science and Engineering, Imperial College, London, UK

[2]Department of Engineering, Durham University, Durham, UK

Correspondence
Robert Bird, Department of Earth Science and Engineering, Imperial College, London, SW7 2BP, UK.
Email: r.bird@imperial.ac.uk

Funding information
Engineering and Physical Sciences Research Council, Grant/Award Number: EP/M507854/1; European Commission, Grant/Award Number: 734370

Abstract

This article presents a *hpr*-adaptive crack propagation method for highly accurate 2D crack propagation paths which requires no *a priori* knowledge of the tip solution. The propagation method is designed to be simple to implement, only *hr*-adaptivity is required, with the propagation step size independent of the initial mesh allowing users to obtain high fidelity crack path predictions for domains containing multiple cracks propagating at different rates. The proposed method also includes a crack path derefinement scheme, where elements away from the crack tip are derefined whilst elements close to the crack tip are small so capture the fidelity of the crack path. The result is that the propagation of cracks over an increasingly larger distances has negligible increased computational effort and effect on the propagation path prediction. The linear elastic problem is solved using the hp discontinuous Galerkin symmetric interior penalty finite element method, which is post-processed to obtain the configurational force at each tip to a user defined accuracy. Several numerical examples are used to demonstrate the accuracy, efficiency, and capability of the method. Due to the method's high accuracy crack path solutions of benchmark problems that are prolifically used in the literature are challenged.

KEYWORDS

configurational forces, crack propagation, discontinuous Galerkin, finite element analysis, *hp*-adaptivity, *r*-adaptivity

1 | INTRODUCTION

Many methods are presented in the literature to determine when and in what direction a crack should propagate. In this context, the notion of configurational (material) forces, (CF) presented by Eshelby,[1,2] for a measure of the forces induced by an elliptical inclusion or discontinuity within a homogeneous domain, and Rice[3] for the classical J-integral and its subsequent variations, are fundamental to these methods. CFs have seen widespread use in computational fracture mechanics, with Gurtin[4,5] and Maugin[6,7] being notable authors in the subject of CF based fracture, as well as general CF theory, and with local variational formulations, either for the use of explicit or implicit CF crack propagation, presented by Kienzler and Herrmann,[8] Gurtin and Podio-Guidugli,[5] and Steinmann and Maugin.[9] Within the context of non-enriched finite element methods the use of CFs to describe a moving fracture front was initially attempted by Mueller and Maugin[10] and later with the eXtended finite element method (XFEM) by Larsson and Fagerström.[11,12] Later a robust

r-adaptive* technique was defined by Miehe and co workers[13-15] for propagating cracks which was examined in Reference 16. Furthermore, the CF framework has recently been applied to materials with non-linear behavior, see for example the works of Runesson et al.[17] and Tillberg and Larsson[18] on elasto-plasticity and Näser et al.[19,20] on time-dependent materials, and the review by Özenç et al.[21]

An alternative to CF-driven crack propagation is the use of the path independent J-integral[3] to determine a crack's mode I and mode II stress intensity factors (SIFs).[22] Ishikawa et al.[23] demonstrated that the J-integral is a weighted sum of its mode I and mode II SIFs which can be separately determined by decomposing the stress and displacement field about the crack tip,[24] when these respective fields are known *a priori*. One of the main advantages of directly evaluating the CF is that no knowledge of the stress and displacement field is required *a priori*. The scalar value obtained by the J-integral has the same value as the component of the CF acting parallel to the crack face and, if the equation is known, the SIFs can be used to determine the CF vector. More commonly the SIFs are used with other methods, such as: the maximal principal stress criterion,[25] the maximum strain energy release rate criterion (MSERRC),[26] maximum circumferential stress criterion (MCSC)[25] (extended to anisotropic materials by Saomua et al.[27]), or the minimum strain energy density criterion (MSEDC).[28] With these methods working in conjunction with the "Gθ method,"[29] to determine the crack growth direction. An alternative method for linear elastic materials to determine the SIFs for pure and mixed mode crack problems, that does not use the J-integral, is the virtual crack extension method introduced independently by deLorenzi,[30] and by Parks et al.[31] and subsequent extensions. Within the context of XFEM,[32] presented a fixed-length crack extension algorithm to model stable, unstable and partially stable fracture fronts.

The majority of the papers outlined above adopt the conventional continuous Galerkin (CG) FEM as their underlying numerical tool. However, there are several advantages to adopting discontinuous Galerkin (DG) FEMs in terms of accuracy and ease of mesh adaptivity due to the weak inter-element connectivity, created by the broken Hilbert space.[33] Hansbo and Hansbo[34,35] present a crack propagation method, for linear and non-linear elasticity, using DG methods with[35] using the weak element interaction to track internal weak and strong discontinuities. However, similar to Heintz,[36] neither the weak interaction of element face communication element nor adaptivity was exploited for crack propagation, only later in Bird et al.,[37] with a geometry driven *rp*-adaptive crack CF propagation method, were these features exploited. More recently an *hr*-adaptive method was used with asynchronous spacetime DG methods for the introduction of discrete fractures from damage.[38] Bird et al.[39] produced a highly accurate domain independent method to determine the CF which combined the advantages of no knowledge *a priori* being required to compute the CF and, ease of error driven *hp*-adaptivity, using the DG FEM, to obtain highly accurate solutions of stress fields without the need for enrichment. Since no *a priori* knowledge was required to produce highly accurate results, the method can be applied directly to anisotropic materials, as shown in Reference 40, and has the potential to be also applied directly to heterogeneous materials. The XFEM has also been extended to DG methods by Shen et al.,[41] where optimal convergence results for the error in displacement and stress were presented. An alternative method to discrete fracture propagation is the hybrid DG method. The hybrid DG method exploits element interfaces and element specific degrees of freedom (DOF) to propagate a crack using a cohesive zone, initial works include.[42,43] This method has the advantage that it naturally allows the initiation of fractures however, it is strongly mesh dependent as the failure criterion is defined across element interfaces rather than nodes at the crack tip. Cracks can therefore only exist at the initialized boundaries of elements of the original mesh, unlike the *r*-adaptive method provided by Miehe et al.[13] where the element interfaces adapt and align with the predicted crack direction. Accurate solutions for crack propagation paths using the hybrid DG method can only be obtained with very refined meshes.[44] Hybrid DG methods have also been explored by Nguyen,[45] Radovitzky et al.,[44] Wu,[46] and Prechtel et al.,[47] amongst others. Within the context of cohesive law fracture, *h*- and *hp*-adaptive schemes have been produced for space-time DG methods.[48,49] A review of the cohesive laws which drive the traction-separation is provided by Park and Paulino.[50] There have also been recent developments using mesh adaptivity for crack propagation, for instance see Hussein et al.[51] where a virtual element technique is used to capture the crack path, and Reference 52 for a local mesh refinement and contraction technique. However, both these methods refine the mesh over the entirety of the crack path, the latter also uses a damage model to model the crack hence does not accurately capture the stress field at the crack tip.

The method proposed here is a *hpr*-adaptive method, including a refinement-derefinement scheme for the efficient propagation of multiple cracks through aligning the elements edges with the crack path. The adaptive strategy is two fold. First, the mesh is *hr*-adapted so that element edges align with the crack path with two regions of crack path resolution; high resolution near the crack tip and lower resolution elsewhere. Both the size of the high resolution region and the level of the resolutions are user defined and mesh independent. Second, error driven *hp*-adaptivity is performed to automatically resolve the CF to a user-defined accuracy. The method is efficient in that the added DOF required to capture the crack path are negligible. Other mesh refinement-derefinement adaptivity methods to capture crack paths also exist, as

shown by Gilbert et al.,[53] however there are several key differences. First Gilbert et al.'s method is applied to XFEM where the mesh is *h*-adapted so that the Heaviside function can capture the crack path rather than aligning element edges with the crack path. Second, this methods have no parameters to determine the desired resolution of the crack path. Third no error estimates are used to automatically resolve the mesh to achieve a desired accuracy, only predefined mesh fidelity around the crack tip is defined.

When reviewing the literature outlined above it is clear that, to date, researchers have failed to combine the following advantages of DG FEMs: straightforward *hp*-adaptivity to provide high fidelity solutions for modest computational effort achieving the accuracy of enriched methods, and the advantage of CF based fracture propagation, where no knowledge *a priori* is required of the local crack tip stress field. The lack of an enriched space does come at a cost of increased refinement, however when an appropriate error driven *hp*-adaptivity scheme is used, as by Bird et al.,[39] the same order of accuracy for the CF, or SIFs, to that achieved by XFEM solutions is possible for single and multiple crack problems for isotropic and the generally anisotropic materials.[39,40] In this article, these two advantages are combined to construct a highly accurate method for propagating cracks to drive fracture propagation in brittle materials providing a highly accurate yet computationally tractable fracture simulation framework without the use of enriched approaches.[†] This allows the developed framework to be applied to general material behavior without modifying the underlying algorithms. However, in this article we restrict the analysis to linear elastic materials undergoing small strains and focus on the development of the DG FEM *hpr*-adaptive fracture modeling framework.

2 | CONFIGURATIONAL FORCE FRACTURE PROPAGATION

This section provides the key equations for the small strain description of the CF approach for brittle fracture based on the work of Miehe et al.,[13] within which a full and thorough derivation can be found. The CF equations in the continuous spatial domain are presented here and subsequently extended to the discrete spatial FEM form in Section 4.1. The derivation of the CF starts with a statement of the global power dissipation postulate which is consistent with the second law of thermodynamics (see Reference 13). It concludes with the statement that the continuous form for the power dissipated by a propagating crack is

$$\mathscr{D} = \partial \boldsymbol{V} \cdot \boldsymbol{g}, \tag{1}$$

where $\partial \boldsymbol{V}$ is the crack tip material velocity with the crack tip CF, $\boldsymbol{g}$, defined as

$$\boldsymbol{g} = \lim_{|C|\to 0} \int_C \boldsymbol{\Sigma} \cdot \boldsymbol{n} \mathrm{d}s. \tag{2}$$

$\boldsymbol{\Sigma}$ is the Eshelby stress matrix defined as $\boldsymbol{\Sigma} = \boldsymbol{\delta}\psi(\boldsymbol{\varepsilon}) - \nabla \boldsymbol{u}^{\top}\boldsymbol{\sigma}$ with $\psi(\boldsymbol{\varepsilon})$ as the strain energy density function, $\boldsymbol{\sigma}$ is Cauchy stress, $\boldsymbol{\varepsilon}$ is small strain, $\boldsymbol{u}$ is displacement and $\boldsymbol{\delta}$ is the identity matrix. Armed with an equation for the CF at a crack tip, the next step is to determine how the crack will propagate. In this article a quasi-static crack propagation framework is employed as presented in References 13,14, a load release quasi-static algorithm. First it is necessary to integrate the discrete dissipation power at the crack, (1), over the time period $[t^n, t^{n+1}]$

$$\Delta\mathscr{D} = \int_{t^n}^{t^{n+1}} \mathscr{D}\, \mathrm{d}t \approx \Delta \boldsymbol{o} \cdot \boldsymbol{g}, \tag{3}$$

where $\Delta\mathscr{D}$ is the energy dissipated over a discrete propagation step. Equation (3) includes an incremental increase in the crack surface length, over the time period $[t^n, t^{n+1}]$, which is defined by the vector $\Delta\boldsymbol{o}$. This crack propagation vector for brittle fracture is expressed as

$$\Delta \boldsymbol{o} = \Delta a \frac{\boldsymbol{g}}{|\boldsymbol{g}|} \quad \text{where} \quad \Delta a = \begin{cases} h_o & \text{for } |\boldsymbol{g}| \geq g_c \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

where g_c is a Griffith material failure criteria, h_o is the step increase in crack length and for brittle crack propagation and Δa is subject to the Karush–Kuhn–Tucker conditions

$$\Delta a \geqslant 0, \quad (|\boldsymbol{g}| - g_c) \leqslant 0 \quad \text{and} \quad \Delta a(|\boldsymbol{g}| - g_c) = 0. \tag{5}$$

It should be highlighted that the motion of nodes is permitted in the material configuration, with the exception of motion that would change the shape of the boundary. It is recognized that is possible to dissipate power by moving nodes in the material configuration other than those at the crack tip,[54,55] and thus achieve a minimal energy solution to the problem. However this is a highly non-linear and computationally expensive problem. Therefore in this article we do not attempt to include this form of *r*-adaptivity to reduce the spurious CFs away from the crack tip. This is consistent with the works of References 13-15,34,35,56,57 and many others, but we do recognize that this form of *r*-adaptivity could potentially improve the solutions, albeit at considerable additional computational expense. Here, only power dissipation in the form of surface generation, or crack propagation, is included.

The key equations for modeling brittle fracture propagation based on CF have now been outlined. The crack tip CF value, for all crack tips, is calculated in a post-processing procedure once the displacement and stress field has been determined throughout the problem under consideration. It should also be stated that it is possible to simultaneously solve for the CF and material velocity as in Reference 16, based on the works of Reference 58. However, this makes the problem non-linear and inherently more difficult and expensive to solve and therefore we follow post-processing CF evaluation once the elasticity problem has been solved.

3 | DISCRETISATION: LINEAR ELASTICITY

In order to solve complex problems the evaluation of the continuous spatial domain description of CF at crack tips needs to be cast into a discrete framework. Here the DG symmetric interior penalty finite element method (SIPM) is used and described in Section 3.1, including the discretized form of the governing linear elastic equations. The corresponding error estimator and *hp*-adaptivity method are respectively provided in Sections 3.2 and 3.3.

3.1 | Symmetric interior penalty method (SIPM)

Let $\mathscr{B}_\Gamma$ be a bounded polygonal domain in $\mathbb{R}^2$ where the union of the Dirichlet $\partial\mathscr{B}_D$ and Neumann $\partial\mathscr{B}_N$ boundaries form the set of all boundaries $\partial\mathscr{B}_\Gamma = \partial\mathscr{B}_D \cup \partial\mathscr{B}_N$, with corresponding outward normal $\mathbf{n}$, where $\partial\mathscr{B}_D \cap \partial\mathscr{B}_N = \emptyset$. The strong form statement of equilibrium and the boundary conditions are defined as:

$$\begin{aligned} -\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}(\boldsymbol{u}) &= \mathbf{0} \quad \text{in} \quad \mathscr{B}_\Gamma, \\ \boldsymbol{u} &= \boldsymbol{u}_D \quad \text{on} \quad \partial\mathscr{B}_D, \quad \text{and} \\ \boldsymbol{\sigma}(\boldsymbol{u})\cdot\boldsymbol{n} &= \boldsymbol{\sigma}_N \quad \text{on} \quad \partial\mathscr{B}_N. \end{aligned} \tag{6}$$

$\boldsymbol{u} = (u, v)$ is displacement and $\boldsymbol{\sigma}(\boldsymbol{u})$ is the Cauchy stress matrix. $\boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{D} : \boldsymbol{\varepsilon}(\boldsymbol{u})$, where $\boldsymbol{D} \in R^{2\times2\times2\times2}$ is the material stiffness matrix and the small strain matrix $\boldsymbol{\varepsilon}$ is,

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) \\ \frac{1}{2}\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) & \frac{\partial v}{\partial y} \end{bmatrix}, \tag{7}$$

where (x, y) is the Cartesian coordinate system describing $\mathscr{B}_\Gamma$. $\boldsymbol{u}_D$ is the applied displacements on $\partial\mathscr{B}_D$ and $\boldsymbol{u}_N$ is the applied traction on $\partial\mathscr{B}_N$.

The domain $\mathscr{B}_\Gamma$ is subdivided by the mesh $\mathcal{T}$ consisting of elements K; $\mathcal{T}$ is in general irregular. $K \in \mathcal{T}$ is the image of the reference triangle under an affine elemental mapping $I_K : \widehat{K} \to K$. $\mathcal{F}(K)$ is the set of the elemental edges of an element K. If the intersection $F = \partial K \cap \partial K'$ of two elements $K, K' \in \mathcal{T}$ is a segment, F is an interior edge of $\mathcal{T}$ with the set of all interior edges denoted by $\mathcal{F}_I(\mathcal{T})$. Analogously, if the intersection $F = \partial K \cap \partial\mathscr{B}_\Gamma$ of an element $K \in \mathcal{T}$ and $\partial\mathscr{B}_\Gamma$ is a segment, we call F a boundary edge of $\mathcal{T}$. The set of all boundary edges of $\mathcal{T}$ is denoted by $\mathcal{F}_B(\mathcal{T})$ and it is the union of $\mathcal{F}_N(\mathcal{T})$ and $\mathcal{F}_D(\mathcal{T})$ of edges on the two boundaries $\partial\mathscr{B}_N$ and $\partial\mathscr{B}_D$. The set of all boundaries is denoted $\mathcal{F} = \mathcal{F}_B \cup \mathcal{F}_I$. For an intersection F, the two connected elements are denoted arbitrarily K^+ and K^- with their corresponding variables given the same + or − superscript. Given this notation, jumps and averages across element boundaries are defined as:

$$[\![\boldsymbol{w}]\!] = \mathbf{n}_K^+\boldsymbol{w}^+ - \mathbf{n}_K^+\boldsymbol{w}^- \quad \text{and} \quad \{\boldsymbol{\sigma}(\boldsymbol{w})\} = \frac{1}{2}\left(\boldsymbol{\sigma}(\boldsymbol{w})^+ + \boldsymbol{\sigma}(\boldsymbol{w})^-\right), \tag{8}$$

and on the boundary domain as,

$$[\![\boldsymbol{w}]\!] = \mathbf{n}_K^+ \boldsymbol{w}^+ \quad \text{and} \quad \{\boldsymbol{\sigma}(\boldsymbol{w})\} = \boldsymbol{\sigma}(\boldsymbol{w})^+. \tag{9}$$

For each element $K \in \mathcal{T}$, we define p_K to be the order of the element. We also define the vector $\underline{p} = \{p_K \ : \ K \in \mathcal{T}\}$ for elements in $\mathcal{T}$. For any mesh $\mathcal{T}$ of $\mathscr{B}_\Gamma$ with the degree vector $\underline{p}$, we then define the *hp*-version DG finite element space by

$$W_{\underline{p}}(\mathcal{T}) = \{\boldsymbol{w} \in [L^2(\mathscr{B}_\Gamma)]^2 \ : \ \boldsymbol{w}|_K \in [\mathcal{P}_{p_K}(K)]^2, \ K \in \mathcal{T}\},$$

which is achieved by using a hierarchical basis, of order p_K, over the reference triangle $\widehat{K}$ defined by Solin.[59]

The SIPM method in the bilinear form is introduced for the approximation of the model problem (6). Find $\boldsymbol{u}_h \in W_{\underline{p}}(\mathcal{T})$, such that,

$$a(\boldsymbol{u}_h, \boldsymbol{w}) = l(\boldsymbol{w}), \quad \forall \boldsymbol{w} \in W_{\underline{p}}(\mathcal{T}), \tag{10}$$

where the bilinear forms are,

$$\begin{aligned} a(\boldsymbol{u}, \boldsymbol{w}) := & \sum_{K \in \mathcal{T}} \int_K \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{w}) \, d\mathbf{x} \\ & - \sum_{F \in \mathcal{F}_I(\mathcal{T}) \cup \mathcal{F}_D(\mathcal{T})} \int_F \{\boldsymbol{\sigma}(\boldsymbol{u})\} \cdot [\![\boldsymbol{w}]\!] + \{\boldsymbol{\sigma}(\boldsymbol{w})\} \cdot [\![\boldsymbol{u}]\!] \, ds \\ & + \sum_{F \in \mathcal{F}_I(\mathcal{T}) \cup \mathcal{F}_D(\mathcal{T})} \frac{\beta p_F^2}{h_F} \int_F [\![\boldsymbol{u}]\!] \cdot [\![\boldsymbol{w}]\!] \, ds, \end{aligned} \tag{11}$$

$$l(\boldsymbol{w}) := - \sum_{F \in \mathcal{F}_D(\mathcal{T})} \int_F \mathbf{g}_D \cdot \boldsymbol{n} \cdot \{\boldsymbol{\sigma}(\boldsymbol{w})\} \, ds + \sum_{F \in \mathcal{F}_D(\mathcal{T})} \frac{\beta p_F^2}{h_F} \int_F \mathbf{u}_D \cdot \boldsymbol{w} \, ds - \sum_{F \in \mathcal{F}_N(\mathcal{T})} \int_F \boldsymbol{\sigma}_N \cdot \boldsymbol{w} \, ds. \tag{12}$$

β is the SIPM penalty constant, which, from the authors experience and from discussions in the literature on the penalty value,[60] has been chosen to have a value of 10 to ensure coercivity, and, the edge polynomial is described as,

$$p_F = \begin{cases} \max(p_K, p_{K'}), & \text{if on the internal edges,} \quad F = \partial K \cap \partial K' \in \mathcal{F}_I(\mathcal{T}), \\ p_K, & \text{if on the external edges,} \quad F = \partial K \cap \partial \mathscr{B}_\Gamma \in \mathcal{F}(\mathcal{T}) \setminus \mathcal{F}_I(\mathcal{T}), \end{cases} \tag{13}$$

and h_F is the length of the edge F, where for an internal edge, h_F is the length of the coincident element edge that is shared between K and K'.

In SIPM the DOF are element specific and instead of sharing DOF at nodes, as in continuous Galerkin methods, the elements interact through three edge integral terms in Equation (11). The first edge term averages the tractions acting between elements so the same traction acts on the elements sharing an interface. The second term symmetrizes the global stiffness matrix which ensures optimal convergence for the given polynomial order of the elements in the mesh. The last term penalizes displacements between elements to stabilize the method. Since elements interact through edge integrals, implementing adjacent elements with relative differences in polynomial order and edge length, such that the mesh is non-conforming, is straightforward and does not require any special treatment. This feature of DG methods is useful for *hp*-adaptive methods where element size and polynomial order will vary between adjacent elements. However, before *hp*-adaptivity can be applied, an error estimate is required to determine the relative error in each element through the physical problem being analyzed.

3.2 | Error estimate

The derivation of an *hp a posteriori* residual based error estimate for SIPM is complex and is not the main focus of this article. The derivation is therefore not included and only a description of the role of each term is provided here; a complete

derivation and proof can be found in Reference 61. The error estimate η, is both reliable and efficient for the error in the DG energy norm, that is

$$c_\eta \eta \leq |||\boldsymbol{u} - \boldsymbol{u}_h|||_{\mathcal{T}} \leq C_\eta \eta, \tag{14}$$

where C_η and c_η are two positive constants independent of the element size, element polynomial order, and magnitude of the loading applied on the boundary and, $\boldsymbol{u}$ is the true displacement solution. For reliability and efficiency to hold for a problem during *hp*-adaptivity, the following conditions must be met;

1. the jump in polynomial order between a neighboring pair of elements p_K and $p_{K'}$ must be bound by arbitrary constant, we choose $|p_K - p_{K'}| \leq 1$; and
2. the number of hanging nodes per element face has to be bound, in this implementation we only allow one hanging node per element edge.

The limit on the number of hanging nodes on an element edge has been set to one, however it could be set to an arbitrarily high number at a cost of increasing data structure complexity. When the initial mesh is contiguous and only homogeneous *h*-refinement is performed during a *hp*-adaptive step, there is little-to-no advantage in allowing more hanging nodes per element edge when the non-regular part of the solution is a singularity at a point, such as a crack tip. This is because the elements at the singularity are the most likely to be chosen for *h*-adaptivity. Therefore, the elements connected to the singularity are unlikely to be associated with any hanging nodes. At the next *hp*-refinement step the new crack tip elements are chosen for *h*-refinement and since they are not associated with any hanging nodes, *h*-refining these elements will only create edges with a maximum of 1 hanging node. The newly elements that exist at the crack tip will again not be associate with any hanging nodes. The result is the element size at the singularity will decrease exponentially with the number of degrees of freedom (NDOF) and the *h*-refinement will be localized only to the singularity; where it is necessary. This results in exponential convergence of the error with DOF for problems containing non-regular parts in the solution.[61,62]

The DG norm for linear elasticity is

$$|||\boldsymbol{u}|||_{\mathcal{T}} := \left(\sum_{K \in \mathcal{T}} \|\boldsymbol{\nabla u}\|^2_{0,K} + \sum_{F \in \mathcal{F}^{\text{int}}(\mathcal{T})} \frac{\gamma p_F^2}{h_F} \|[\![\boldsymbol{u}]\!]\|^2_{0,F} + \sum_{F \in \mathcal{F}^{\text{D}}(\mathcal{T})} \frac{\gamma p_F^2}{h_F} \|\boldsymbol{u}\|^2_{0,F} \right)^{1/2}, \tag{15}$$

When η is combined with a suitable *hp*-adaptive scheme, described in Section 3.3, exponential convergence of the error estimated value and the true error can be achieved; since η is both reliable and efficient for the error in the DG norm (14). The global error estimate η is defined as,

$$\eta = \sqrt{\sum_{K \in \mathcal{T}} \left(\eta^2_{R,K} + \eta^2_{J,K} + \eta^2_{F,K} \right)}, \tag{16}$$

and where the single element error estimate is $\eta^2_K = \eta^2_{R,K} + \eta^2_{J,K} + \eta^2_{F,K}$. The first component is an area integral defined as

$$\eta^2_{R,K} = \frac{h_K^2}{p_K^2} \|\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u}_h)\|^2_{0,K}, \tag{17}$$

where h_K is the inner diameter of element K. This term measures how well the strong form equilibrium equation (the first term in Equation (6)) has been satisfied by the weak finite element formulation.

As DG methods do not satisfy C^0 across elements, jumps in displacement exist between elements on the segment $F \in \mathcal{F}_I(\mathcal{T})$. As the Dirichlet boundary conditions are also imposed weakly jumps in the displacement boundary condition will also always exist between the displacement solution on the element boundary and the boundary condition imposed on them. Since the true solution for displacement should be continuous across these boundaries, the error in the jump in displacement on the edge $F \in \mathcal{F}_I(\mathcal{T})$ and the Dirichlet boundary $F \in \mathcal{F}_D(\mathcal{T})$ are measured as

$$\eta^2_{J,K} = \frac{1}{2} \sum_{F \in \mathcal{F}_I} \frac{\gamma^2 p_F^3}{h_F} \|[\![\boldsymbol{u}_h]\!]\|^2_{0,F} + \sum_{F \in \mathcal{F}_D} \frac{\gamma^2 p_F^3}{h_F} \|\boldsymbol{u}_h - \mathbf{g}_D\|^2_{0,F}. \tag{18}$$

Like most FEMs, SIPM does not satisfy C^1 across the interior edges $F \in \mathcal{F}_I(\mathcal{T})$ and the Neumann boundary $F \in \mathcal{F}_N(\mathcal{T})$. These error measures are as

$$\eta_{F,K}^2 = \frac{1}{2}\sum_{F\in\mathcal{F}_I}\frac{h_F}{p_F}\|[\![\boldsymbol{\sigma}(\boldsymbol{u}_h)]\!]\|_{0,F}^2 + \sum_{F\in\mathcal{F}_N}\frac{h_F}{p_F}\|\boldsymbol{\sigma}(\boldsymbol{u}_h)\cdot\mathbf{n} - \mathbf{g}_N\|_{0,F}^2. \tag{19}$$

The above terms allow the computation of an estimate of the error on each element in the finite element mesh. These errors can then be used to drive h and p adaptivity, as described in Section 3.3.

3.3 | *hp*-adaptivity

The steps involved in the *hp*-adaptive process are given in Algorithm 1 and is based on the adaptivity algorithm presented in Reference 63. This strategy has been shown to be proficient for finite elements in Reference 62 and achieved exponential convergence for both smooth and non-smooth problems linear elastic problem modeled using SIPM in Reference 61. The adaptive process is controlled by two parameters δ_2 and δ_1, where $\delta_2 \geq \delta_1$ and $\delta_2, \delta_1 \in [0, 1]$.

Elements that have an error such that $\eta_K^2 > \delta_2\eta_{\max}^2$, where $\eta_{\max}^2$ is the maximum error in any element, are refined in h as it is assumed that the regions of high error are associated with non-smooth regions of the solution. This assumption is particularly appropriate for problems involving fractures where the largest errors occur in regions close of crack tips. Elements with errors that fall in the range $\delta_2\eta_{\max}^2 > \eta_K^2 > \delta_1\eta_{\max}^2$ are assumed to be modeling a smooth portion of the problem, but the polynomial order is not sufficiently high, so these elements are refined in p. The values of δ_2 and δ_1 are problem dependent and typical values for problems involving fractures and detailed as part of the numerical examples. This adaptivity scheme was shown in Bird et al.[39] to be effective for problems containing multiple cracks; h-adaptivity always occurred at all crack tips. Additionally p-adaptivity is generally observed away from the crack tip, but some p-adaptivity will occur at the crack tip. This is consistent with observations on the performance of *hp*-adaptivity schemes for elliptic problems containing point singularities.[61,62]

When a triangular element is refined in h it is refined homogeneously, as in Figure 1. The parent element is split into four new child triangle elements, all of which are similar to their parent and inherit the material properties and polynomial order of their parent. Refining a triangle in p simply amounts to increasing p_K by one for $K \in \mathcal{T}_{p\_ref}$ and updating the vector $\overline{p}$. These steps are followed by a smoothing algorithm that enforces a maximum of one hanging node per element edge and a maximum jump in polynomial of 1 between elements. If the jump in element order between adjacent elements is found to be greater than 1, the polynomial order of the lower order element is increased by 1.

During an *hp*-adaptive step only h- and p-refinement occurs, there is no mesh derefinement. The strategy of only refining the mesh is shown to achieve exponential convergence for the error for multi-crack static problems by Bird et al.[39] It could be possible to increase the convergence rate with a suitable derefinement scheme but it not considered during a *hp*-adaptive step. Mesh derefinement is however entirely necessary when the domain boundaries or boundary conditions evolve with quasi-static time, such as a crack propagation. The mesh derefinement used in this work is described in Section 5.1 during the *hr*-adaptive step.

Algorithm 1. hp-refinement strategy: *For parameters* δ_1, δ_2 *with* $1 \geq \delta_2 \geq \delta_1 \geq 0$.

1: Compute the maximum error $\eta_{\max}^2 = \max_{K\in\mathcal{T}}\eta_K^2$
2: Identify the set of elements to refine in p: $\mathcal{T}_{p\_ref} = \{K \in \mathcal{T} \,|\, \delta_2\eta_{\max}^2 \geq \eta_K^2 > \delta_1\eta_{\max}^2\}$
3: Increase p_K by one for $K \in \mathcal{T}_{p\_ref}$
4: Identify the set of elements to refine in h: $\mathcal{T}_{h\_ref} = \{K \in \mathcal{T} \,|\, \eta_K^2 > \delta_2\eta_{\max}^2\}$
5: Identify any elements $K \in \mathcal{T}'$, where $\mathcal{T}'$ is the refined mesh, that will have more than one hanging node on a face and add to $\mathcal{T}_{href}$
6: h refine all elements $K \in \mathcal{T}_{h\_ref}$ to create the new mesh $\mathcal{T}'$
7: Ensure for every pair of neighbors $K, K' \in \mathcal{T}'$ that only there only exists a maximum of a polynomial order difference of 1, otherwise add K', where $p_{K'} < p_K$, to the set $\mathcal{T}'_{p\_ref}$
8: Increase p_K by one for $K \in \mathcal{T}'_{p\_ref}$

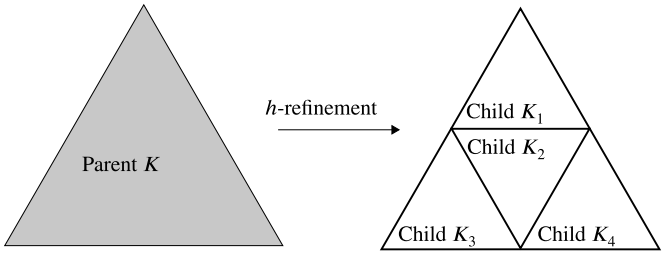


FIGURE 1 Homogeneous h-refinement

4 | CONFIGURATIONAL FORCE CALCULATION AND ERROR ESTIMATION

As outlined in the Introduction, the works of Eshelby,[1,2] Rice,[3] and Irwin,[22,64] are fundamental to derivations that calculate the CF at crack tips. Within the framework of finite elements, Miehe et al.[13,14] present a derivation, for a continuous problem, for the energy released by a crack and the associated CF value at the crack tip. The result of the continuous derivation was then linked to a discrete FE formulation to show that reasonable crack propagation paths can be predicted when only using the CF value at the crack tip.[‡] An alternative approach is presented by Denzer et al.,[65] named a *domain* formulation, where the nodal values of CF are summed in a domain around the crack tip and this combined value of CF is used to determine the energy released by a crack. However, the formulations presented by Denzer et al.,[65] Miehe et al.,[13] and Miehe and Gurses[14] do not consider the jump in energy across the crack faces. This means that the methods are flawed from problems where this term is non zero—basically any problem that is non-symmetric in terms of energy either side of the crack faces. The result of this oversight is that these methods: (i) do not converge with mesh refinement and/or (ii) the mixed mode component of the CF is highly domain size dependent, as shown in Reference 39. The works of References 23,56,66,67 do include the jump in energy term across the crack faces directly, or indirectly, in the CF computation. However, they all have one the following shortcomings: (i) consider mixed mode notched cracks and as such the true stress solution on the crack edges at the crack tip is in the finite element solution space;[56] (ii) use *a priori* knowledge of the stress solution at the crack tip combined with a partial integration of the crack faces to determine the mixed mode component of the CF;[67] or (iii) compute the CF component that acts perpendicular to the crack tip using an algorithm that combines the computed CF component that acts parallel to the crack faces with *a priori* knowledge of the stress solution at the crack tip.[23,66] This limits the applicability of the methods and therefore in this work we follow a different approach.

4.1 | Configurational force

As given in Equation (2), and repeated here for readability, the continuous form of the CF acting at the crack tip and the corresponding dissipated power is

$$\boldsymbol{g} = \lim_{|C|\to 0} \int_C \boldsymbol{\Sigma}\cdot\boldsymbol{n}\mathrm{d}s \qquad \text{and} \qquad \mathscr{D} = \partial\boldsymbol{V}\cdot\boldsymbol{g}. \tag{20}$$

The equation for the CF in Equation (20) is not in a practical form for discrete analysis due to $\lim_{|C|\to 0}$. The derivations presented by Li et al.,[68] Raju and Shivakumar,[66] and Heintz et al.[56] overcome this issue by reformulating the infinitesimal integral in (20) into a finite domain integral with a surface term that can be used in conjunction with a FEM. In this article we adopt the discrete formulation defined in Reference 39 for the dissipation associated with a growing fracture

$$\mathscr{D}_h = \boldsymbol{l}V\boldsymbol{G} = \boldsymbol{l}\sum_{K\in A}\int_K \boldsymbol{\nabla}V\boldsymbol{\Sigma}^h d\mathbf{x} + \boldsymbol{l}\sum_{\partial K\in(\Gamma^+\cup\Gamma^-)\setminus R}\int_{\partial K} V\psi^h(\boldsymbol{\varepsilon})\boldsymbol{n}ds, \tag{21}$$

where the limits of the summations and integrals are shown in Figure 2. The domain A is bound by the two crack faces $\Gamma^+\cup\Gamma^-$ and the outer edge away from the crack faces is O. V is a smooth scalar function which has a value of 1 at the crack tip and 0 at the domain boundary O, (as shown in Figure 2), and $\boldsymbol{l}$ is the velocity vector of the crack tip. The Eshelby

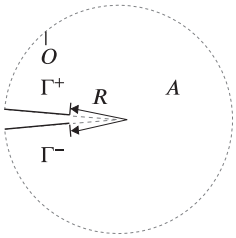


FIGURE 2 The integral domain around a crack tip

stress $\boldsymbol{\Sigma}^h$ and the strain energy ψ^h are computed using the finite element displacement solution. The region R is a subset of the crack face $R \subseteq \Gamma^+ \cup \Gamma^-$ at the crack tip and is equally shared across the two faces such that $|R|/2$ exists on Γ^+ and $|R|/2$ exists on Γ^-. It is important to highlight here that Γ^+ and Γ^- refer to the entire crack faces within the integral region A. The region R is excluded from the integral due to the spurious oscillations that are induced in the energy term, $\psi^h(\boldsymbol{\varepsilon})$, along the crack faces near the crack tip.[39] To obtain highly accurate solutions the formulation (21) is computed using an *hp*-adaptive algorithm that considers small values of R that reduce with h refinement at the crack tip. The algorithm is governed by three error estimates, an error measure for the area component defined in Reference 39, an error measure for the surface integral and a error measure for not including the integral region R. The two latter error measure are derived and explained in the following section.

4.2 | Reliability of an error estimator for the configurational force calculation

The global error estimate *a posteriori* η, (16), was shown, up to an arbitrary constant C_η, in Reference 61 to bound the true error of the solution in the DG norm (14). A proof, and numerical demonstration, was later provided in Reference 39 that for the subdomain $A \subset \mathcal{T}$, η bound the error in the area integral component of the CF calculation (21) up to an up to an arbitrary constant C_A,

$$\|\boldsymbol{\nabla} V \cdot \boldsymbol{\Sigma}^h - \boldsymbol{\nabla} V \cdot \boldsymbol{\Sigma}\|_{0,K\in A} \leq \|\boldsymbol{\nabla} V\|_{0,K\in A} \|\boldsymbol{\Sigma}^h - \boldsymbol{\Sigma}\|_{0,K\in A} \leq C_A \|\boldsymbol{\nabla} V\|_{0,\mathcal{T}} \eta_A^2 = \|\boldsymbol{\nabla} V\|_{0,\mathcal{T}} C_A \beta_A. \tag{22}$$

The line integral component of (21), not including the region R, can also be bound by the error estimate η by considering a trace inequality. First an inequality is defined for an element edge integral component of (21), which separates V from the other terms in the integral

$$\int_{\partial K} V \cdot \psi^h(\boldsymbol{\varepsilon}) \cdot \boldsymbol{n} ds \leq \int_{\partial K} V ds \int_{\partial K} \psi^h(\boldsymbol{\varepsilon}) \cdot \boldsymbol{n} ds, \tag{23}$$

where $\psi^h(\boldsymbol{\varepsilon}) > 0$ and $V > 0$ are always true. The function V is known as it is defined by the user, and is unchanging during mesh refinement, it therefore can be computed directly once. Hence, only an error estimate for the integral over the energy component of the integral is required. In order to bound the error in energy component the divergence theorem is used to state the following

$$\int_{\partial\Omega} \psi^2 \cdot \boldsymbol{x} \cdot \boldsymbol{n} = \int_{\Omega} \boldsymbol{\nabla} \cdot (\psi^2 \cdot \boldsymbol{x}), \tag{24}$$

for when the strain energy density is $\psi \in H_n^1$ and $n \geq 0$, Ω is a star shaped domain with boundary $\partial\Omega$, and $\boldsymbol{x} \in \Omega$ is a coordinate. By the definition of Ω being a star shaped domain, such as a triangle, the following statement is true

$$\zeta |\boldsymbol{x}| \leq \boldsymbol{x} \cdot \boldsymbol{n}. \tag{25}$$

where $\zeta > 0$ and $\boldsymbol{n}$ is the outward normal to $\partial\Omega$. By substituting (25) into (24), the following inequality can be defined,

$$\int_{\partial\Omega} \psi^2 \cdot \boldsymbol{x} \cdot \boldsymbol{n} \geq \zeta \inf_{\boldsymbol{x}\in\partial\Omega} |\boldsymbol{x}| \int_{\partial\Omega} \psi^2 = \zeta \inf_{\boldsymbol{x}\in\partial\Omega} |\boldsymbol{x}| \cdot \|\psi\|_{0,\partial\Omega}, \tag{26}$$

The left hand side of Equation (26) can be expanded so that the inequality becomes more useful for computing error terms,

$$\begin{aligned}\zeta \inf_{\boldsymbol{x}\in\partial\Omega}|\boldsymbol{x}|\cdot\|\psi\|_{0,\partial\Omega} &\leq \int_{\Omega}\boldsymbol{\nabla}\cdot(\psi^2\cdot\boldsymbol{x})\\ &= \int_{\Omega}\psi^2\cdot\boldsymbol{\nabla}\boldsymbol{x}+\int_{\Omega}\boldsymbol{\nabla}\psi^2\cdot\boldsymbol{x}\\ &= 2\|\psi\|^2_{0,\Omega}+\int_{\Omega}\boldsymbol{\nabla}\psi^2\cdot\boldsymbol{x}\\ &\leq 2\|\psi\|^2_{0,\Omega}+\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\int_{\Omega}\boldsymbol{\nabla}\psi^2\\ &= 2\|\psi\|^2_{0,\Omega}+2\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\int_{\Omega}\psi\cdot\boldsymbol{\nabla}\psi\\ &\leq 2\|\psi\|^2_{0,\Omega}+2\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\int_{\Omega}|\psi|\cdot|\boldsymbol{\nabla}\psi|\\ &\leq 2\|\psi\|^2_{0,\Omega}+2\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\|\psi\|_{0,\Omega}\cdot\int_{\Omega}|\boldsymbol{\nabla}\psi|.\end{aligned} \tag{27}$$

Using the definition of the gradient of the strain energy density function

$$|\boldsymbol{\nabla}\psi| = \left|\frac{\partial\psi}{\partial\boldsymbol{u}}\cdot\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{x}}\right| \leq \left|\frac{\partial\psi}{\partial\boldsymbol{u}}\right|\cdot\left|\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{x}}\right| \leq |\boldsymbol{\nabla}\boldsymbol{\sigma}|\cdot\left|\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{x}}\right| \leq C\cdot|\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}|\cdot|\varepsilon|. \tag{28}$$

The inequality (27) can be further expressed as

$$\begin{aligned}\zeta \inf_{\boldsymbol{x}\in\partial\Omega}|\boldsymbol{x}|\cdot\|\psi\|_{0,\partial\Omega} &\leq 2\|\psi\|^2_{0,\Omega}+2\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\|\psi\|_{0,\Omega}\cdot\int_{\Omega}|\boldsymbol{\nabla}\psi|\\ &\leq 2\|\psi\|^2_{0,\Omega}+2C\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\|\psi\|_{0,\Omega}\cdot\int_{\Omega}|\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}|\cdot|\boldsymbol{\varepsilon}|\\ &\leq 2\|\psi\|^2_{0,\Omega}+2C\sup_{\boldsymbol{x}\in\Omega}|\boldsymbol{x}|\cdot\|\psi\|_{0,\Omega}\cdot\|\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}\|_{0,\Omega}\cdot\|\boldsymbol{\varepsilon}\|_{0,\Omega}.\end{aligned} \tag{29}$$

Currently the domain provided by Ω is a general star-shaped however, in the meshes used here only triangles exist, and so the last inequality statement in (29) can be made more specific to triangles through the statements

$$\begin{aligned}&\sup_{\boldsymbol{x}\in K}|\boldsymbol{x}| \leq h_K\\ &\inf_{\boldsymbol{x}\in\partial K}|\boldsymbol{x}| \geq \rho_K \geq h_K/r_K\end{aligned}, \tag{30}$$

where ρ_K is a radius of the circle that encompasses the triangle, r_K is a measure of the triangle's shape regularity, and the origin of coordinates $\boldsymbol{x}$ corresponds to the center of the triangle.[69] Substituting (30_1) and (30_2) into (29) and, assuming that the triangles in the mesh are sufficiently regular such that $r_K \leq D\ \forall K$ and $\zeta \leq E\ \forall K$, where D are E are positive constants, (29) becomes

$$\|\psi\|^2_{0,\partial K} \leq \frac{C}{h_K}\cdot\left(\|\psi\|^2_{0,K}+h_K\cdot\|\psi\|_{0,K}\cdot\|\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}\|_{0,K}\cdot\|\boldsymbol{\varepsilon}\|_{0,K}\right), \tag{31}$$

where C is a positive constant. Inspecting the line integral term of the CF calculation (21), the corresponding error can be written as

$$\|V\cdot\psi\cdot\boldsymbol{n}-V\cdot\psi^h\cdot\boldsymbol{n}\|_{0,\partial K} \leq \|V\cdot\psi-V\cdot\psi_h\|_{0,\partial K} \leq \|V\|_{0,\partial K}\cdot\|\psi-\psi^h\|_{0,\partial K}. \tag{32}$$

This gives the result that the error in the calculation of the last term can be found by inspection of (31) to give

$$\|\psi-\psi_h\|^2_{0,\partial K} \leq \frac{C}{h_K}\cdot\left(\|\psi-\psi^h\|^2_{0,K}+h_K\cdot\|\psi-\psi_h\|_{0,K}\cdot\|\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}-\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^h\|_{0,K}\cdot\|\boldsymbol{\varepsilon}-\boldsymbol{\varepsilon}^h\|_{0,K}\right), \tag{33}$$

which using the definitions

$$\begin{aligned} \|\nabla \cdot \boldsymbol{\sigma} - \nabla \cdot \boldsymbol{\sigma}^h\|_{0,K} &= \|\nabla \cdot \boldsymbol{\sigma}^h\|_{0,K} \\ \|\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^h\|_{0,K} &\leq \frac{1}{|\boldsymbol{D}|} \cdot \eta_K \\ \|\psi - \psi_h\|_{0,K} &\leq \frac{C}{|\boldsymbol{D}|} \cdot \eta_K^2 \end{aligned} \quad , \tag{34}$$

becomes,

$$\|\psi - \psi^h\|_{0,\partial K}^2 \leq \frac{C}{h_K} \left(\frac{1}{|\boldsymbol{D}|^2} \cdot \eta_K^4 + \frac{h_K}{|\boldsymbol{D}|^2} \cdot \eta_K^3 \cdot \|\nabla \cdot \boldsymbol{\sigma}^h\|_{0,K} \right). \tag{35}$$

The above equation bounds the error in the energy norm squared for a single element edge ∂K. It is used to estimate the error in edge contribution to the CF calculation in (21) by considering all element edges $\partial K \in (\Gamma^+ \cup \Gamma^-)$. The total error estimate for the edge energy calculation from these element edges is therefore

$$\begin{aligned} \|\psi - \psi^h\|_\tau &< C_\tau \cdot \sqrt{\sum_{\partial K \in \tau} \left(\frac{1}{h_K} \left[\cdot \frac{1}{|\boldsymbol{D}|^2} \cdot \eta_K^4 + \frac{h_K}{|\boldsymbol{D}|^2} \cdot \eta_K^3 \cdot \|\nabla_{\boldsymbol{j}} \cdot \boldsymbol{\sigma}^h\|_{0,K} \right] \right)} \\ &= C_\tau \cdot \chi_\tau \end{aligned} \quad , \tag{36}$$

where $\tau = (\Gamma^+ \cup \Gamma^-) \setminus R$ and C_τ is an arbitrary constant for the length τ. The error in the CF calculation, Equation (20), can now be bound by two error measures. The area integral component is bound by the error estimate β_A (22) and the edge integral component is bound by χ_τ (36), this can be summarized as

$$\sum_{K \in A} \int_K \nabla V \cdot (\boldsymbol{\Sigma}^h - \boldsymbol{\Sigma}) d\mathbf{x} \lesssim \beta_A \qquad \text{and} \qquad \sum_{\partial K \in \tau} \int_{\partial K} V \cdot (\psi^h - \psi) \cdot \boldsymbol{n} ds \lesssim \chi_\tau. \tag{37}$$

The values β_A and χ_τ are used in the later sections as stopping criteria to determine when the computation of the configuration force has reached a sufficient accuracy.

4.3 | Configurational force calculation and stopping criteria

It was shown in Reference 39 that Equation (21) can achieve highly accurate predictions for the CF by forcing $R \to 0$ with *hp*-adaptivity. Since the CF method used here does not use any crack tip enrichment functions, there will be significant refinement in element size and polynomial order around the crack tip.[39] The method is different to the domain integral M-integral method proposed by Yau et al.[70] which uses an auxiliary stress function which requires *a priori* knowledge of the stress field around the crack tip which is material dependent. The advantage of the method proposed in this article is that since no *a priori* knowledge of the stress field around the crack tip is required it can be applied directly to any material type and achieve highly accurate results when compared to enriched and/or the interaction integral.[39,40] However, since (21) is being used to propagate a crack, a stopping criterion is required to:

1. determine when to stop reducing $|R|$ to 0; and
2. determine when the calculation of the CF, for a fixed R, is sufficiently accurate such that *hp*-refinement can be stopped and the crack can be propagated.

Therefore an alternative algorithm is required here, compared to the case of static cracks which was the focus of Reference 39. The stopping criteria is governed by three error estimates for the CF calculation:

1. estimated error β_A for the area integral component, (37);
2. estimated error χ_τ for the edge integral component, (37); and
3. the error associated with excluding the region R in the edge integral.

These three error measures are combined with a user specified parameter $\Theta \in (0, 1)$ which together define the stopping criterion: Stop *hp*-adaptivity if, for all cracks, the statements

$$\Theta > \frac{\beta_A^m}{\beta_A^1}, \quad \Theta > \frac{\chi_\tau^m}{\chi_\tau^{m_R}} \quad \text{and} \quad \Theta > \frac{|R|}{|\Gamma^+ \cup \Gamma^-|}, \tag{38}$$

are true. Here, m is the current *hp*-refinement step number and m_R is the first step number when $|R| < \Theta|\Gamma^+ \cup \Gamma^-|$ is true. Unlike the method presented in Reference 39, where $|R| \to 0$ with *hp*-adaptivity, only a specified accuracy is required with Θ and once the relation $|R| < \Theta|\Gamma^+ \cup \Gamma^-|$ is true the reduction of $|R|$ with *hp*-adaptivity is stopped. The criteria in Equation (38) are based on error estimates normalized at a step m to a previous step to remove the unknown constant associated with (37). The second criterion is normalized with respect to m_R since at this step R, and the domain size of the CF line integral component, are no longer changing and thus the constant associated with the inequality for χ_τ in (37) is the same for all subsequent refinement steps. The error estimate β_A for the area integral is normalized with respect to step 1 since the domain size, and therefore inequality constant, in (37) does not change. The definition of the first two criterion is based on the assumption that when the first estimated error calculations are performed the error estimates correspond to actual errors of less than 100%. The result is that the ratios β_A^m/β_A^1 and $\chi_\tau^m/\chi_\tau^{m_R}$ for the *hp*-adaptive step m estimate an error greater than the actual error.

The third criterion is based on the argument that the closer the edge integral gets to the crack tip, the smaller its contribution to the CF calculation. This is because near the crack tip the energy along each crack face approaches infinity however conversely, the energy difference across the crack faces is finite, otherwise the configuration force will have an infinite value, which is impossible. Therefore, the smaller Θ becomes the more the integral of the energy difference along the ignored region $|R|$ tends to zero. If Θ is sufficiently small, it is an upper bound to the true error, the smaller the value of Θ the more accurate the edge integral and the more certainty Θ is an upper bound to the actual error in excluding the region R in the edge integral.

The method for reducing $|R| \to 0$ with *hp*-adaptivity is described fully in Algorithm 2. Initially, and until $R < \Theta|\Gamma^+ \cup \Gamma^-|$, R is taken to be the portion of the crack edges which correspond to the two element edges, respectively on Γ^+ and Γ^-, that lie on the crack tip, as shown in Figure 3A. During *hp*-adaptivity the elements at the crack tip are *h*-refined and these edges are reduce in length, which also reduces $|R|$. When R is reduced sufficiently that $|R| < \Theta|\Gamma^+ \cup \Gamma^-|$, as in Figure 3B, $|R|$ is no longer reduced, the length $\tau = |R \setminus (\Gamma^+ \cup \Gamma^-)|$ is calculated and the *hp*-adaptive step when this condition met is stored $m_R = m$. The significance of this is that the length of the crack edges for the CF calculation is now defined with τ and as τ is no longer changing, the edge contribution to the CF and the corresponding error can be computed. In other words, since $|R| < \Theta|\Gamma^+ \cup \Gamma^-|$ is true, the edge estimated error for the step m_R, $\chi_\tau^{m_R}$, and steps $m > m_R$, χ_τ^m, can now occur.

The algorithm for computing the CF for a single crack, with the stopping criteria, is provided in Algorithm 2. For the sake of readability the error parameters β, χ, m_R, the regions, τ, Γ, and R, and, an edge flag, are not defined for a specific crack although, each crack will have individual values of these quantities.

5 | NUMERICAL INGREDIENTS FOR ADAPTIVE FRACTURE PROPAGATION

This section describes the numerical tools that are required for *hpr*-adaptive fracture propagation. The *hp*-adaptive component was described in Section 4.3 and is used for the accurate computation of the CF at the crack tip. The *hr*-adaptive component is described in Section 5.1 and describes and how element faces are aligned and adapted (*hr*-adaptivity) to facilitate fracture propagation. The propagation method described in Section 5.1 relies upon the availability of homogeneous *h*-refinement, as described in Section 3.3. The overall crack propagation algorithm is split three main processes:

1. *hp*-adapt the mesh using the error estimate defined in Section 3.2 until the stopping criteria (38) is met;
2. in a post processing calculation determine the CF at all crack tips and subsequently predict the crack propagation direction and increment (as described in Section 4.3); and
3. use the CF values calculated in the previous step to propagate the cracks in the mesh. An *hr*-adaptive method is used to capture the crack path and crack tip within the initial mesh stencil via face alignment and splitting coupled with

Algorithm 2. Static CF calculation for a all cracks in the mesh

$m \to m+1$
Determine the displacement $\boldsymbol{u}^h$ and the corresponding estimated error η_K^2, for all elements K in the mesh $\mathcal{T}$
while Stopping criteria is not met **do**
  $m \to m+1$
  Determine the displacement $\boldsymbol{u}^h$ and the corresponding estimated error η_K^2, for all elements K in the mesh $\mathcal{T}$
  for all cracks **do**
    if $m == 1$ **then**
      Calculate β_A^1
    else
      Calculate β_A^m
    end if
    if edge flag== 0 **then**
      Compute $|R|$ as the element edges at the crack tip, Figure 3A
    end if
    if $|R| < \Theta|\Gamma^+ \cup \Gamma^-|$ **then**
      if edge flag== 0 **then**
        $m_R = m$
        Define the length the proportion of the crack faces $\tau = (\Gamma^+ \cup \Gamma^-) \setminus R$
        Compute $\chi_\tau^{m_R}$
        Set the edge flag= 1 for the current crack
      else
        Compute χ_τ^m
      end if
    end if
  end for
  if For all cracks $\Theta > \beta_A^m/\beta_A^1, \quad \Theta > \chi_\tau^m/\chi_\tau^{m_R}$ and $\Theta > |R|/|\Gamma^+ \cup \Gamma^-|$ **then**
    Stopping criteria is met
    Compute the configuration force using Equation (21) for all cracks.
  else
    hp-adapt the mesh
  end if
end while

h-adaptivity, which may be required based on the position of the crack tip within the mesh and/or when *r*-adaptivity creates poor quality elements.

The first two steps were described in Section 4.3. The last step for adapting the mesh to capture the crack path, the *hr*-adaptivity component, is described in Section 5.1 with a detailed description of the algorithm implementation provided in Appendix A.

5.1 | *hr*-adaptive crack propagation

The *hr*-adaptive crack propagation method presented in this section is designed so that a generic mesh $\mathcal{T}$, either with or without hanging nodes, can be considered. Two tools are required for this propagation method, the ability to *h*-refine and *r*-adapt the mesh. The method is based on two principles:

1. The first principle is, it is only necessary to have an accurate representation of the crack path at the crack tip to predict the crack path direction and advancement rate accurately. This premise manifests itself in the mesh as having a high number of elements along the crack path close to the crack tip; this accurately captures the curvature of the crack

path. Whereas, for the remainder of the crack path away from the crack tip, elements with longer edges can be used to approximate the path. The result is, as the crack propagates through the mesh, the mesh is adapted such that small elements along the crack path are only located at the moving tip. This achieves having both an accurate prediction of the crack path and, not having elements with small edges far from the crack tip which contribute a significant computational cost but with only a small increase in accuracy.

2. The second principle is a direct consequence of the first principle and stems from an implementation/data management point of view. It is difficult to refine and, particularly, derefined along a crack path such that: the elements capture the crack path to a desired resolution, are of an acceptable shape and is efficient such that, small elements do not appear in derefined areas. In addition to this, there could be potentially a high number of derefinement and refinement strategies that have to be hard coded to capture particular circumstances that could arise. The second principle is that it is therefore much easier, especially for a crack path that is undergoing continuous derefinement and refinement, to completely recompute the element distribution along the crack path using the original data structure, with relatively small extra computational cost, rather than attempt to adapt an existing mesh. This simultaneously avoids the reloading and assignment of a new data structure if a mesher is used, gives the user greater control over the algorithmic-arguments which govern where new elements are introduced (avoiding unnecessary elements), and avoids the difficult task of implementing a refinement/derefinement algorithm to capture the crack path.

These two principals are used to construct the *hr*-adaptive crack propagation algorithm, the general overview of which is provided in Algorithm 3.

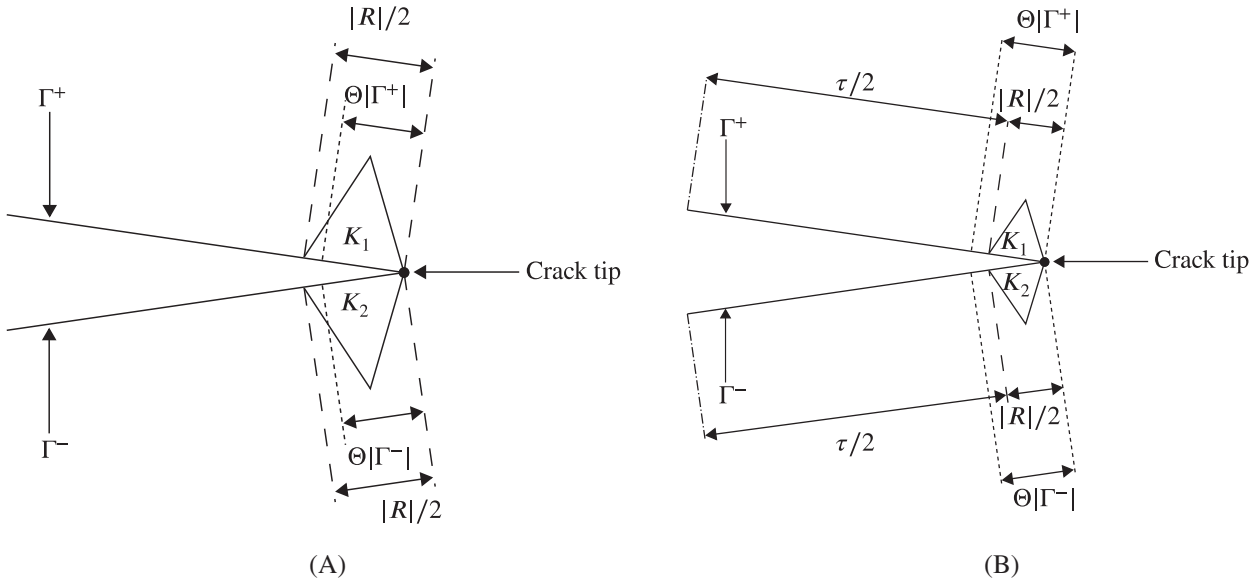


FIGURE 3 Schematics showing the elements at the crack tip for the their total edge length is greater the $\Theta(|\Gamma^+ \cup \Gamma^-|)$ (A), and when their total edge length is less than $\Theta(|\Gamma^+ \cup \Gamma^-|)$ (B)

Algorithm 3. Crack propagation method

1: **for** All crack propagation steps $n \in [1, N]$ **do**
2: Perform *hr*-adaptivity, Algorithm 4, to determine the mesh $\mathcal{T}^n$ using the crack paths determined in step $n-1$ and the stencil mesh $\mathcal{T}^0$
3: Solve the linear elastic problem and determine the position on the new crack tip for all cracks
4: For all cracks, add the new current crack tip location to the end of the crack path list
5: If any crack path intersects the boundary or another crack path, the algorithm terminates
6: **end for**

As shown in Algorithm 3, there are two main data structures that required to determine the mesh $\mathcal{T}^n$:

1. The initial stencil mesh $\mathcal{T}^0$, and the corresponding data structure that corresponds to the time $t = 0$; and
2. the current crack path for all cracks in the mesh at the time $n - 1$.

The stencil mesh, $\mathcal{T}^0$, never changes and is always the mesh that contains the crack paths at time 0, that is, the mesh describing the initial problem. The current crack path is a list that contains all computed crack tip locations which is updated on line 4 of Algorithm 3 and, the crack tip location at time 0. The *hr*-adaptive step, occurring on line 2 of Algorithm 3, determines the mesh for the current crack path and is described in Appendix A.

Generally, the *hr*-adaptive algorithm works by finding the element faces which intersect the crack path and, moving the nearest node of each of these edges onto point where the edge intersects the crack path; the *r*-adaptive step. If a *r*-adaptive move causes either, edges along the crack path to be longer than the value specified by the user, an inverted element, or no nodes are available for *r*-adaptivity, *h*-refinement step occurs to create new nodes in the mesh and the *r*-adaptive step is re-attempted. Once the *hr*-adaptive algorithm has captured all the crack paths, any unreasonably distorted elements, such that their minimum angle is less than a user-defined value, are selected. A simple edge flipping algorithm is then applied to this element list to try and improve the quality, see the works of Reference 71 for a study on smoothing and flipping techniques. However, any poor quality elements that still remain and have high error associated with them will be captured by the *hp*-adaptivity procedure, these elements will be refined and the associated error will decrease. The edge flipping is considered to a computational cheap way to improve the initial error which may reduce refinement and computational cost. The remainder of this section describes the flagging mechanism for *h*-refinement during the *hr*-adaptive algorithm. *h*-adaptivity is used as a mechanism to ensure that the crack path is captured to a sufficient detail and to resolve any issues associated with moving nodes *r*-adaptivity.

5.1.1 | Refinement flags and constraints

The *hr*-adaptive method adapts the mesh so that elements far away from the crack tip approximate the crack path with longer edges, whilst near the crack tip short edges are used. The maximum length of element edges on the crack path far from the crack tip is L_l, and the maximum length of elements close to the crack tip is L_s, highlighted in Figure 4.

The region close to the crack tip that contains the short element edges is defined by L_t, the remainder of the crack path with the longer edges is L_p, as shown in Figure 4. The values of L_l, L_s, and L_t are all user defined whilst the course path length, L_p, corresponds to the remainder of the crack path that is not refined. The rate at which the crack grows can either be user-defined or calculated. All of these lengths are independent of the initial mesh element size.

The *hr*-algorithm has three stages which occur in the following order:

1. *hr*-adaptivity corresponding to the crack path portion L_p and element edge length L_l.
2. *hr*-adaptivity corresponding to the crack path portion L_t and element edge length L_s

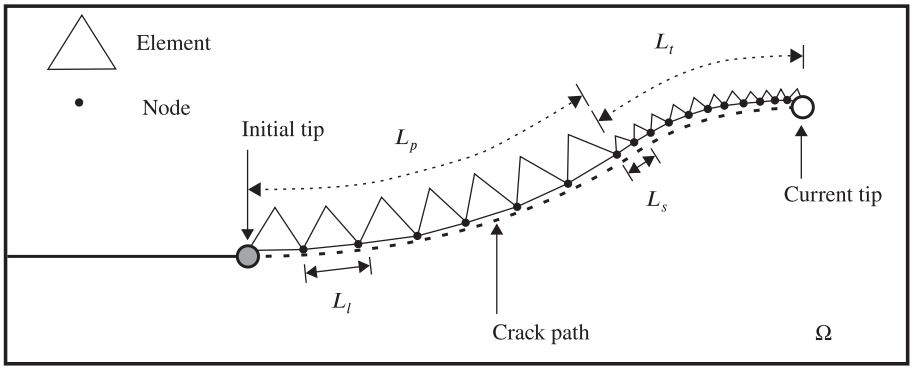


FIGURE 4 An example of a crack propagating from initial tip to the current tip in the domain Ω, where L_l, L_s, and L_t are user-defined lengths controlling the crack path discretization. The length of L_p corresponds to the remainder of the crack path that is not refined

3. Crack tip positioning, prescribed edges for this stage have edges with
$L < \min\{Ls, (2/3) \times \text{length of last propagation step}\}$.

For Stages 1 and 2 the *r*-adaptive process only has to move a node onto the crack path, that is the nodes do not have to coincide with the points that define the crack path. For Stage 3 the *r*-adaptive step moves a node onto the last point of the crack path, this node is the crack tip node. As the lengths are user defined and independent of the element size, moving nodes in the mesh onto the crack path invariably leads to either, element lengths that are too long for the prescribed stage, or inverted elements. Both of these issues are resolved with *h*-adaptivity. It should be noted that nodes that exist on the boundary or are associated with any element that describes a crack path, cannot be chosen for *r*-adaptivity. The exception is for Stage 3, where the final node moved onto the crack path can already be associated with the current crack path.

All stages that require nodes to be moved onto the crack path use an *r*-adaptive procedure and the process is summarized using Figure 5A,B. The most recent *r*-adaptive node is defined as the *"current crack tip node"* and is highlighted in white. This node has been undergone *r*-adaptivity and is associated with the crack path. Next the intersection point of the crack path with one of the element edges is found, marked by the grey circle. The nearest node, marked by the grey box, is then moved to the intersection point. Once the nearest node has been moved, it is associated with the crack path and becomes the new current crack tip node and a new intersection point is found. The process is repeated until the nearest node is moved onto the crack path's crack tip. However sometimes it is not possible to perform an *r*-adaptive step, this is resolved with *h*-adaptivity as described in the following paragraph.

The marking for the *h*-adaptive step can be described with the aid of Figure 5A,B. Figure 5A shows the situation where the node marked to be moved (shown by the grey square) is a hanging node. Moving a hanging node will result in holes or overlaps in the mesh. In this case the element that contains the current crack tip node (the most recent node moved to the crack path in the *hr*-adaptive algorithm) and the edge F that contains the intersection point, are refined. All other cases are described with Figure 5B. In these cases moving the node, marked in grey, to the intersection point either: creates an element edge along that crack path that is too long for the current stage; or, inverts an element. In this case the element which contains the current crack tip node and the edge F with the intersection point is refined.

Stage 3 of the *hr*-algorithm is to ensure that the element edge that has a node at the crack tip is aligned with the final segment of the crack path. This is achieved by enforcing any element that has a node on the last propagation length to have edge lengths of size $L < (2/3) \times$ (length of last propagation step). This is shown in Figure 6, on the left of the figure is a poorly resolved crack tip where the length of L is greater than the last propagation step size, the result is that the element edge of the last element is not aligned with the final crack path direction. On the right of Figure 6 is the same crack path but resolved accurately, here L is less than $(2/3) \times$ (length of last propagation step) and hence the element edge is aligned with the last segment of the propagation path. This has the effect that the direction of the last element edge captures the direction of the crack path at the crack tip.

This section has detailed the numerical ingredients that are required for the proposed fracture modeling framework. The following section will demonstrate the capabilities of the framework. Detail on the implementation of the three stages of *hr*-adaptive method are described in detail in Appendix A, the details have been excluded here for the sake of brevity.

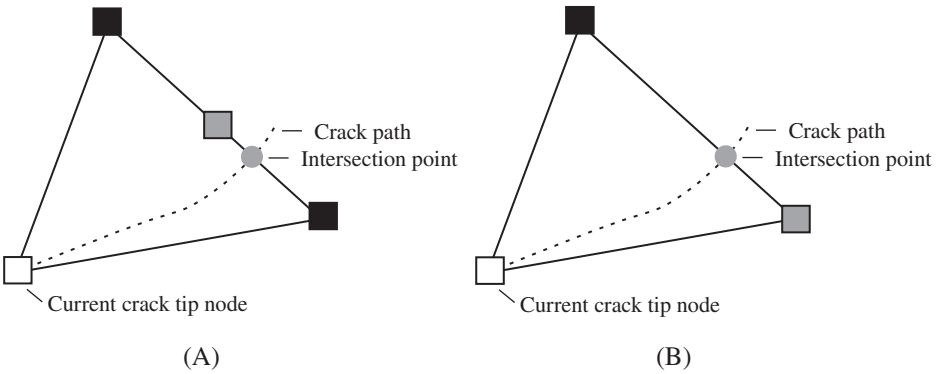


FIGURE 5 Flagging elements for *h*-refinement when the node to undergo *r*-adaptivity, marked in grey, is either (A) when a hanging node or (B) creates an edge along the crack path that is too long or creates an inverted element

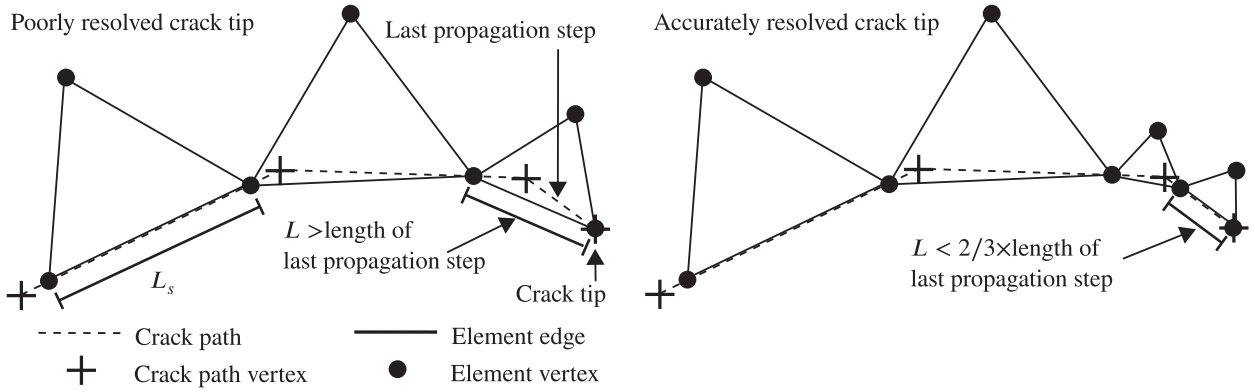


FIGURE 6 A diagram showing element edges poorly (left) and accurately (right) resolving the crack path at the crack tip

6 | NUMERICAL VALIDATIONS AND EXPERIMENTS

The purpose of the four numerical examples are to test/demonstrate specific aspects of the developed formulation, namely:

1. *Westergaard embedded static crack*: validation of the error estimate for the CF and the appropriateness of the stopping criteria for a problem with a closed form analytical solution;
2. *Single shear crack*: to demonstrate that the *hpr*-adaptive method is highly accurate in terms of computing the crack propagation path; and
3. *Y-shaped crack*: to demonstrate the importance of derefinement in terms of the overall numbers of DOF of the finite element problem, investigate and show that it is only necessary to have small elements near the crack tips in terms of the accuracy of the predicted crack propagation path and to show that the method is further show that the method is highly accurate; and
4. *Tree crack*: to demonstrate the ability of the *hr*-adaptive method to deal with problems with a large number of cracks propagating at different rates.

For each problem in the following sections a number of parameters are used and are provided. However to improve the readability all the parameters are also provided here in Table 1.

6.1 | Westergaard embedded static crack

In Reference 39 the error estimate for the CF area integral, (21), was shown to reliable and efficient in that it bounded the true error from above and below. Here the error estimate χ_τ (36) for the line integral component of the CF has already been shown to be reliable, see Section 4.2, and in this section will be shown to also efficient using the Westergaard stress solution.[72] The Westergaard problem considers a double ended crack, of length $2a$, in an infinite plate when the origin of the coordinates $(x, y) = (0, 0)$ is at the crack center. The closed form stress solution for the mixed mode case is found by summing the stress solution for the pure mode I and II case provided by:[72]

$$\begin{aligned}\sigma_{xx} &= \mathrm{Re}(Z_{\mathrm{mixed}}) - y \cdot \mathrm{Im}(Z_{\mathrm{mixed}})\\ \sigma_{yy} &= \mathrm{Re}(Z_{\mathrm{mixed}}) + y \cdot \mathrm{Im}(Z'_{\mathrm{mixed}}), \quad \text{where} \quad Z_{\mathrm{mixed}} = \frac{(\sigma_0 + \tau_0)z}{\sqrt{z^2 - a^2}} \quad \text{and} \quad z = x + iy.\\ \sigma_{xy} &= -y \cdot \mathrm{Re}(Z'_{\mathrm{mixed}})\end{aligned} \tag{39}$$

In Equation (39), a is half crack length, z is the complex coordinates, Z_{mixed} is a complex number (with complex conjugate Z'_{mixed}) and the stress terms σ_{xx}, σ_{yy}, and σ_{xy} are real numbers. The normal and shear stress at the infinite boundary are respectively σ_0 and τ, for this example both are set to 1 Pa.

Since the geometry of the problem is symmetric, and the stress solution is anti-symmetric, it is only necessary to model half the problem, the geometry of which is provided in Figure 7A with $W = 1$ m, $H = 0.5$ m, and $a = 0.5$ m, where the crack tip labeled with $\partial\Gamma$. The plate acts in plane stress with a Young's modulus of 1 Pa and Poisson's ratio of 0.3. The mixed mode Westergaard stress solution is applied to the boundary of the domain as a Neumann boundary condition g_i^N however, since no Dirichlet boundaries exist an average boundary condition is also applied to prevent rigid body motion.[§] The initial mesh of the problem is shown in Figure 7B, comprising of 172 elements with a uniform polynomial order of $p_K = 2$.

TABLE 1 Parameters used for the numerical validations and experiments in Section 6

		Westergaard	Single shear	Y-shape	Tree
hp static parameters	δ_2	0.7	0.7	0.7	0.7
	δ_1	0.07	0.07	0.07	0.07
hr propagation parameters	L_t	-	0.1 m	1.25 m	0.25 m
	L_l	-	0.5 m	1 m	0.5m
	L_s or $\Delta a_{\max}$	-	0.01 m	0.1 m	0.05 m
	Θ	0.1, 0.01, 0.001, 0.0001	0.005, 0.8	0.01	0.01
	Steps	-	45	150	60
Material constants	E	1 Pa	-	1 Pa	1 Pa
	ν	0.3	0.3	0.3	0.3
	G	-	8 GPa	-	-
Crack material constants	g_c	-	10^3 N/m	-	-
	C	-	-	-	1
	m	-	-	-	2

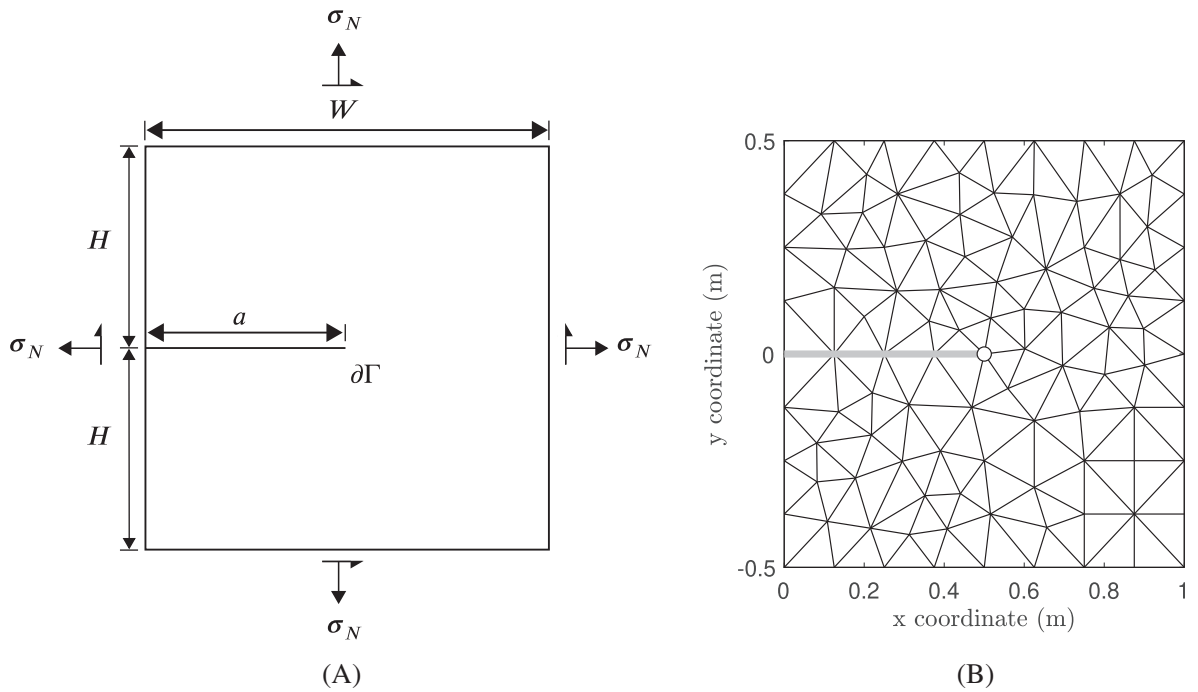


FIGURE 7 Westergaard embedded crack: (A) Problem geometry and (B) initial mesh with the crack faces highlighted in grey and crack tip with the white circle

To show that χ_τ is efficient for the true error of the line integral component of the CF (21), four different values for Θ are considered, 0.1, 0.01, 0.001, and 0.0001. The value of Θ corresponds to the proportion of the crack face of Γ^+ and Γ^- that is excluded from the line integral of the CF; $|R| = \Theta|\Gamma^+| = \Theta|\Gamma^-|$; see (21) and Figure 2 for the equation and diagrammatic description respectively. The problem is then modeled with the initial mesh shown in Figure 7B, which is *hp*-adapted over 25 steps using the refinement parameters $\delta_2 = 0.7$ and $\delta_1 = 0.07$. Since the crack is considered static there is no need to define the propagation parameters L_t, L_l, and L_s and, the number of propagation steps. Finally, the definition of the domain for the CF integral, which is unchanged throughout this numerical experiment, is defined as the elements that exist at the crack tip shown in Figure 7B.

The ratio between the estimated error and the true error[¶] with *hp*-adaptivity is shown in Figure 8A. The convergence of the estimated error $\chi_\mathcal{T}$ with the square root of the number degrees of freedom (NDOF) is shown in Figure 8B, with a linear abscissa and a logarithmic ordinate.

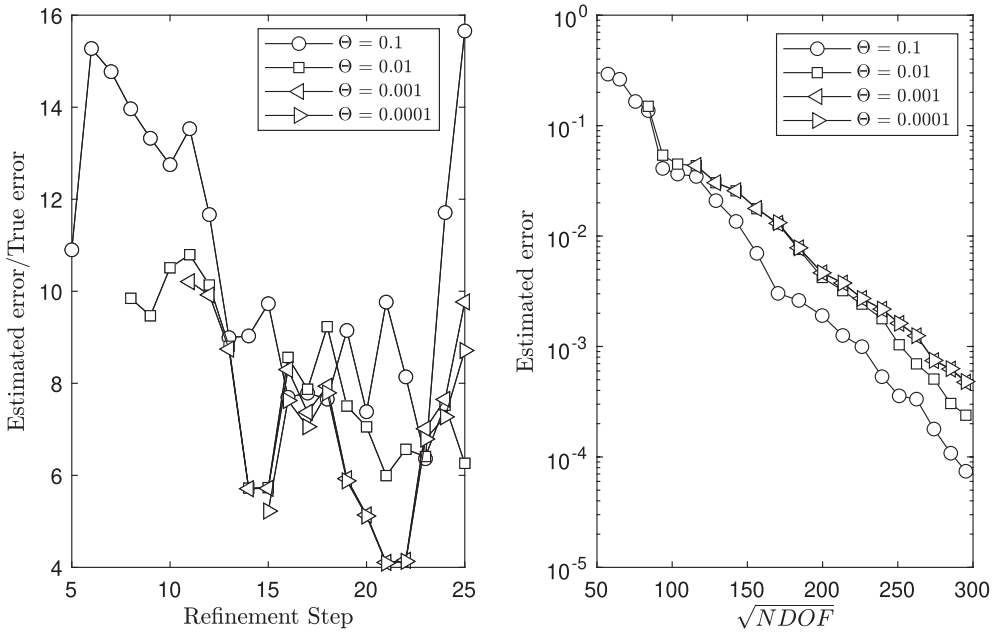
Two conclusions can be made from the Westergaard problem:

1. Independent of the choice of Θ, the error estimator for the line integral terms, χ_τ, is shown numerical to bound the true error for the CF line integral from above and below, up to a positive constant.
2. The estimated error, and therefore also the true error provided the above conclusion, convergences exponentially with *hp*-adaptivity for any choice of Θ.

The first conclusion is shown through the results presented in Figure 8A. In Section 4.2 it was shown the error estimator χ_τ bounds the line integral of the CF from above and Figure 8A demonstrates numerically through the random oscillations of the estimated error over the true error that $\chi_\mathcal{T}$ bounds the true error from below. This result demonstrates that it is reasonable to extend (36) to include an additional inequality

$$c_\tau \chi_\tau < \|\psi - \psi^h\|_\tau < C_\tau \chi_\tau, \tag{40}$$

where c is an positive constant. Hence the error estimate $\chi_\mathcal{T}$ is never too far away from the true error and the convergence of the true error is at worst as good as $\chi_\mathcal{T}$, or better. Further through the consideration of multiple values of Θ, Figure 8A demonstrates that, irregardless of the length of the crack edges over which the error is being estimated, (40) holds.



(A) effectivity index (B) convergence

FIGURE 8 Westergaard embedded crack: (A) Variation of the effectivity index (ratio of the estimated/true error) with refinement steps and (B) estimated error evolution with increasing NDOF for $\Theta = 0.1$, 0.01, 0.001, and 0.0001

The second conclusion is shown with Figure 8B. For all values of Θ exponential convergence of χ_τ is observed with *hp*-adaptivity. This is shown in the plot as the convergence is rough straight line on a log-linear scale, see References 62,73. Additionally the figure shows how changing the value of Θ, and hence also the magnitude of $|R|$, changes the value of the error estimate. When Θ is decreased so is the region R near the crack tips which is not included in the line integral. The result is that the domain of the CF line integral increases to include more of the region close to the crack tip singularity, which is where the underlying finite element basis functions are unable[#] to represent the true stress field variation. This leads to an increase in the computed CF error. Further, it is important to note that for lower values of Θ, more *hp*-adaptive steps are required before the computation of the line integral error can occur. This is a direct results of Algorithm 1, if the element at the crack tip is not small enough to be contained completely within R, where R is defined by Θ, then the line integral component of the CF cannot occur and thus there is no corresponding error.

The final observation from Figure 8B is that for the initial *hp*-adaptive steps, the estimated errors have initially similar values which subsequently diverge with increasing adaptive steps. This feature occurs more strongly with low values of Θ. In these cases the CF line integral includes more of the crack edges near the crack tip. As Θ is decreased the estimated error considers a length of the crack faces that becomes asymptotically close to the full crack edge length, hence convergence towards a value is observed. Last, the estimate error diverges with increasing *hp*-adaptive step, this is because a lower Θ value includes more of the crack tip solution, which is non-smooth and hence, a slower convergence is observed compared to a larger Θ that could potentially only include a smooth region of the solution and thus converges faster.

6.2 | Single shear crack

The shear crack problem, shown in Figure 9A, is used to demonstrate the accuracy of the CF, evaluated using Algorithm 3. The shear crack domain, Figure 9A, has dimensions $H = W = 1$ m, crack length $a = 0.5$ m, shear modulus $\mu = 8.0$ GPa and Poisson's ratio of 0.3, the crack has a critical energy release rate of 10^3 N/m, and the problem acts in plane stress. The bottom edge of the domain is fixed and an inhomogeneous Dirichlet boundary condition is applied to the top with $g_i^D = [u\ 0]$, where u is the displacement required to cause fracture growth. The initial mesh for this problem is shown in Figure 9B, comprising of 172 elements with a homogeneous polynomial order $p_k = 2$. The mesh is *hp*-adapted using the refinement parameters $\delta_2 = 0.7$ and $\delta_1 = 0.07$. For both problems 45 propagation steps were performed, each propagation step was set to $L_s = 0.01$ m, the element edge length away from the crack tip was set to $L_l = 0.5$ m and the length of the crack path near the crack tip with refined elements was set as $L_t = 0.1$ m.

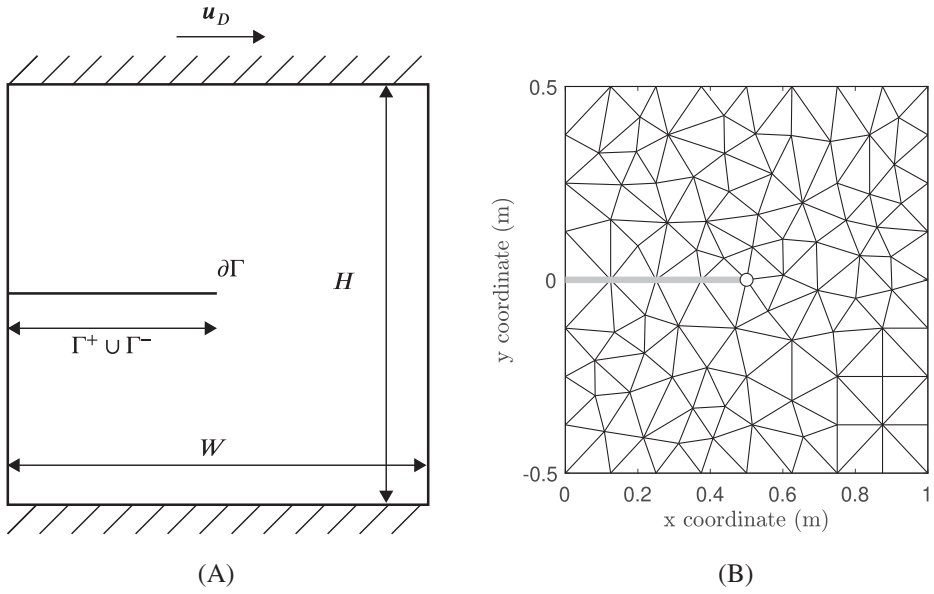


FIGURE 9 Single shear crack: (A) Problem geometry and (B) initial mesh with the crack faces highlighted in grey and crack tip with the white circle

The shear crack problem is a particularly interesting problem due to its high sensitivity to any heterogeneity, numerical or material. The result is that there are three distinct crack paths that could arise, either downwards, straight across or upwards depending on the nature of the heterogeneity. This benchmark crack path problem is based on the experimental work by Erdogan et al.[25] where the crack is observed to propagate downwards. Many authors in the literature have examined this problem using a range of: numerical methods (mesh-based and meshless), crack propagation analysis techniques (discrete or smeared)[74] and criteria for the crack propagation direction, see the non-exhaustive list of References 13,51,74-77. They all show that the crack propagation path is downwards and that the angle of the initial crack propagation path is the same, or similar, to that observed in the experiment. However, the authors here postulate that the crack path solution predicted in the simulations is simply the result of spurious numerical heterogeneity. In the experiment the heterogeneity is due to the boundary conditions not exactly being a pure displacement shear, the domain geometry not being exactly symmetric and the material properties not being homogeneous and precisely isotropic.

Inspection of the continuous form of the CF calculation (2) shows that if the plane Eshelby stress component Σ_{yy} is symmetric and the shear stress component Σ_{yx} is antisymmetric about the crack tip, only the CF component that acts parallel to the crack faces will be non-zero (this has been confirmed numerically). The consequence being that the direction of propagation that maximizes the dissipation of energy is a pure mode II fracture path, hence any smeared crack modeling that predicts crack growth on a maximization of energy dissipation should also predict a horizontal crack propagation path. Further, for this homogeneous isotropic problem, only the mode II SIF, K_{II}, will be non-zero and as such the MSERRC, MCSC, and MSEDC methods, will all also predict a horizontal crack propagation path. However, if due to numerical inaccuracies the crack deviates slightly from the centerline, the problem becomes non-symmetric and mixed mode propagation occurs with the new mixed mode direction of propagation corresponding to the maximum energy dissipation. Normally such distinct crack propagation paths would not exist however, since the crack tip lies on a trifurcation point any heterogeneity in the simulation has the result of distinctly different paths and failures. Through the consideration of a non-symmetric mesh, the analysis of the shear crack with the *hpr*-propagation method[‖] will demonstrate that this method is robust to any initial numerical heterogeneity. This will show that the algorithm provides an ideal platform for the analysis of spatial and material heterogeneity, and anisotropy, for highly sensitive problems safe in the knowledge that numerical artefacts have been removed.

To investigate how mesh heterogeneity affects the crack propagation path for this problem, two values of Θ (the error threshold for the CF calculation) for the determination of the CF, Algorithm 1, are considered, 0.005 and 0.8. In total 45 propagation steps occurred with each propagation step having a length of $L_s = 0.01$ m, the element edge length away from the crack tip was set to $L_l = 0.1$ m and the length of the crack path near the crack tip with refined elements was set as $L_t = 0.1$ m. The final meshes, and crack propagation path, are shown respectively in Figures 10A and 11A. Figure 10A show two of the three crack propagation paths that can occur—straight across and downwards.

When Θ is set to a low value Algorithm 1 computes the CF to a high accuracy. This is achieved with *hp*-adaptivity which minimizing any effects the mesh has on the stress solution, thus removing mesh induced stress non-symmetries induced by the initial non-symmetric mesh. When $\Theta = 0.005$ sufficient *hp*-adaptivity occurs for a mode II crack path to be predicted, see Figure 10A. However for $\Theta = 0.8$, significant mesh heterogeneity still exists resulting in a mixed mode

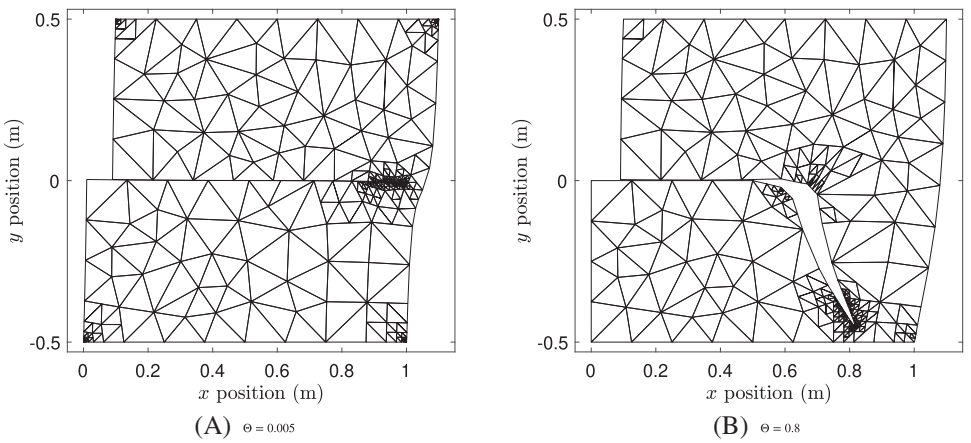


FIGURE 10 Single shear crack: Deformed mesh at end of the analysis for (A) $\Theta = 0.005$ and (B) $\Theta = 0.8$

downward propagation path, Figure 10B. Initially, for $\Theta = 0.8$, the mesh heterogeneity is not significant enough to cause an initial sudden downward propagation path but does cause a slight deviation of the crack tip from the centerline, making the geometry non-symmetric. The combined effect of the geometric and the mesh non-symmetric causes a positive feedback loop, the stress solution about the crack tip becomes increasing non-symmetric which in turn causes the next propagation step to be further from the centerline, Figure 11A. The result is the crack tip deviates from the crack centerline at a rate which increases with propagation step. This highlights the difficulty of analyzing this problem, quickly after the crack tip deviates from the centerline the propagation path becomes strongly mixed mode with no possible recovery to the centerline. This feature demonstrates the accuracy achieved with the *hpr*-adaptive method when a low value of Θ is set.

The total load applied to the top boundary, and corresponding energy, to cause brittle crack propagation for each propagation step is recorded respectively in Figure 11B,C. Since the energy dissipated by a brittle crack per unit length is constant and the propagation step size is constant, every propagation step dissipates the same quantity of energy from the system. Hence, in Figure 11C the lower the applied energy on the boundary the higher the proportion of the strain energy that is dissipated by a crack propagating. Inspection of Figure 11B,C shows that for $\Theta = 0.005$ there is a continuous decrease in the load, and the energy, required to propagate the crack. Whereas, for $\Theta = 0.8$ there appears to three stages of propagation: (i) from $0 \rightarrow 0.03$ m a similar load and energy is required as for $\Theta = 0.005$; (ii) A sudden decrease in the load and energy from $0.04 \rightarrow 0.08$ m; and (iii) a continuous decrease in the load and energy for the remained of the propagation. Inspecting load and crack path deviation values for stages (i) to (ii) demonstrates the for the two simulations how quickly the crack paths become different. Until 0.03 m the two propagation paths are similar, with a respective deviation for $\Theta = 0.005$ and $\Theta = 0.8$ from the centerline of -1×10^{-4} and -2×10^{-4} m, Figure 11A, however at 0.04 m the derivation becomes -2×10^{-3} and -4×10^{-4} m. At 0.04 m, the loads are also similar, 4759 N and 4846 N, but rapidly become different; at 0.08 m the loads are 2866 N and 4563 N respectively. The results highlight the sensitivity of this problem, the crack path needs only to deviate -2×10^{-3} m, corresponding to an angle of 2.9° at a total length of 0.04 m, for unrecoverable symmetry is lost and for the crack to propagate downwards. The plot of the applied energy in Figure 11C shows how, if the initial symmetry is lost, how much for favorable the initial deviation of the crack path from the centerline is compared to continued pure mode II propagation. Between $0.04 \rightarrow 0.08$ m, the required into energy to propagate a crack downwards roughly halves, when compared to the pure mode II case ($\Theta = 0.005$), meaning that the proportion of energy dissipated from the system by crack propagation for $\Theta = 0.8$ is roughly double that of $\Theta = 0.005$. After 0.08 m all symmetries of the stress field are lost and the crack continues to propagate downwards. In total the crack path only deviates by 6×10^{-4} m, or an angle of 0.036°, from the centerline for $\Theta = 0.005$.

As discussed above, any slight geometric non-symmetry caused by crack propagation produces a positive feedback loop of increased loss in geometric and stress symmetry resulting in a increasingly downwards crack propagation path.

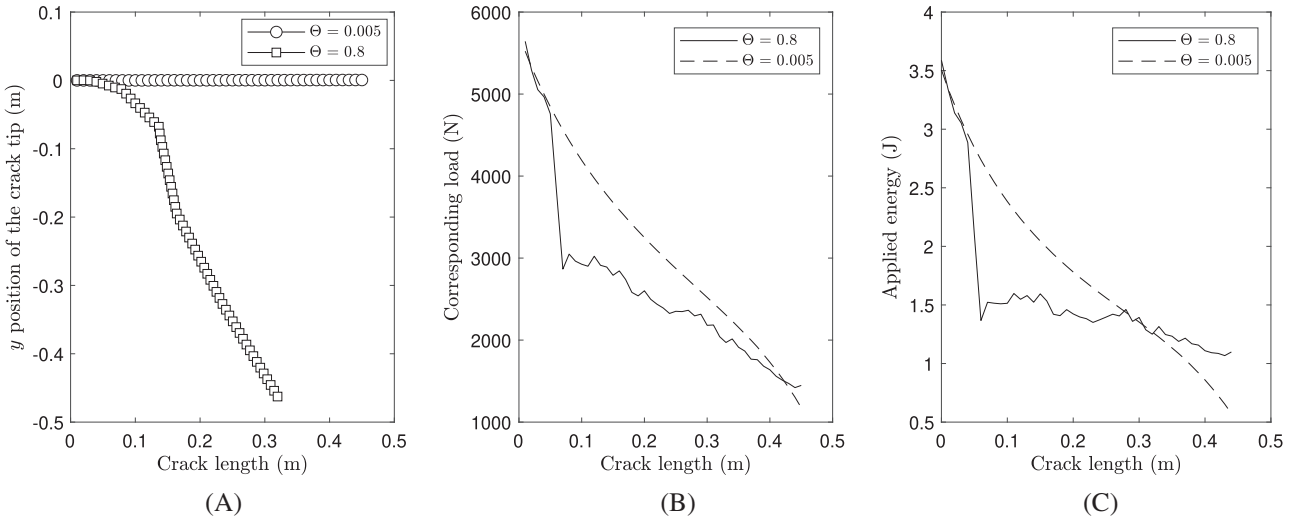


FIGURE 11 Single shear crack: (A) Single shear crack: crack propagation paths for $\Theta = 0.005$ and $\Theta = 0.8$, where the discrete points show the individual propagation step crack tip positions, (B) reaction force versus displacement for the two crack paths, and (C) the energy applied to the boundary

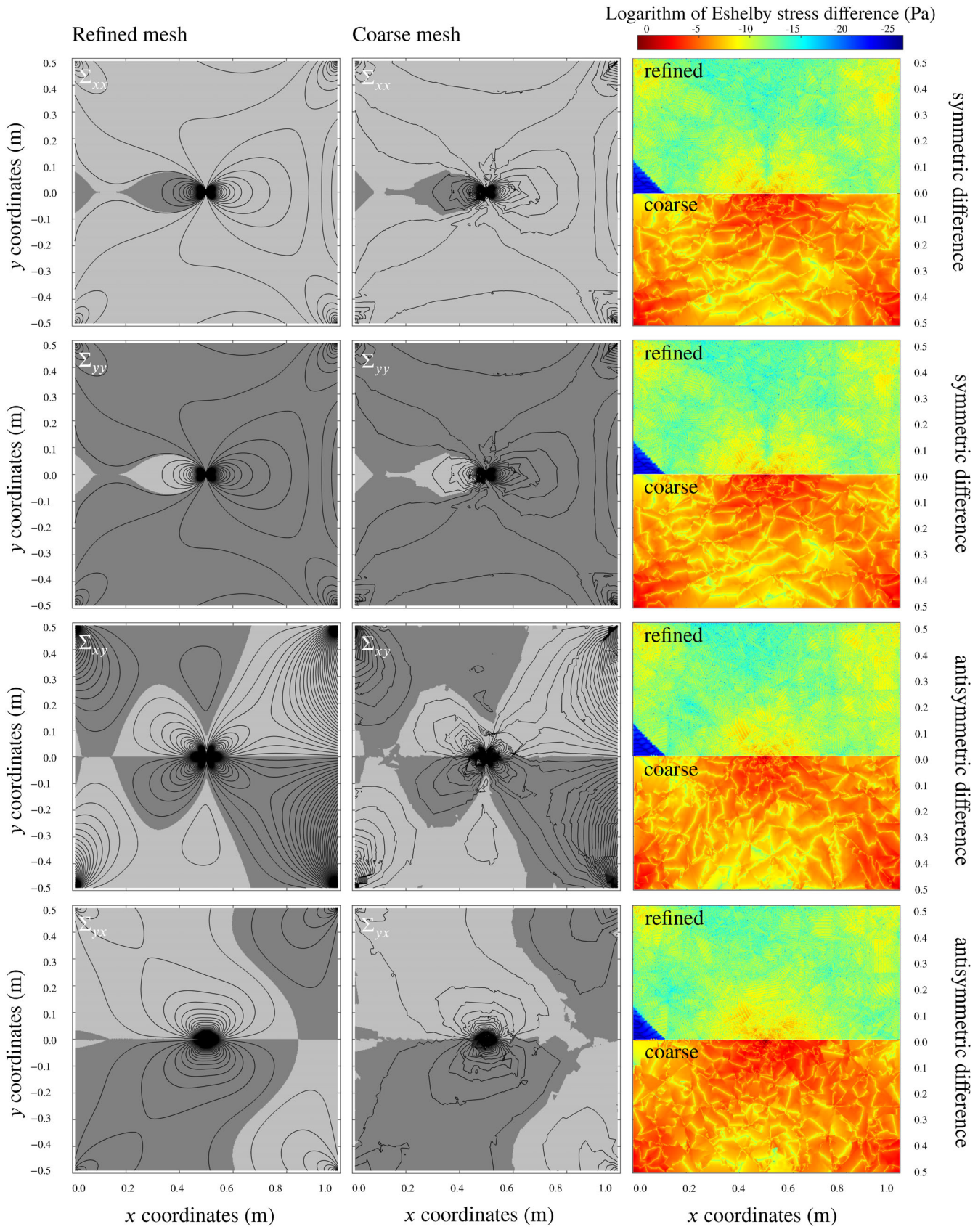


FIGURE 12 Single shear crack: Spatial variation of the Eshelby stress components prior to crack propagation for $\Theta = 0.005$ (left) and $\Theta = 0.8$ (center) along with the variation from anti-symmetry of the stress components (right) for $\Theta = 0.005$ ($y > 0$) and $\Theta = 0.8$ ($y < 0$)

However, for the crack to initially propagate downwards the CF direction, which dictates the direction of crack growth must also point away from the centerline of the problem. Therefore in order for this to happen the CF must be a function of a non-symmetric and non-anti-symmetric Eshelby stress field, even in the case when the geometry is symmetric. By considering the initial problem domain, Figure 9A, Figure 12 shows how the symmetry and anti-symmetry of the Eshelby stress terms varies with a refinement. The figure is split into four rows corresponding to Σ_{xx}, Σ_{yy}, Σ_{xy}, and Σ_{yx}. The first two columns show the Eshelby stress distribution for when the $\Theta = 0.005$ and $\Theta = 0.8$, the isolines are the logarithm of the absolute values of each Eshelby stress term and, the light and dark grey colors correspond to respectively to negative and positive areas of Eshelby stress. The last column provides a color plot of either the symmetric or anti-symmetric difference in the Eshelby stress terms for both the refined and course mesh. For an Eshelby term, the top half of the color plot corresponds to the refined mesh, the bottom the coarse mesh, with the absolute position of the y-coordinate starting at the centerline of the plot and increasing both down and upwards. Inspecting the first two columns of the figure clearly shows that when the geometry is symmetric and a high number of *hp*-refinement steps occur, the plane Eshelby stress terms are symmetric whilst the shear terms are anti-symmetric. However, in the case when only a few *hp*-refinement steps, non-symmetries are present in the Eshelby stress. The third column shows the value of the non-symmetry is significantly higher for the coarse problem, at approximately six orders of magnitude larger than that of the refined problem for the plane Eshelby terms and approximately three orders of magnitude for the shear Eshelby terms, see Table 2.[**] Table 2 highlights specifically that the difficulties in modeling this problem are more strongly associated with the shear Eshelby stress terms, both in terms of their absolute values but also in terms of their convergence with *hp*-adaptivity. The table there shows that the second component of the CF, the mixed mode component that acts perpendicular to the crack faces, has its main contribution from the Σ_{yx} as for both refinements, this anti-symmetric difference is larger when compared to the symmetric difference of the plane stress Eshelby components.

The conclusions to this analysis are:

1. This is a pure mode II crack propagation problem however, the result lies on a highly sensitive bifurcation point where the problem can become mixed mode due to numerical inaccuracy.
2. The analysis shows that as soon as a small deviation of the crack path exists from the problem's centerline, (2 mm of a 1 m square domain) the domain is sufficiently non-symmetric that a positive feedback loop is generated; the problem becomes more non-symmetric and the crack propagates downwards at an increasing rate.
3. The example highlights the accuracy of the *hpr*-propagation method. Only a method that can accurately compute the crack path direction, such that it is insensitive to any mesh heterogeneity, and is capable of capturing the true crack path.
4. Any deviation from a horizontal crack path for this quasi-static, symmetric, and isotropic problem is a numerical artefact associated with poor stress and displacement field approximation, as confirmed by analyzing the Eshelby stresses distribution.

6.3 | Y-shaped crack

This problem contains three crack tips, two of which are mixed mode and propagate symmetrically whilst the third is mode I. There are four reasons for exploring this numerical example:

i to demonstrate that CF driven cracks can model kinked fractures;
ii that a very high precision for the crack path can be achieved through inspection of the symmetric crack growth;

TABLE 2 Single shear crack: Mode Eshelby stress differences about $y = 0$

	$\log_{10}$(symmetric difference)		$\log_{10}$(anti-symmetric difference)	
	Σ_{xx}	Σ_{yy}	Σ_{xy}	Σ_{yx}
Refined mode	−5.4	−5.4	−1.6	−2.3
Coarse mode	0.34	0.34	0.9	1.15

iii capturing the crack path accurately only at the crack tip requires significantly fewer DOF compared to the case of capturing accurately the entirety of the crack path; and
iv only accurately capturing the crack path at the crack tip is necessary for accurate crack propagation, with the results indistinguishable when compared to the case with no derefinement along the crack path.

The domain for the Y-shaped crack is shown in Figure 13A, it acts in plane strain and has a Young's modulus of 1 Pa and a Poisson's ratio of 0.3. The values of the domain parameters of the Y-crack are $W = 20$ m, $H = 16$ m, $a = 1$ m, $b = 1/\sqrt{2}$ and $\theta = 45°$. On its top and bottom boundary a uniform normal outwards pressure of 1 Pa is applied so that the plate experiences a uniaxial tensile load. As only Neumann boundary conditions are applied to the boundary of the problem, average boundary conditions are applied, as in Reference 39, to prevent rigid body motion. The initial mesh is shown in Figure 13B, it contains 409 elements and has an initial uniform polynomial order of $p_K = 2$ and is subject to the *hp*-adaptive parameters $\delta_2 = 0.7$ and $\delta_1 = 0.07$.

In this example the *hr*-adaptive propagation algorithm, Algorithm 4, propagates the crack with a total of 150 propagation steps. The case when the crack path is only accurately captured near the crack tip is referred to the derefinement case. When derefinement is present the *hr*-parameters for crack growth are: $L_t = 1.25$ m, $L_l = 1$ m, $L_s = 0.1$ m, and $\Theta = 0.01$. When there is no derefinement: $L_t =$ crack length, $L_s = 0.1$ m and $\Theta = 0.01$.

Since the problem is symmetric about $y = 0$ m, the absolute relative position of the crack paths for crack tips B and C (see Figure 13A) should be identical. The crack paths are shown in Figure 14A, where it is clear that crack A propagates in a horizontal direction and cracks B and C follow the same absolute relative crack path. The crack paths for B and C also demonstrate kinked propagation, this is observed in the initial propagation steps where the cracks propagate immediately horizontally across the domain compared to the initial inclined angle of the crack faces.

A more precise investigating into the error in the symmetric precision is achieved through the measures,

$$\text{crack A propagation error} = \frac{|\boldsymbol{x}_{\mathrm{A}}|}{L_{\mathrm{A}}} \quad \text{and} \quad \text{cracks B \&C propagation error} = \frac{|\boldsymbol{x}_{\mathrm{B}} - \boldsymbol{x}_{\mathrm{C}}|}{\frac{1}{2}(L_{\mathrm{B}} + L_{\mathrm{C}})},$$

where $\boldsymbol{x}_{\mathrm{A}}$, $\boldsymbol{x}_{\mathrm{B}}$, and $\boldsymbol{x}_{\mathrm{C}}$ are the relative positions of the crack tips A, B, and C relative to their initial position marked with the black squares in Figure 13A. L_{A}, L_{B}, and L_{C} are the total created crack propagation lengths at each propagation step. These error measures were recorded for the 150 propagation steps and plotted in Figure 14B.

The results in Figure 14B show that a high symmetric precision is obtained for all cracks. For crack A the maximum symmetric precision error was $\approx 3 \times 10^{-5}$, and after 100 propagation steps plateaued to $\approx 2 \times 10^{-5}$. The error in precision

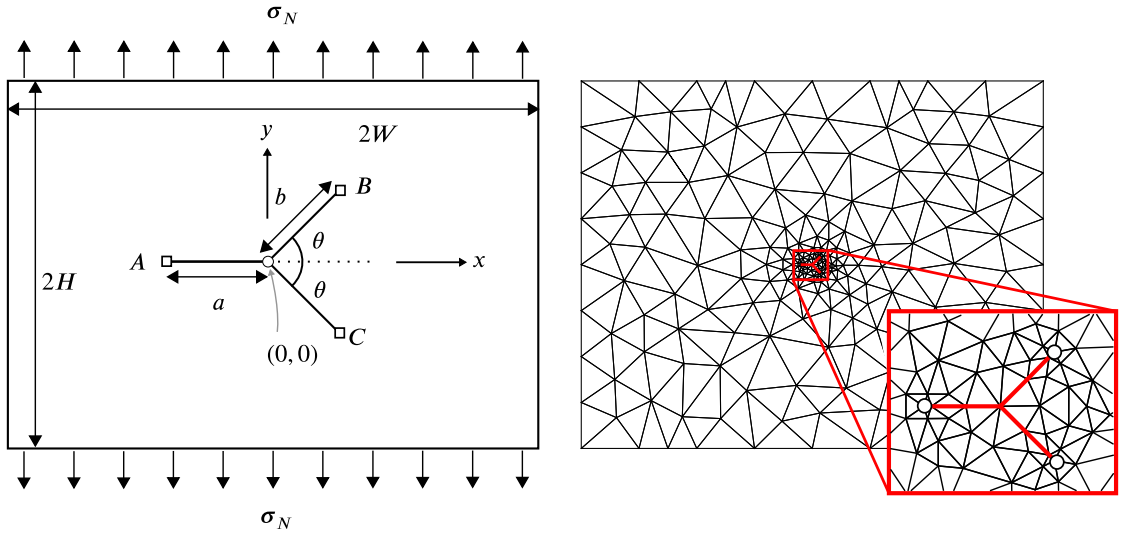


FIGURE 13 Y-crack: (A) Problem geometry, where the Y-crack has been enlarged for clarity, and (B) initial mesh, showing the true initial crack geometry, and exploded view of the mesh around the Y-crack

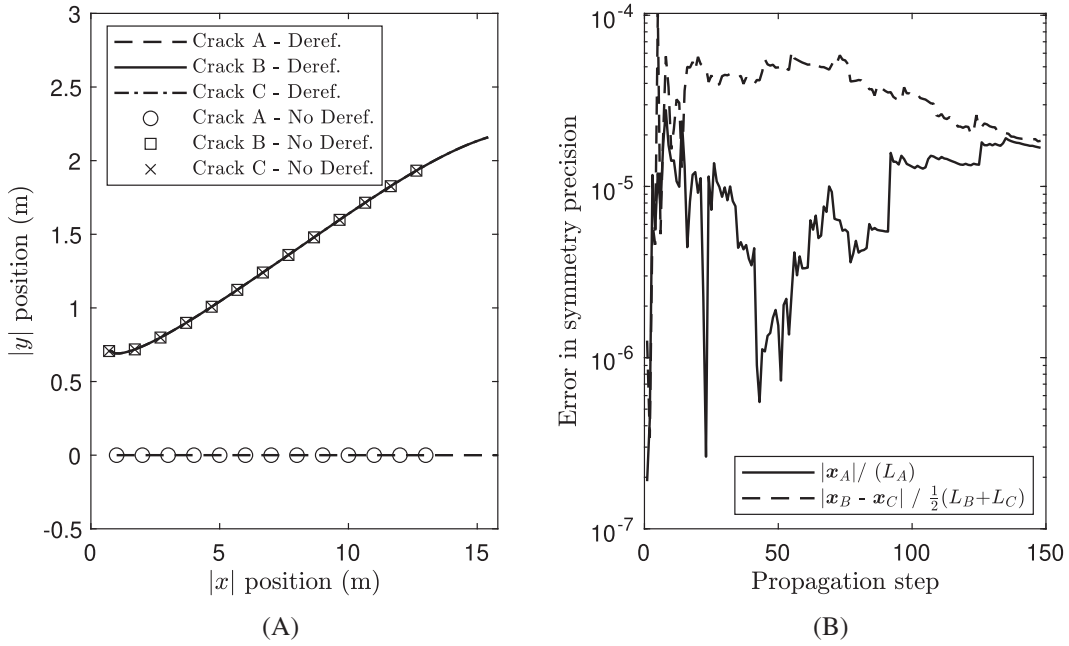


FIGURE 14 Y-crack: (A) Absolute crack paths and (B) normalized non-symmetry of the crack paths over 150 propagation steps

for cracks B and C is consistently slightly higher, with a maximum of 10^{-4} which decreases to $\approx 2 \times 10^{-5}$ at the 150th propagation step. The error values for cracks B and C demonstrate the accuracy in *hpr*-adaptive method. When the mesh is *hr*-adapted, elements are refined and nodes are moved to align with the crack path. The result is for a propagation step, the CF calculation (21) for cracks B and C will have different area integral domain shapes and sizes as well as different crack edge lengths of Γ^+ and Γ^-. Further, the length of R, the ignored region at the crack tip, is different at each propagation step for cracks B and C. The high precision obtained shows the accuracy of Algorithm 1 and demonstrates that provided Θ is sufficiently small, the calculation is robust to different values of $|R|$. This is particularly notable since the high precision in the crack path is consistently maintained with propagation, despite the causality, and resultant cumulative error, associated with subsequent propagation steps.

The crack propagation paths for the derefinement and no-derefinement case, shown in Figure 14A, are indistinguishable. This demonstrates that it is only necessary to maintain a high accuracy description of the crack path at the crack tip in order to obtain precise crack propagation paths. As the crack propagates through the mesh, the mesh evolves to capture the crack path. Whether derefinement is present or not, the number of nodes describing the fracture path will increase as shown in Figure 15A. However, to capture the crack path using the *hr*-adaptive method, the rate increase in the number of nodes required the crack path is significantly smaller when derefinement is present compared to no-refinement. Figure 15B shows the cumulative NDOF required to calculate the CF using *hp*-adaptivity when $\Theta = 0.01$ for both the derefinement, the no-derefinement case and, as a control the cumulative DOF required to compute the first *hp*-adaptive step 150 times. The first observation is that the computation of the no-derefinement case always requires more DOF than the derefinement case, this is observed through the steeper gradient of the former. The second observation is that on average the derefinement case requires the same NDOF as the control (initial mesh multiplied by the propagation step), this is shown by these two lines having the same gradient. This demonstrates that the added DOF required to capture the crack path accurately about the crack tip using *hr*-adaptivity, Algorithm 4, has no impact on the total NDOF to compute the CF at each propagation step using *hp*-adaptivity, Algorithm 1. However, when no-derefinement is present the DOF required to capture the crack path, shown in Figure 15A, increases the NDOF to required to accurate compute the CF at the crack tip Figure 15B. It is concluded that the derefinement strategy has only an indistinguishable effect on the predicted crack path whilst also maintaining computational efficiency adding no noticeable cost to the accurate computation of the CF.

The last aspect of the Y-crack problem to consider is the mesh quality when derefinement occurs. Here the minimum angle observed for all elements in the mesh for the 150 propagation steps is shown in Figure 16. The angle plotted corresponds to the initial mesh generated by *hr*-adaptivity and during *hp*-adaptivity remains unchanged due to the homogeneous *h*-refinement.

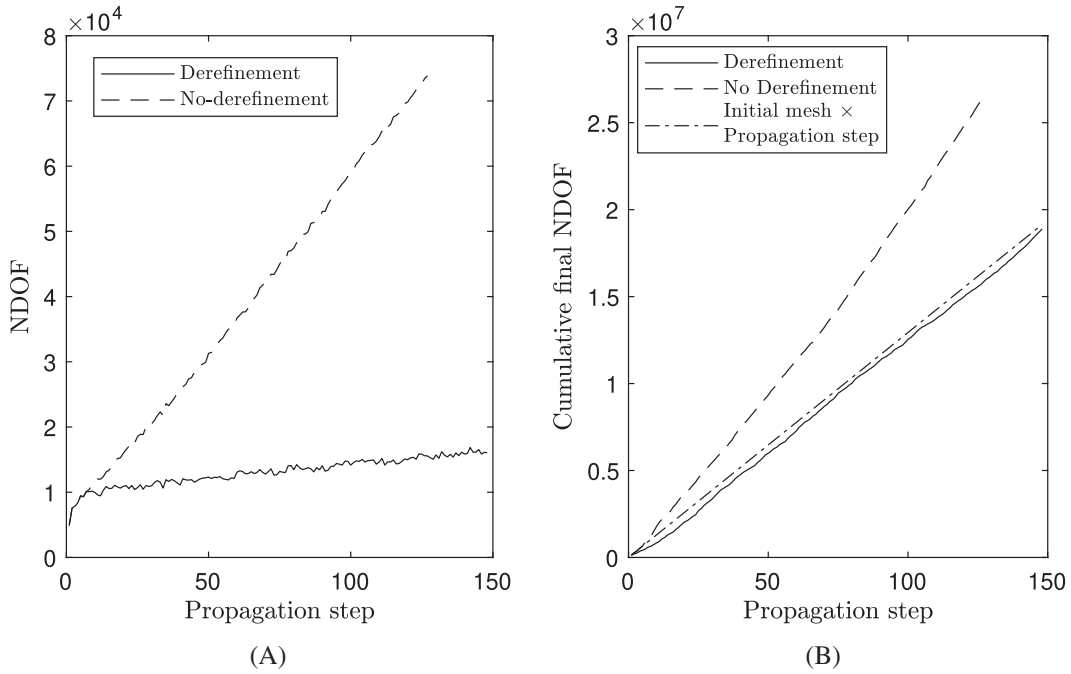


FIGURE 15 Y-crack: (A) Number of degrees of freedom (NDOF) required to describe the crack path for the derefinement and no-derefinement propagation schemes and (B) the final NDOF for each propagation step for the derefinement scheme

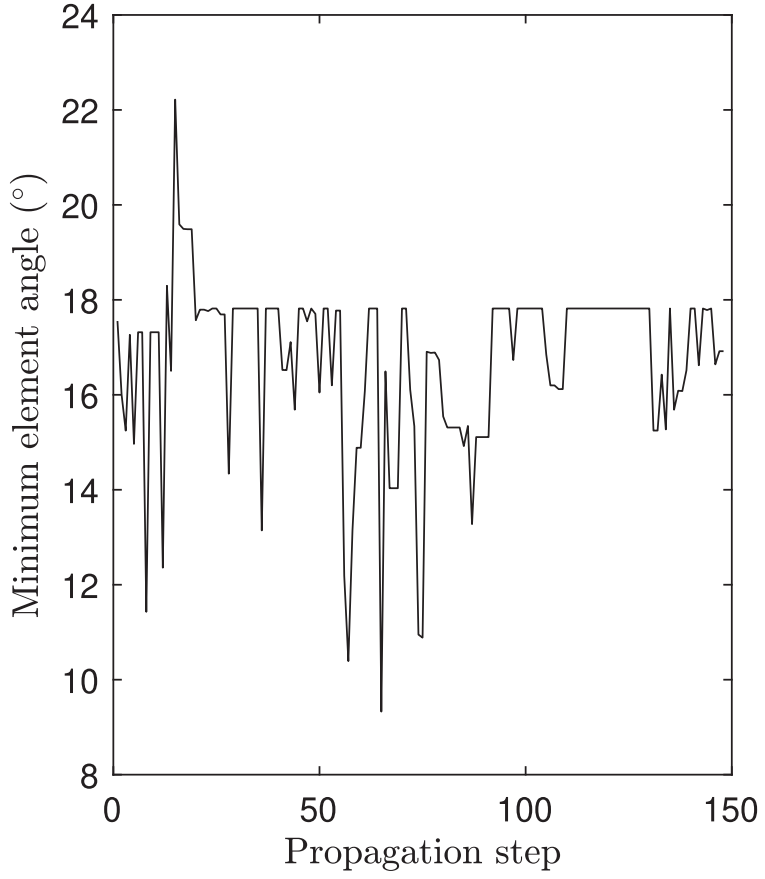


FIGURE 16 Y-crack: Element quality for the derefinement analysis

The maximum, average and minimum angle observed in the mesh is 22.2°, 16.7°, and 9.3° respectively. Occasionally the minimum angle falls to a value below 12° in a single propagation step. However in all occasions the minimum angle recovers to a value greater than 16° in either the next, or next couple, or propagation steps. There are two observations here. First, the *hr*-adaptive propagation algorithm is able to recover with propagation step when a poor element exists. The second observation is that the *hp*-adaptive algorithm for the CF computation is sufficiently robust that the poor quality elements do not affect the results; this is observed through the crack path precision plot, Figure 14B.

6.4 | Tree crack

The final numerical example is the tearing of a cantilever beam containing a tree crack with 8 crack tips, which is used to demonstrate that the proposed algorithm is able to deal with numerous fractures propagating at different rates. This problem is similar to that analyzed by Bird et al.,[39] Ai and Augarde,[78] and Ai[76] but there are some key differences in terms of the applied boundary conditions. The geometry of the problem is shown in Figure 17A, where $L = 20$ m, $H = 4$ m, $a = 1$ m, $b = 0.5$ m, and $\theta = \pi/4$ (note that the crack has been enlarged within the beam for clarity). The end of the beam is subject to a tearing traction (as seen in Figure 17A) at its free end which insures that all of the cracks open when loaded, specifically traction loads of $\sigma_i^N = [1\ 1]^T \times 10^{-4}$ Pa and $\sigma_i^N = [1\ -1]^T \times 10^{-4}$ Pa were applied above and below the mid depth of the beam, respectively. The opposite end of the beam is fully fixed. The material of the beam acts in plane stress with a Young's modulus of 1 Pa and a Poisson's ratio of 0.3. The initial mesh used to analyses the problem is shown in Figure 17B, where the true geometry of the tree crack is shown by the thick red lines and the crack tips highlighted by white circles. All elements in the initial mesh have $p_K = 2$, has the *hp*-adaptive parameters $\delta_2 = 0.7$ and $\delta_1 = 0.07$ and the CF tolerance for propagation is set as $\theta = 0.01$. It is worth highlighting that other researchers have analyzed the same crack geometry but with spurious results as their applied loading causes some cracks to close and overlap as they do not enforce contact conditions on the crack faces.[76,78]

In order to analyses this problem it is necessary to include Paris' law[79] so that the various cracks propagate at a rate consistent with the stress intensity at each of the crack tips. The Paris law crack growth rate is defined as

$$\frac{da}{dN} = C(\Delta K_{\text{equ}})^m, \tag{41}$$

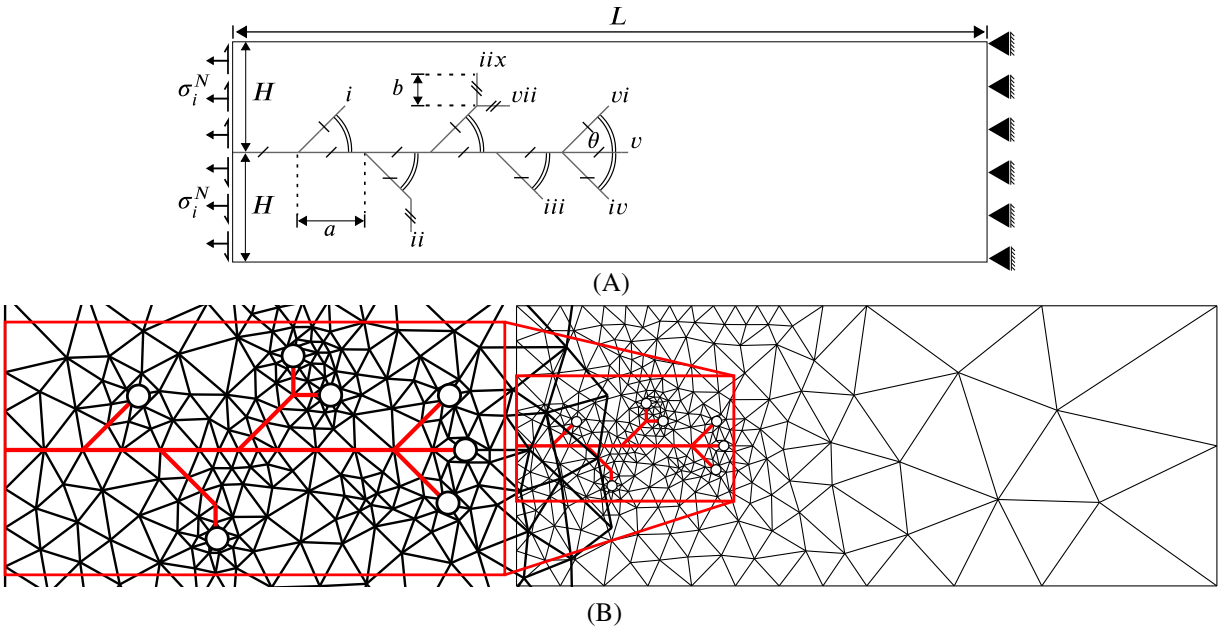


FIGURE 17 Tree-crack: (A) Schematic problem geometry and (B) initial mesh and true crack geometry (thick red lines) with the crack tips shown by white-filled circles

where a is the crack length, N is the number of load/stress cycles, and C and m are the Paris law material constants set to 1 and 2 respectively. Last K_{equ} is the equivalent SIF which can be obtained from the CF as

$$\Delta K_{\text{equ}} = \sqrt{|\boldsymbol{g}_{\max}|/E} - \sqrt{|\boldsymbol{g}_{\min}|/E}, \tag{42}$$

where $\boldsymbol{g}_{\max}$ and $\boldsymbol{g}_{\min}$ are the maximum and minimum CF values at the crack tip through the load cycle. E is the Young's modulus of the material. Since the load cycles between a maximum of $|\boldsymbol{\sigma}^N| = [1\ 1]^T \times 10^{-4}$ and minimum of $|\boldsymbol{\sigma}^N| = [0\ 0]^T \times 10^{-4}$, $\boldsymbol{g}_{\min}$ has a value of zero. In to order to propagate the cracks in steps, the discrete form of (41) is required, and found through integration,[80]

$$\frac{\Delta a}{C(\Delta K_{\text{equ}})^m} = \Delta N. \tag{43}$$

To propagate multiple cracks in the mesh, the crack propagation step Δa for the crack tip associated with the largest configuration force value is kept constant and defined $\Delta a_{\max}$. The propagation step of the remain cracks is found through the ratio of the discrete propagation step of the crack with the maximum CF and the current crack i,

$$\Delta a_i = \Delta K^m_{\text{equ,i}} \frac{\Delta a_{\max}}{K^m_{\text{equ,max}}}, \tag{44}$$

where propagation occurs at the same ΔN for all cracks. If Δa_i is predicted to be less than $0.01 \times \Delta a_{\max}$ then the crack is not propagated. For this problem $L_t = 0.25$ m and $L_l = 0.5$ m and L_t for the crack tip with the maximum configuration force is set equal to $\Delta a_{\max} = 0.05$ m. Last, 60 propagation steps occur.

The final mesh after 60 propagation steps is shown in Figure 18A. The figure shows that the *hr*-adaptive propagation method is able to propagate a large number of cracks all at different rates, as shown in Figure 18B. The precise propagation paths are provided in Figure 19A,B (note that the axes for the figures are of different scales). The first feature to observe is that the cracks that are inclined and furthest away from the cantilever, cracks *iv* and *vi*, experience the greatest CF value and thus propagate the furthest. However, as shown in Figure 18B, towards the second half of the simulation crack *iv* dominates in terms of crack growth rate. This is due to crack *iix* having a stress shielding effect on *vi*. Cracks *i* and *vii* propagate the least number of steps, respectively 0 and 2 steps. Crack *i* is closest to the loading of the beam and thus experiences the least moment, whilst crack *vii* is shadowed by crack *iix* from the load. Last, cracks *ii*, *iii*, *iix*, and *v* all propagate a non-negligible distance, however the solution is dominated by cracks *iv* and *vi* since they initially travel the

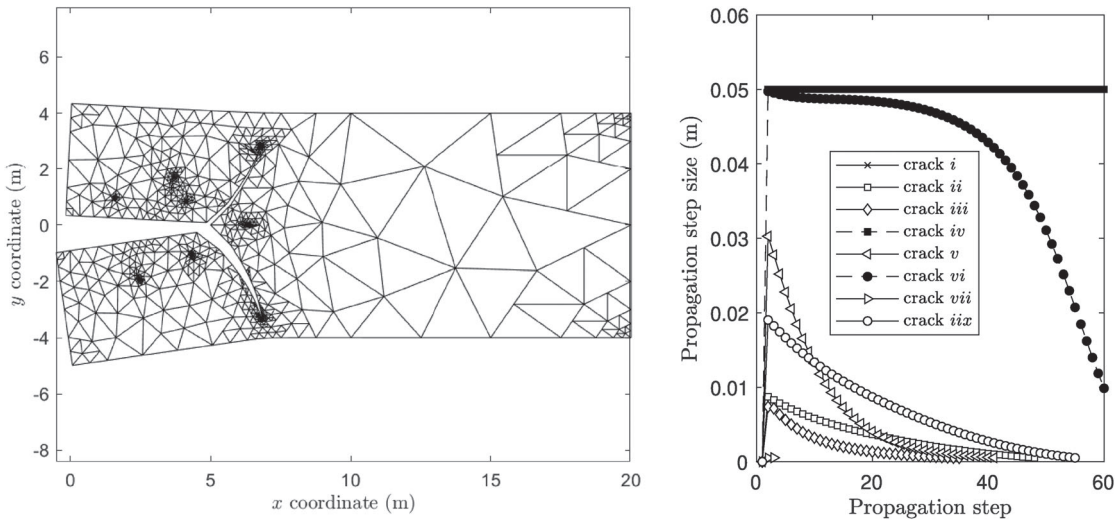


FIGURE 18 Tree-crack: (A) Deformed finite element mesh after 60 propagation steps (2 times displacement magnification) with corresponding crack step size at each propagation step (B)

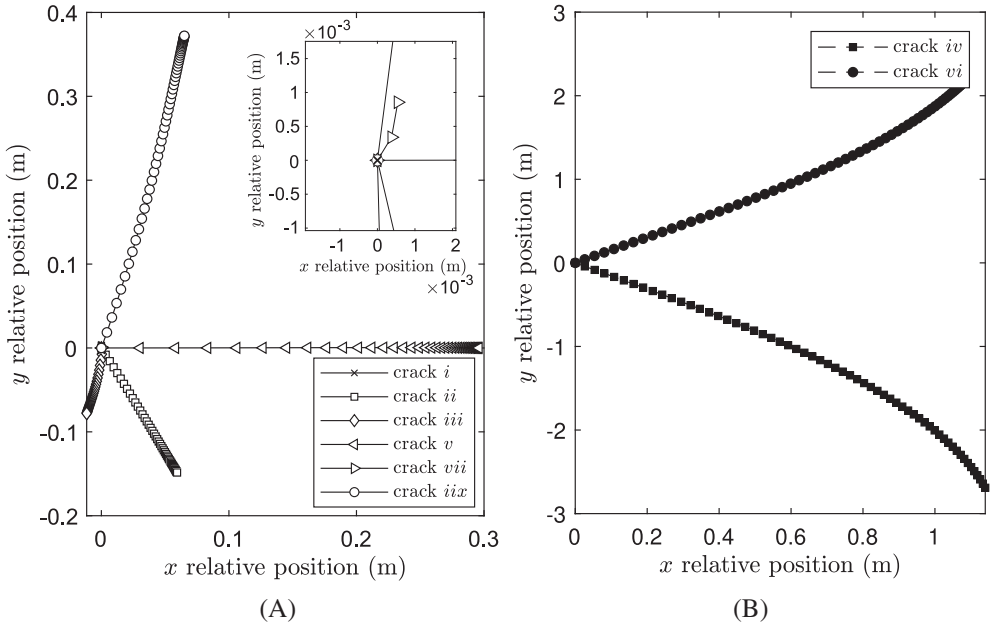


FIGURE 19 Tree-crack: Crack propagation paths relative to their initial position: (A) Cracks *i–iii*, *v*, *vii*, and *iix* where the inset figure shows a zoomed in view at the crack tips; and (B) cracks *iv* and *vi*

fastest and towards the boundary which has a positive feedback effect of increases their stress intensity at increasing rates, compared to the other crack tips.

7 | CONCLUSION

This article has presented a CF *hpr*-adaptive propagation method which achieved highly accurate predictions of the crack paths of multiple cracks propagating at different rates. The *hpr*-adaptive approach is formed from two algorithms: the first is a *hp*-adaptive method which can achieve highly accurate predictions of the CF at the crack tip; and second a *hr*-adaptive method, which is a computationally efficient propagation algorithm where the propagation step size is independent of the mesh.

The *hp*-adaptive procedure is driven by a residual based *a posteriori* error estimator and was used to obtain highly accurate values of the CF at the crack tips. The error in the crack CF is bound by two error estimators, the area integral component was shown to be reliable in Reference 39, whilst the error estimator for the edge integral component was shown to reliable in this article. Both error estimators have also been shown be efficient using numerical examples. In addition to this, the edge integral component of the CF was shown analytical to be reliable. The error estimators were then used to define a stopping criterion for the CF calculation which has two uses: (i) from a static point of view, it means that the CF can be computed up to a user desired accuracy; and (ii) for quasi-static crack propagation the CF can be computed up to a desired accuracy for each propagation step.

The developed *hr*-adaptive method is independent of the finite element scheme and criteria used to determine and how and when cracks propagate. It is designed to be simple to implement, for example since the crack propagation paths at each time step are determined from the same initial stencil no special case derefinement and/or refinement strategies are required. As well as being simple to implement, the *hr*-adaptive method is efficient, and was shown to be through numerical examples. DOF are efficiently used as only the crack path near the crack tip is captured to a high fidelity with small element edges whilst away from the crack tip the crack path is captured coarsely with larger elements. It was shown through numerical examples that negligible accuracy is lost in the propagation path prediction when only the crack path at the crack tip is modeled accurately whilst, there is a substantial gain in performance.

Finally, this article has included a range of numerical results which explored: (i) the accuracy of the *hpr*-adaptive method; (ii) the importance, efficacy, and accuracy of employing a derefinement scheme; and (iii) demonstrate the ability of the *hr*-adaptive method to deal with problems with a large number of cracks propagating at different rates. The

highly accurate nature of the method has allowed the re-examination of widely used benchmark solutions, such as the shear crack problem. It was shown that this is a highly sensitive problem in which any slight numerical heterogeneity causes a mixed mode propagation path. Inspection of the predicted crack path and the Eshelby stress tensor terms has shown that the shear crack problem is a pure mode II case and not, as prolific in the literature, a mixed mode problem and a horizontal crack propagation path is the only correct solution once spurious artefacts have been eliminated.

ACKNOWLEDGMENTS

This work was supported by the Engineering and Physical Sciences Research Council [Grant number EP/M507854/1]. The authors acknowledge the support of the European Commission-funded RISE-project BESTOFRAC (734370). All data produced in this article is available at https://doi.org/10.14469/hpc/9999.

ENDNOTES

*r-adaptivity refers to the movement of nodes of the finite element mesh to improve the solution and/or adapt to changes in geometry, such as fracture propagation.

†We avoid enriched approaches as they can limit the applicability of the numerical method in terms of the materials that can be analyzed (due to assumptions about material behavior in the additional basis functions) and the size of the analysis/number of fractures due to linear system stability issues.

‡It should be mentioned here that the presence of a discontinuity in a finite element mesh will induce non-zero CFs at nodes that are in the vicinity, but not at, the crack tip. These CFs are caused by the inability of the finite element basis functions to represent the displacement (and stress) field around the discontinuity.

§The average boundary condition approach, as detailed in Reference 39, is a Lagrange multiplier method that introduces three additional global unknowns linked to constraining rigid body translation in the x and y directions and rigid body rotation within the x-y plane. Many researchers instead choose to fix a single point in the problem to restrain rigid body motion but this introduces a spurious singularity that is not present in the true physical, or in this case analytical, problem which will destroy convergence towards the true solution, especially in terms of the level of accuracy achieved in Reference 39 and in this article.

¶This ratio is known as the effectivity index.

#As shown in Reference 39, the solution space of the stress on the crack faces at the crack tip is outside the finite element solution space.

‖The hp-adaptive component for determining the CF and the hr-adaptive method for propagating the crack through the mesh.

**The mode was found by splitting the continuous data into 100 equally spaced bins and then, through inspection of the corresponding histogram, ensuring that there was one distinct peak.

ORCID

Robert Bird https://orcid.org/0000-0001-7984-3198

REFERENCES

1. Eshelby JD. The force on an elastic singularity. *Phil Trans R Soc A*. 1951;224(877):87-112.
2. Eshelby J. Energy relations and the energy-momentum tensor in continuum mechanics; 1999:82-119; Springer, Berlin, Heidelberg.
3. Rice J. A path independent integral and the approximate analysis of strain concentration by notches and cracks. *J Appl Mech*. 1968;35(2):379-386.
4. Gurtin M. *Configurational Forces as Basic Concepts of Continuum Physiscs*. Springer-Verlag; 2000.
5. Gurtin M, Podio-Guidugli P. Configurational forces and the basic laws for crack propagation. *J Mech Phys Solids*. 1996;44:905-927.
6. Maugin G, Trimarco C. Pseudomomentum and material forces in nonlinear elastic: variational formulations and applications to brittle fracture. *Acta Mech*. 1992;94:1-28.
7. Maugin G. Material force: concepts and applications. *Appl Mech Rev*. 1995;23:213-245.
8. Kienzler R, Herrmann G. *Mechanics in Material Space with Applications to Defect and Fracture Mechanics*. Springer-Verlag; 2000.
9. Steinmann P, Maugin G. *Mechanics of Material Forces*. Springer-Verlag; 2005.
10. Mueller R, Maugin G. On material forces and finite element discretizations. *Comput Mech*. 2002;29(1):52-60.
11. Larsson R, Fagerström M. A framework for fracture modelling based on the material forces concept with XFEM kinematics. *Int J Numer Methods Eng*. 2005;62(13):1763-1788.
12. Fagerström M, Larsson R. Theory and numerics for finite deformation fracture modelling using strong discontinuities. *Int J Numer Methods Eng*. 2006;66(6):911-948.
13. Miehe C, Gürses E, Birkle M. A computational framework of configurational-force-driven brittle fracture based on incremental energy minimization. *Int J Fract*. 2007;145(4):245-259.
14. Miehe C, Gürses E. A robust algorithm for configurational-force-driven brittle crack propagation with r-adaptive mesh alignment. *Int J Numer Methods Eng*. 2007;72(2):127-155.
15. Gürses E, Miehe C. A computational framework of three-dimensional configurational-force-driven brittle crack propagation. *Comput Methods Appl Mech Eng*. 2009;198(15):1413-1428.

16. Kaczmarczyk Ł, Nezhad M, Pearce C. Three-dimensional brittle fracture: configurational-force-driven crack propagation. *Int J Numer Methods Eng*. 2014;97(7):531-550.
17. Runesson K, Larsson F, Steinmann P. On energetic changes due to configurational motion of standard continua. *Int J Solids Struct*. 2009;46(6):1464-1475.
18. Tillberg J, Larsson F, Runesson K. On the role of material dissipation for the crack-driving force. *Int J Plast*. 2010;26(7):992-1012.
19. Näser B, Kaliske M, Müller R. Material forces for inelastic models at large strains: application to fracture mechanics. *Comput Mech*. 2007;40(6):1005-1013.
20. Näser B, Kaliske M, Dal H, Netzker C. Fracture mechanical behaviour of visco-elastic materials: application to the so-called dwell-effect. *ZAMM-J Appl Math Mech/Zeitschrift für Angewandte Mathematik und Mechanik*. 2009;89(8):677-686.
21. Özenç K, Kaliske M, Lin G, Bhashyam G. Evaluation of energy contributions in elasto-plastic fracture: a review of the configurational force approach. *Eng Fract Mech*. 2014;115:137-153.
22. Irwin G. Analysis of stresses and strains near the end of a crack traversing a plate. *SPIE Milestone Ser MS*. 1957;137:167-170.
23. Ishikawa H, Kitagawa H, Okamura H. J-integral of a mixed mode crack and its application. *Mech Behav Mater*. 1980;3:447-455.
24. Yan A, Nguyen-Dang H. Multiple-cracked fatigue crack growth by BEM. *Comput Mech*. 1995;16(5):273-280.
25. Erdogan F, Sih G. On the crack extension in plates under plane loading and transverse shear. *J Basic Eng*. 1963;85(4):519-525.
26. Bouchard PO, Bay F, Chastel Y. Numerical modelling of crack propagation: automatic remeshing and comparison of different criteria. *Comput Methods Appl Mech Eng*. 2003;192(35):3887-3908.
27. Saouma V, Ayari M, Leavell D. Mixed mode crack propagation in homogeneous anisotropic solids. *Eng Fract Mech*. 1987;27(2):171-184.
28. Sih G, Macdonald B. Fracture mechanics applied to engineering problems-strain energy density fracture criterion. *Eng Fract Mech*. 1974;6(2):361-386.
29. Destuynder P, Djaoua M, Lescure S. Quelques remarques sur la mécanique de la rupture élastique. *Journal de Mécanique Théorique et Appliquée*. 1983;2(1):113-135.
30. deLorenzi H. On the energy release rate and the J-integral for 3-D crack configurations. *Int J Fract*. 1982;19:183-193.
31. Parks D. A stiffness derivative finite element technique for determination of crack tip stress intensity factors. *Int J Fract*. 1974;10(4):487-502.
32. Sutula D, Kerfriden P, Van Dam T, Bordas SP. Minimum energy multiple crack propagation. Part-II: discrete solution with XFEM. *Eng Fract Mech*. 2018;191:225-256.
33. Riviere B. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*. SIAM; 2008.
34. Hansbo A, Hansbo P. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Comput Methods Appl Mech Eng*. 2002;191(47):5537-5552.
35. Hansbo A, Hansbo P. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Comput Methods Appl Mech Eng*. 2004;193(33):3523-3540.
36. Heintz P. On the numerical modelling of quasi-static crack growth in linear elastic fracture mechanics. *Int J Numer Methods Eng*. 2006;65(2):174-189.
37. Bird R, Coombs W, Giani S. A quasi-static discontinuous Galerkin configurational force crack propagation method for brittle materials. *Int J Numer Methods Eng*. 2017;113(7):1061-1080.
38. Abedi R, Omidi O, Enayatpour S. A mesh adaptive method for dynamic well stimulation. *Comput Geotech*. 2018;102:12-27.
39. Bird R, Coombs W, Giani S. On the accurate determination of the configurational force for brittle fracture. *Eng Fract Mech*. 2019;216:106370.
40. Bird R. *A Hp-Adaptive Discontinuous Galerkin Finite Element Method for Accurate Configurational Force Brittle Crack Propagation*. PhD thesis. Durham University, Durham, UK; 2020.
41. Shen Y, Lew A. An optimally convergent discontinuous Galerkin-based extended finite element method for fracture mechanics. *Int J Numer Methods Eng*. 2010;82(6):716-755.
42. Mergheim J, Kuhl E, Steinmann P. A hybrid discontinuous Galerkin/interface method for the computational modelling of failure. *Commun Numer Methods Eng*. 2004;20(7):511-519.
43. Huang H, Costanzo F. On the use of space-time finite elements in the solution of elasto-dynamic fracture problems. *Int J Fract*. 2004;127(2):119-146.
44. Radovitzky R, Seagraves A, Tupek M, Noels L. A scalable 3D fracture and fragmentation algorithm based on a hybrid, discontinuous Galerkin, cohesive element method. *Comput Methods Appl Mech Eng*. 2011;200(1):326-344.
45. Nguyen VP. Discontinuous Galerkin/extrinsic cohesive zone modeling: Implementation caveats and applications in computational fracture mechanics. *Eng Fract Mech*. 2014;128:37-68.
46. Wu L, Becker G, Noels L. Elastic damage to crack transition in a coupled non-local implicit discontinuous Galerkin/extrinsic cohesive law framework. *Comput Methods Appl Mech Eng*. 2014;279:379-409.
47. Prechtel M, Ronda P, Janisch R, et al. Simulation of fracture in heterogeneous elastic materials with cohesive zone models. *Int J Fract*. 2011;168(1):15-29.
48. Abedi R, Hawker M, Haber R, Matouš K. An adaptive spacetime discontinuous Galerkin method for cohesive models of elastodynamic fracture. *Int J Numer Methods Eng*. 2010;81(10):1207-1241.
49. Tago J, Cruz-Atienza VM, Virieux J, Etienne V, Sánchez-Sesma FJ. A 3D hp-adaptive discontinuous Galerkin method for modeling earthquake dynamics. *J Geophys Res Solid Earth*. 2012;117(B9):1-21.
50. Park K, Paulino G. Cohesive zone models: a critical review of traction-separation relationships across fracture surfaces. *Appl Mech Rev*. 2011;64(6):060802(20).

51. Hussein A, Aldakheel F, Hudobivnik B, Wriggers P, Guidault PA, Allix O. A computational framework for brittle crack-propagation based on efficient virtual element method. *Finite Elem Anal Des*. 2019;159:15-32.
52. Areias P, Reinoso J, Camanho P, De J, Rabczuk T. Effective 2D and 3D crack propagation with local mesh refinement and the screened Poisson equation. *Eng Fract Mech*. 2018;189:339-360.
53. Gibert G, Prabel B, Gravouil A, Jacquemoud C. A 3D automatic mesh refinement X-FEM approach for fatigue crack propagation. *Finite Elem Anal Des*. 2019;157:21-37.
54. Thoutireddy P, Ortiz M. A variational r-adaption and shape-optimization method for finite-deformation elasticity. *Int J Numer Methods Eng*. 2004;61(1):1-21.
55. Scherer M, Denzer R, Steinmann P. On a solution strategy for energy-based mesh optimization in finite hyperelastostatics. *Comput Methods Appl Mech Eng*. 2008;197(6):609-622.
56. Heintz P, Larsson F, Hansbo P, Runesson K. Adaptive strategies and error control for computing material forces in fracture mechanics. *Int J Numer Methods Eng*. 2004;60(7):1287-1299.
57. Arranz A, Petrinic N, Süli E. Discontinuous element approximation for dynamic fracture; Vol. 1, 2005:545-548; Citeseer.
58. Kuhl E, Askes H, Steinmann P. An ALE formulation based on spatial and material settings of continuum mechanics. Part 1: generic hyperelastic formulation. *Comput Methods Appl Mech Eng*. 2004;193(39):4207-4222.
59. Solin P, Segeth K, Dolezel I. *Higher-Order Finite Element Methods*. CRC Press; 2003.
60. Epshteyn Y, Rivière B. Estimation of penalty parameters for symmetric interior penalty Galerkin methods. *J Comput Appl Math*. 2007;206(2):843-872.
61. Bird R, Coombs W, Giani S. A posteriori discontinuous Galerkin error estimator for linear elasticity. *Appl Math Comput*. 2019;344:78-96.
62. Eibner T, Melenk JM. An adaptive strategy for hp-FEM based on testing for analyticity. *Comput Mech*. 2007;39(5):575-595.
63. Heuer N, Mellado ME, Stephan EP. hp-Adaptive two-level methods for boundary integral equations on curves. *Computing*. 2001;67(4):305-334.
64. Irwin G. Linear fracture mechanics, fracture transition, and fracture control. *Eng Fract Mech*. 1968;1(2):241-257.
65. Denzer R, Barth FJ, Steinmann P. Studies in elastic fracture mechanics based on the material force method. *Int J Numer Methods Eng*. 2003;58(12):1817-1835.
66. Raju I, Shivakumar K. An equivalent domain integral method in the two-dimensional analysis of mixed mode crack problems. *Eng Fract Mech*. 1990;37(4):707-725.
67. Eischen J. An improved method for computing the J2 integral. *Eng Fract Mech*. 1987;26(5):691-700.
68. Li F, Shih C, Needleman A. A comparison of methods for calculating energy release rates. *Eng Fract Mech*. 1985;21(2):405-421.
69. Prudhomme S, Pascal F, Oden J, Romkes A. Review of a priori error estimation for discontinuous Galerkin methods; 2000.
70. Yau J, Wang S, Corten H. A mixed-mode crack analysis of isotropic solids using conservation laws of elasticity. *J Appl Mech*. 1980;47(2):335-341.
71. Khamayseh A, Kuprat A. Anisotropic smoothing and solution adaption for unstructured grids. *Int J Numer Methods Eng*. 1996;39(18):3163-3174.
72. Westergaard H. Bearing pressures and cracks. *J Appl Mech*. 1939;6(2):A49-A53.
73. Giani S, Hall E. An a posteriori error estimator for hp-adaptive discontinuous Galerkin methods for elliptic eigenvalue problems. *Math Models Methods Appl Sci*. 2012;22(10):1250030.
74. Wu JY, Nguyen V, Nguyen C, Sutula D, Sinaie S, Bordas S. Phase-field modeling of fracture. *Adv Appl Mech*. 2020;53:1-183.
75. Miehe C, Hofacker M, Welschinger F. A phase field model for rate-independent crack propagation: robust algorithmic implementation based on operator splits. *Comput Methods Appl Mech Eng*. 2010;199(45-48):2765-2778.
76. Ai W. *Computational Fracture Modelling by an Adaptive Cracking Particle Method*. PhD thesis. Durham University, Durham,UK; 2018.
77. Ai W, Bird R, Coombs W, Augarde C. A configurational force driven cracking particle method for modelling crack propagation in 2D. *Eng Anal Bound Elem*. 2019;104:197-208.
78. Ai W, Augarde C. A multi-cracked particle method for complex fracture problems in 2D. *Math Comput Simul*. 2018;150:1-24.
79. Paris P. A rational analytic theory of fatigue. *Trends Eng*. 1961;13:9-14.
80. Sajith S, Krishna Murthy K, Robi P. Prediction of accurate mixed mode fatigue crack growth curves using the Paris' law. *J Inst Eng (India) Ser C*. 2019;100(1):165-174.

How to cite this article: Bird R, Coombs WM, Giani S. Adaptive configurational force-based propagation for brittle and fatigue crack analysis. *Int J Numer Methods Eng*. 2022;1-37. doi: 10.1002/nme.6911

APPENDIX A. THE HR-ADAPTIVE ALGORITHM

This appended section is an algorithmic description of the *hr*-adaptivity method for capturing the crack path. It leads directly on from Section 5 which describes the three stages of *hr*-adaptive algorithm. The *hr*-adaptive process is described by Algorithm 4. The algorithm is governed by two loops: a for loop over all crack paths and a while loop over the crack

path which continues until the whole crack path has been approximated by the mesh. Each crack path is defined as $X^c = \{\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{n-1}, \boldsymbol{x}_n\}$, where c is the crack number, n is the number of crack propagation steps and the point $\boldsymbol{x}_n$ corresponds to the crack tip.

The algorithm starts by selecting the crack c in the outer for loop and subsequently setting up the variables for the first search of the intersection of the crack path with the element edges in the mesh: the points $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$ and the line that spans between them, the initial crack tip node n_c (which has coordinates $\boldsymbol{x}_0$), and the reference point R_p that is initiated as having the coordinates of the node n_c.

The point $\boldsymbol{x}_R$ is set the value $\boldsymbol{x}_0 \in X^c$, with its position corresponding to the initial crack tip node n_c. Next, the pair point P is set to 1, where P corresponds to a crack path position number in the set X^c, $\boldsymbol{x}_P = \boldsymbol{x}_1 \in X^c$. The straight length between the points $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$ forms the first search edge.

The while loop then operates to find suitable nodes and moving them to the crack path. This starts by checking whether the line between the points $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$ intersects the mesh. If no intersection is found the points are updated and a new search line is defined. This occurs on lines 29 and 30 of Algorithm 4, where P is increased by 1 and the first point of the edge is redefined as $\boldsymbol{x}_R = \boldsymbol{x}_{P-1}$. The algorithm then goes back to the start of the while loop and the new search line is checked to see whether it intersects the mesh. If the search edge intersects the mesh, the edge with the nearest intersection point to $\boldsymbol{x}_R$ is defined F and, the nearest node on F to the intersection point is moved to the intersection point. If the newly aligned edge is of an allowable edge length, no element is inverted and the node selected is a vertex node, the mesh is r-adapted as in Figure A1:

Step 1: Algorithm 4 lines 29–30: The search edge between the points $\boldsymbol{x}_P$ and $\boldsymbol{x}_R$ is defined as the dashed line and multiple element edge intersections are found.

Step 2: Algorithm 4 lines 7 and 10: In this case two (this is just illustrative—it could be one or more that two) intersections points are found between the search line that spans the points $\boldsymbol{x}_P$ and $\boldsymbol{x}_R$ and, element edges in the mesh, marked in grey. The nearest intersection point to the crack tip node is then selected and the nearest node n_2 along the element intersection edge F is defined n_I.

Step 3: Algorithm 4 lines 17–19: The element edge that exists between the node n_I and n_c is defined as cracked edge. n_I is then moved to the intersection point marked in grey and the point $\boldsymbol{x}_R$ is assigned the new coordinates of n_I. Last, the reference point R_p is updated to P and the new crack tip node is defined $n_c = n_I$. The algorithm has successfully assigned an element edge to the crack path and a can begin.

However, it is sometimes the case that a r-adaptive move either selects a hanging node, creates an inverted element or, aligns an element edge with the crack path that is too long, as shown in Section 5.1.1. This makes the method to move a node onto the crack path more complex than that described in Figure A1. An example of Algorithm 4 managing the constraint of an element edge along the crack path being too long is shown in Figure A2, requiring four steps:

Step 1: Algorithm 4 lines 7, 29, and 30: The search edge, defined between the points $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$ finds no intersection. P is increased by one and then the point $\boldsymbol{x}_R$ is set equal $\boldsymbol{x}_{P-1}$.

Step 2: Algorithm 4 lines 7 and 15: The new search line that spans the point $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$ intersects the mesh. However when checked if this node move can occur, on line 15, the length of the line L that connects the crack tip node and the intersection point is too long.

Step 3: Algorithm 4 lines 24–26 and 7: Since L is too long, the element that contains the current crack tip node and the edge on which the intersection point resides is refined producing nodes n_5, n_6, and n_7. The intersection search is then restarted by setting $P = R_p$, and defining the $\boldsymbol{x}_R$ as equal to the point of the current crack node n_c (which is n_1 in this case). With this new search edge and element mesh, the intersection point, marked in grey, can be determined.

Step 4: Algorithm 4 lines 10, 17–19: The nearest node along the intersection edge F (n_5) to the intersection point is defined n_I. Since the move of n_I to the intersection point does not break any constraints, the element edge that exists between n_c and n_I is defined a cracked edge. The reference point number is updated $R_p = P$. n_I is then moved to the intersection point marked in grey and the point $\boldsymbol{x}_R$ is assigned the new coordinates of n_I. Last, the reference point R_p is updated to P and the new crack tip node is defined $n_c = n_I$. The algorithm has successfully assigned an element edge to the crack path and a can begin.

Algorithm 4. *hr*-adaptivity

for Loop over all crack paths **do**
  Define the first point for the edge intersect to $\boldsymbol{x}_R = \boldsymbol{x}_0$
  Set the current crack tip node, n_c, to the node that resides on $\boldsymbol{x}_0$
  Set the nodal pair search point $P = 1$
  Reference point number $R_p = 1$
  while exit flag = 0 **do**
    Determine if there is an intersection between the points $\boldsymbol{x}_R$ and $\boldsymbol{x}_P$
    if Intersection is found OR the $\boldsymbol{x}_P$ is the crack tip **then**
      if Intersection found **then**
        Find the nearest node n_I on the edge F that the intersection point lies on.
      else
        Find the nearest node n_I to the point $\boldsymbol{x}_N$.
      end if
      Set the maximum element edge $L_{\max}$
      Check if the move can occur with the constrains defined in Section 5.1.1
      if Node move can occur **then**
        Define the element edge that spans the node n_I and n_c as a cracked edge
        Move the node n_I to the intersection point and define as the crack tip node n_c
        Set $R_p = P$ and $\boldsymbol{x}_R$ to the coordinates of the new crack tip node n_c
        if The last node move is the last point **then** exit flag = 1
        end if
      else
        Refine the elements that prevent the node move
        Set $P = R_p$.
        Set $\boldsymbol{x}_R$ to the position of the crack tip node n_c
      end if
    else
      Set $P = P + 1$
      Set $\boldsymbol{x}_R = \boldsymbol{x}_{P-1}$
    end if
  end while
  Make a vector $\mathcal{T}_{\min}$ of all K where the minimum angle criterion is false
  for All $K \in \mathcal{T}_{\min}$ **do**
    Define F, the element edge that is opposite to the corner of the K with the maximum angle
    Find the two elements K^+ and K^- on F
    if F is conforming, not on any boundary, and has not been previously flipped **then**
      Flip the edge F so it now spans the two alternative internal corners of K^+ and K^-.
      if Edge flip does not improve the quality of both K^+ and K^-. **then**
        Undo edge flip
      end if
    end if
  end for
end for

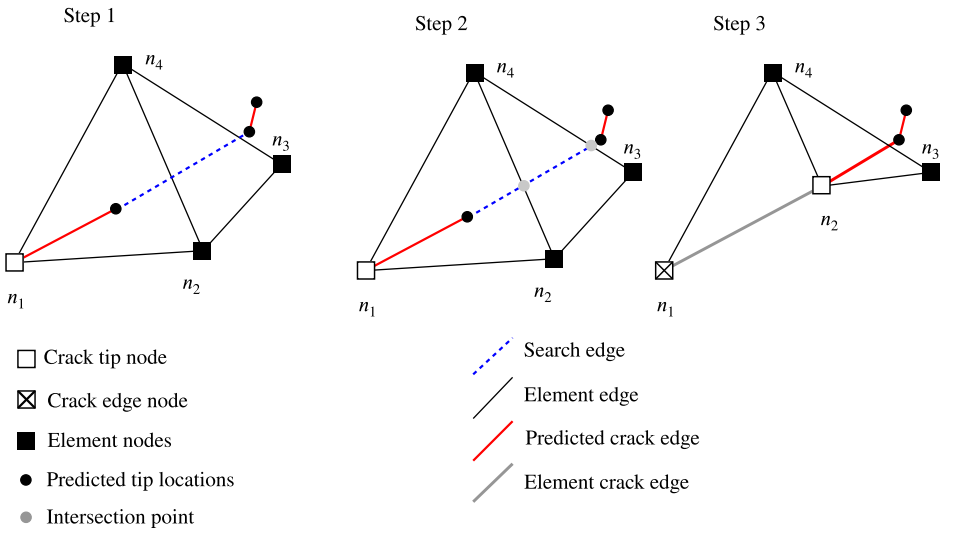


FIGURE A1 An example of an *r*-adaptive move, using Algorithm 4, where a node move does not break the constraints defined in Section 5.1.1

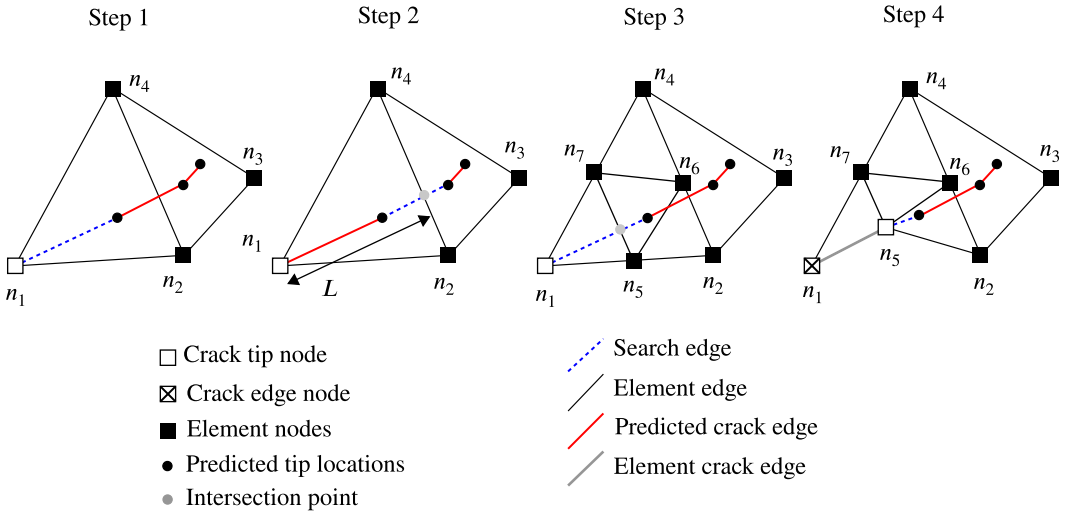


FIGURE A2 An example of an *r*-adaptive move, using Algorithm 4, where a node move does break the constraints defined in Section 5.1.1 and *h*-refinement has to occur

The only difference in the way the *hr*-adaptive algorithm functions to overcome different constraints is through the choice of elements that are *h*-refined, as in Step 3, as explained in Section 5.1.1.

The final part of the algorithm is moving a node in the mesh to the crack tip, this corresponds to Stage 3 in Section 5.1.1. This part of the algorithm is instigated when the search line has the point $\boldsymbol{x}_P$ equal to last point in the crack path, that is, $P = n$. When $P = n$, a node move is attempted even if there is no intersection between the points of the search line and the element edges in the mesh. If $\boldsymbol{x}_P = \boldsymbol{x}_n$ and there is an intersection, a node is moved to the point of intersection. However, if there is no intersection, the nearest node of the element that the point $\boldsymbol{x}_n$ resides in is moved to the point $\boldsymbol{x}_n$, the nearest node could also include the node n_c. One of the requirements highlighted in Stage 3 in Section 5.1.1, is that for there is always at least one crack node between the penultimate and last crack path points. This is achieved on line 14 of Algorithm 4, by setting the maximum edge length allowed when $P = n$ to

$$L_{\max} = \min\{L_s, (2/3) \times |\boldsymbol{x}_{n-1} - \boldsymbol{x}_n|\},$$

which enforces at least node residing between the penultimate and last crack point. A case when this occurs is shown in Figure A3 and comprises of seven steps:

Step 1: Algorithm 4 lines 7, 10, 14, 15, 24–26: The last point in the crack path is being considered but, there is an intersection between the search line and the mesh. However as $L > L_{\max}$, no move occurs and the element is refined. The counter P is reset to R_p and the first point of the search line, $\boldsymbol{x}_R$, is set to the position of the current tip node, n_c (which in this case is node number n_1).

Step 2: Algorithm 4 lines 7, 10, 14, and 15: The last point in the crack path is no longer being considered and there is an intersection between the search line and the mesh, marked in grey. The node n_5 is the nearest node on the intersected edge F to the intersection point, the intersection node is defined $n_I = n_5$. Moving the node n_I to the intersection point does not break any constraints.

Step 3: Algorithm 4 lines 17–19, 7, and 29–30: The new crack edge can be defined between n_c and n_I, the crack node and R can be updated, $n_c = n_I$ and $R = P$ respectively, and the first point of the search line, $\boldsymbol{x}_R$, is set to coordinates of n_c. The new search line finds no intersection, and so P is increased by 1 and $\boldsymbol{x}_R = \boldsymbol{x}_{P-1}$.

Step 4: Algorithm 4 lines 7, 10, 14, 15, 17–19: The last point in the crack path is being considered ($P = n$), there is an intersection between the search but unlike Step 1, $L < L_{\max}$, and so a node move can occur. The node n_6 is the nearest node on the intersected face F to the intersection point and as such the intersection node is defined $n_I = n_6$. Since moving n_I does not break any constraints outline in Section 5.1.1, the mesh can be updated using lines 17–19.

Step 5: Algorithm 4 lines 7, 8, 12, 15, 24–26: There is no intersection but since the last point in the crack path is being considered, $P = n$, the if statement on line 8 is true. The nearest node to the point $\boldsymbol{x}_n$ is identified as n_4 however, moving it to the point $\boldsymbol{x}_n$ results in an inverted element. Therefore, the element in which $\boldsymbol{x}_n$ resides in is refined, and the search edge variables are reset, $P = R_p$ and $\boldsymbol{x}_R$.

Step 6: Algorithm 4 lines 7, 8, 12, 17–19: There is no intersection but since the last point in the crack path is being considered, $P = n$, the if statement on line 8 is true. The nearest node to the point $\boldsymbol{x}_n$ is identified as n_8 and set to n_I. The node n_I is moved to the last point in the crack path $\boldsymbol{x}_n$; this move is allowed since no constraints are broken. The edge that spans nodes n_c and n_I is made into a cracked edge

Step 7: Algorithm 4 line 21: Since the crack node n_c now coincides with the last point in the crack path $\boldsymbol{x}_n$ the exit flag is set to 1, and the while loop for this crack exits.

This appended section has detailed the *hr*-adaptive algorithm, Algorithm 4, for aligning element edges with crack faces from multiple propagating crack tips. This algorithm has implemented and used to provide the numerical validations and results in Section 6 for both brittle and fatigue driven fracture.

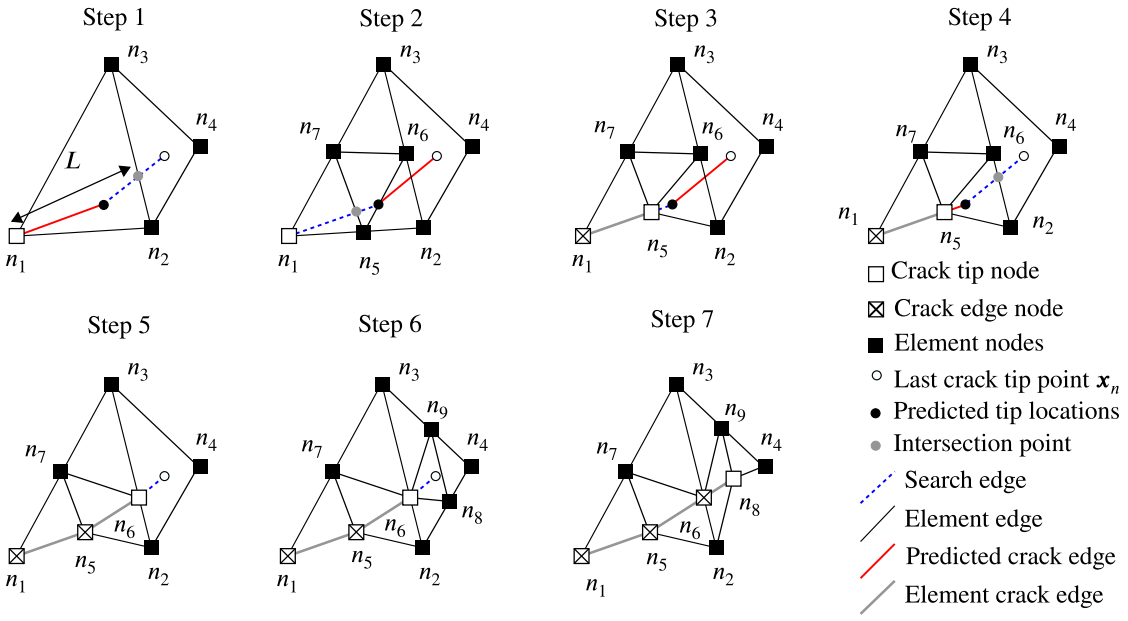


FIGURE A3 An example of resolving the mesh about the crack tip using *hr*-adaptivity