

Three-dimensional finite element modeling of ductile crack initiation and propagation

Citation for published version (APA):

Javani, H. R., Peerlings, R. H. J., & Geers, M. G. D. (2016). Three-dimensional finite element modeling of ductile crack initiation and propagation. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1), 1-25. [19].
<https://doi.org/10.1186/s40323-016-0071-y>

DOI:

[10.1186/s40323-016-0071-y](https://doi.org/10.1186/s40323-016-0071-y)

Document status and date:

Published: 01/12/2016

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

RESEARCH ARTICLE

Open Access



Three-dimensional finite element modeling of ductile crack initiation and propagation

H. R. Javani^{1,2}, R. H. J. Peerlings^{2*} and M. G. D. Geers²

*Correspondence:

R.H.J.Peerlings@tue.nl

²Department of Mechanical Engineering, Eindhoven University of Technology, PO Box 513, 5600 MB Eindhoven, The Netherlands

Full list of author information is available at the end of the article

Abstract

A crack initiation and propagation algorithm driven by non-local ductile damage is proposed in a three-dimensional finite strain framework. The evolution of plastic strain and stress triaxiality govern a non-local ductile damage field via constitutive equations. When the damage reaches a critical threshold, a discontinuity in the form of a crack surface is inserted into the three-dimensional continuum. The location and direction of the introduced discontinuity directly result from the damage field. Crack growth is also determined by the evolution of damage at the crack tip and the crack surface is adaptively extended in the computed direction. Frequent remeshing is used to computationally track the initiation and propagation of cracks, as well as to simultaneously maintain a good quality of the finite elements undergoing large deformations. This damage driven remeshing strategy towards fracture allows one to simulate arbitrary crack paths in three-dimensional evolving geometries. It has a significant potential for a wide range of industrial applications. Numerical examples are solved to demonstrate the ability of the proposed framework.

Keywords: Finite element method, Ductile fracture, Remeshing, Nonlocal damage

Background

Controlling crack initiation and propagation is one of the important aspects in maintaining the integrity of an engineering structure. In some other cases, however, cracks are introduced on purpose. Examples can be found in forming processes such as cutting or blanking. Computational models are indispensable for the predictive analysis of the mechanics of ductile fracture. Algorithms for dealing with two-dimensional (2D) crack propagation problems are by now well established. However, at present, three-dimensional (3D) problems cannot be analyzed routinely, particularly if they are accompanied by large (plastic) strains. This is due to the complex topology and geometry changes, accompanied with localized deformation and material degradation. At the same time, full 3D modelling of cracks provides a more realistic prediction tool for studying true 3D structures, as well as local features like crack tunneling, e.g. [1].

There is an extensive literature on modelling cracks in general. They can either be modelled in a continuous way, by degrading and/or deleting elements, or by introducing a true discontinuity. A discontinuity can be implicitly modelled by element or nodal enrich-

ment [2–14]. However, most of these methods are applicable for small displacements and cannot be directly applied for large deformations. In a second category of discontinuous approaches remeshing is used to explicitly model the discontinuity, i.e. by alignment of the mesh with the crack and nodal decoupling perpendicular to the crack [15–23].

Here we concentrate on the second category and extend a continuum damage mechanics approach to 3D crack initiation and propagation. Along these lines, Mediavilla et al. [22] suggested a continuous-discontinuous methodology for modeling cracks in 2D problems, in which the crack geometry is incorporated in the mesh by frequent remeshing. This algorithm is attractive especially when dealing with ductile failure, where large local deformations occur and remeshing is necessary even for the continuous part of the problem. Incorporating the additional geometrical changes due to crack growth then requires only a limited intervention in the algorithms used.

In this study, we develop an extension of Mediavilla et al.'s remeshing strategy to 3D problems in which damage growth and 3D crack propagation occur in a large deformation setting. Remeshing is used to deal with geometrical changes due to large deformations as well as crack growth [22]. Crack initiation and crack growth are governed by a continuum damage model which is intrinsically coupled to the underlying elasto-plastic constitutive model. The damage formulation is nonlocal (of the implicit gradient type) to ensure proper localization properties [24]. Once the damage reaches a critical level somewhere in the geometry, a discrete crack is introduced in the geometrical description of the body. This crack is extended when the damage field at its front becomes critical, whereby the orientation is governed by the direction of maximum nonlocal damage driving variable. As a result, no additional fracture criterion is required to control the crack growth. The crack surface is constructed by computing the propagation direction and distance for each node on the crack-front. By splitting the nodes along the crack surface, discontinuities are allowed along the element faces. Robustness of the simulations is ensured by temporarily applying the element internal forces as external forces on the crack nodes and gradually reducing them to zero.

In 3D, compared with the 2D case considered by Mediavilla et al. [22], the remeshing strategy which we follow presents a number of important additional computational challenges. (1) A reliable tetrahedral finite element is required to enable robust automatic remeshing of complex geometries. We adopt a bubble-enhanced mixed finite element formulation of the continuum model [25,26]. (2) An accurate transfer operator is required to map history data from one mesh to the next. Here special precautions need to be taken to ensure consistency between the transferred fields [27,28]. (3) Algorithms are needed to manipulate the 3D geometrical description of the problem upon initiation of a crack, as well as for every increment of crack growth. This is the main topic of the present paper.

The algorithm developed here is based on a geometrical description by a surface mesh, which is adapted according to the computed nonlocal damage field. To initiate a crack, elements with damage values higher than a critical limit are first identified. They form a cloud which is either completely inside the body or in contact with a surface. For internal clouds we use an averaging technique to compute the center of the cloud. This point is taken as the center of the emerging crack surface. Using the damage distribution, a plane is defined for inserting a discontinuity. For clouds which are in contact with an external boundary, a crack-front is constructed and this front is connected to the external surface by a discontinuity surface. When crack propagation is predicted by the damage

evolution ahead of a crack, that part of the surface mesh which represents the crack faces is extended. For this the strategy followed in 2D by Mediavilla et al. [22] is applied in planes perpendicular to the crack front. Care needs to be taken to ensure the consistency of the crack front and to respect the outer surface of the body. At all stages of the simulation, the damage field is also used in order to refine the discretisation in critical regions of the geometry. We illustrate the methodology by showing two numerical examples, one illustrating crack initiation inside a body (i.e. a rectangular bar under tension) and one at the surface (of a double notched specimen).

This paper is structured as follows. In the next section, the continuum damage model, element technology, remeshing and transfer are briefly reviewed. We then first present the 3D crack propagation algorithm, since elements of it are used in the crack initiation algorithm, which is subsequently discussed for internal as well as surface cracks. After presenting two numerical examples, we conclude by highlighting the newly added features of the algorithm.

Continuum model and finite element discretisation

In this section we briefly review the coupled plasticity-damage continuum modelling, as well as its FEM implementation, which forms the basis of our developments. For a more detailed discussion we refer to Ref. [26] and references cited therein.

Continuum nonlocal damage model

The balance of momentum can be expressed in terms of Kirchhoff stress τ as

$$\vec{\nabla} \cdot \left(\tau \frac{1}{J} \right) = \vec{0} \tag{1}$$

where $\vec{\nabla} \cdot$ represents the divergence operator (with respect to the current configuration) and $J = \det(\mathbf{F})$ is the volume change factor. The following boundary condition is applied on the free surfaces of the body considered and, in particular, also on the crack surfaces, see Fig. 1:

$$\vec{t} = \vec{n} \cdot \frac{\tau}{J} = \vec{0} \tag{2}$$

The Kirchhoff stress is related to the elastic deformation via the effective Kirchhoff stress tensor $\hat{\tau}$ as follows:

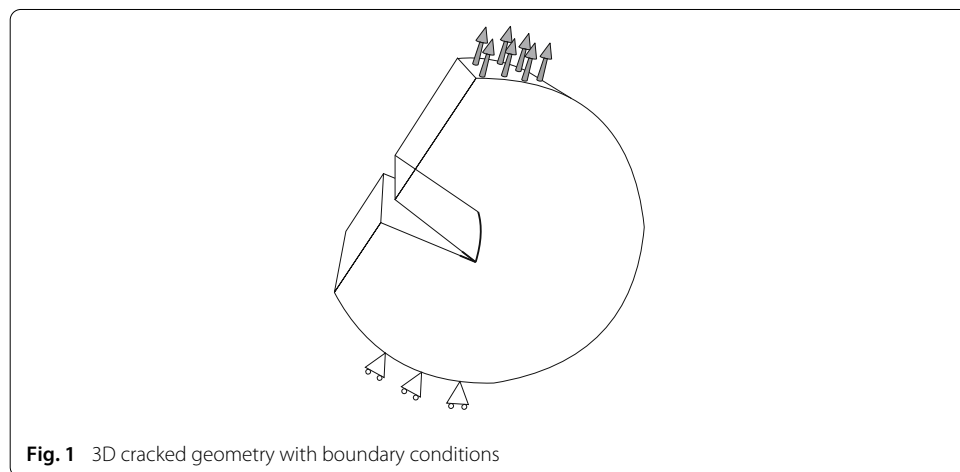


Fig. 1 3D cracked geometry with boundary conditions

$$\hat{\boldsymbol{\tau}} = \frac{\boldsymbol{\tau}}{(1 - \omega_p)} \tag{3}$$

$$\hat{\boldsymbol{\tau}} = \frac{1}{2} {}^4\mathbf{H} : \ln \mathbf{b}_e \tag{4}$$

$$\mathbf{b}_e = \mathbf{F}_e \cdot \mathbf{F}_e^T \tag{5}$$

where \mathbf{F}_e is elastic part of the multiplicatively split deformation gradient \mathbf{F} and ${}^4\mathbf{H}$ is the standard fourth order elasticity tensor; $\hat{\boldsymbol{\tau}}$ is the effective stress tensor due to the presence of the (isotropic) damage characterised by ω_p .

Expressed in terms of the effective stress space, the elastic domain is bounded by the following equation:

$$\phi(\hat{\boldsymbol{\tau}}, \hat{\tau}_y) = \hat{\tau}_{eq} - \hat{\tau}_y \leq 0 \tag{6}$$

where $\hat{\tau}_{eq} = \sqrt{\frac{3}{2} \hat{\boldsymbol{\tau}}^d : \hat{\boldsymbol{\tau}}^d}$ and $\hat{\tau}_y$ is the current yield stress. J_2 associative flow theory is used to model the plastic response of the material [29].

The evolution of the damage variable ω_p is governed by the following equations:

$$\dot{z} = h_z \dot{\varepsilon}_p \tag{7}$$

$$h_z = \left\langle 1 + A \frac{\hat{\tau}_h}{\hat{\tau}_{eq}} \right\rangle \varepsilon_p^B \quad \text{with } \langle x \rangle = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases} \tag{8}$$

$$\bar{z} - \ell^2 \nabla^2 \bar{z} = z, \quad \vec{\nabla} \bar{z} \cdot \vec{n} = 0 \tag{9}$$

$$\dot{\kappa} \geq 0, \quad \bar{z} - \kappa \leq 0, \quad \dot{\kappa}(\bar{z} - \kappa) = 0 \tag{10}$$

$$\dot{\omega}_p = h_\omega \dot{\kappa} \tag{11}$$

In these relations, z is a local damage driving variable, the evolution of which depends on the effective plastic strain ε_p and the (effective) stress triaxiality $\hat{\tau}_h/\hat{\tau}_{eq}$; A and B are material constants as introduced by Goijaerts et al. [30]. Equation (9) uses the local damage driving variable z together with a Neumann boundary condition (with normal \vec{n}) to compute a nonlocal damage driving variable \bar{z} , which controls the damage evolution. The use of this nonlocal quantity is necessary to regularise the localisation of deformation and damage, which would otherwise become pathological [31]. The final link to the damage evolution is made via a history variable, κ , and the evolution law (11).

Finite element form of the equations

In order to avoid locking effects due to isochoric plastic straining, the above constitutive model is implemented using a mixed formulation in a tetrahedral element [25,26]. Therefore, the constitutive law of Eq. (4) is split into a mixed pressure/deviatoric form by decomposing the effective stress tensor as $\hat{\boldsymbol{\tau}} = \hat{\tau}^h \mathbf{I} + \hat{\boldsymbol{\tau}}^d$. The weak forms of Eq. (1), the volumetric elastic law and Eq. (9) are then obtained by the usual weighted residuals arguments as:

$$\begin{aligned} \int_{\Omega} (\vec{\nabla} \bar{\varphi})^T : (\tau^h \mathbf{I} + \boldsymbol{\tau}^d) \frac{1}{J} d\Omega &= \int_S \bar{\varphi} \cdot \vec{t} dS \\ \int_{\Omega} \psi \left(\hat{\tau}^h - \frac{1}{2} K \mathbf{I} : \ln \mathbf{b}_e \right) d\Omega &= 0 \\ \int_{\Omega} \left(\chi \bar{z} + \ell^2 \vec{\nabla} \chi \cdot \vec{\nabla} \bar{z} - \chi z \right) d\Omega &= 0 \end{aligned} \tag{12}$$

where $\bar{\varphi}$, ψ and χ are weight functions corresponding to \vec{u} , $\hat{\tau}^h$ and \bar{z} .

It is well known that the weak form of the governing Eqs. (12) when used with equal order interpolation for the hydrostatic stress $\hat{\tau}^h$ and displacement \vec{u} is not stabilised. Stabilisation is performed by enriching the standard displacement with a displacement bubble, similar to the well known Mini element. The simplified version of this enrichment results in an enhanced strain so that the resulting algorithmic stress tensor reads [26]

$$\begin{aligned} {}^m\hat{\tau}^d &= \hat{\tau}^d + {}^4\mathbf{H}^d : \alpha \boldsymbol{\varepsilon}_b \\ {}^m\hat{\tau}^h &= \hat{\tau}^h + K\mathbf{I} : \boldsymbol{\varepsilon}_b \end{aligned} \tag{13}$$

where α is an element dependent stabilisation factor and $\boldsymbol{\varepsilon}_b$ denotes the symmetric part of the gradient of the bubble displacement:

$$\boldsymbol{\varepsilon}_b = {}^4\mathbf{I}^s : \vec{\nabla} \underline{N}_b^T \vec{u}_b \tag{14}$$

in which the column matrix \underline{N}_b contains one bubble shape function per element. Note that $\boldsymbol{\varepsilon}_b$ uses a small strain definition with respect to the deformed configuration given by the conventional part of the displacement interpolation. This conventional part, however, is fully objective and rigorously deals with large strains.

Details of the discretisation of the weak forms (12) and their linearization are omitted here; see Ref. [26] for a detailed derivation. Here, we merely summarize the resulting set of nonlinear algebraic equations for future reference. The mixed formulation of equilibrium, including the bubble stabilisation, results in a combined set of equations which can be written as:

$$\begin{aligned} \vec{F}_i(\vec{u}, \vec{u}_b, \hat{\tau}^h, \vec{z}) &= \vec{F}_e \\ \vec{F}_b(\vec{u}, \vec{u}_b, \hat{\tau}^h, \vec{z}) &= \vec{0} \\ F_\tau(\vec{u}, \vec{u}_b, \hat{\tau}^h, \vec{z}) &= 0 \end{aligned} \tag{15}$$

Likewise, the nonlocal Eq. (9) results in an additional set of equations as follows:

$$F_z(\vec{u}, \vec{u}_b, \hat{\tau}^h, \vec{z}) = 0 \tag{16}$$

In the above, we have

$$\begin{aligned} \vec{F}_i &= \int_{\Omega} \vec{\nabla} \underline{N} \cdot {}^m\hat{\tau} \frac{1-\omega_p}{J} d\Omega \\ \vec{F}_b &= \int_{\Omega} \vec{\nabla} \underline{N}_b \cdot {}^m\hat{\tau} \frac{1-\omega_p}{J} d\Omega \\ F_\tau &= \int_{\Omega} \underline{N} \left({}^m\hat{\tau}^h - K\mathbf{I} : \left[\frac{1}{2} \ln \mathbf{b}_e + \boldsymbol{\varepsilon}_b \right] \right) d\Omega \\ F_z &= \int_{\Omega} \left[\underline{N} \vec{z} + \ell^2 \vec{\nabla} \underline{N} \cdot \vec{\nabla} \vec{z} - \underline{N} \vec{z} \right] d\Omega \\ \vec{F}_e &= \int_{S_i} \underline{N} \vec{t} d\Gamma \end{aligned} \tag{17}$$

Equation (15.2) is the result of the introduced enrichment in Eq. (13). This equation is solved at the element level for the bubble displacements \vec{u}_b and therefore does not lead to additional global degrees of freedom.

Remeshing

Our strategy to deal with 3D crack growth, as well as the large deformations which we wish to model, necessitates frequent remeshing on a global level. After a predefined number of increments, the surface mesh of the 3D body is extracted from the model. If crack growth is detected, the surface mesh must be modified to incorporate a new crack segment, see the next section. Otherwise, the existing surface mesh is used as input to the 3D tetrahedral mesh generator TetGen [32], together with an indicator field for the desired element size. The remeshing aims to produce smaller elements in areas where the damage evolves significantly and larger elements in undamaged regions or regions where the damage growth has stopped. The damage rate is used as a pointwise indicator to set the element size. Elements with the largest damage rate have the smallest volume and vice versa. For the intermediate damage rates, the volume of the elements is interpolated between the maximum and minimum values, proportional to their damage rate.

To avoid an overly refined discretisation, the triangular surface mesh is coarsened wherever element edges become smaller than a predefined value. Mesh decimation is done using an edge collapse technique [33,34]. In each step, the shortest edge of the surface triangles (if shorter than a predefined value) is collapsed by unifying the two adjacent vertices (a and b in Fig. 2). Vertex a and the two adjacent faces vanish from the topology. Vertex b is moved to a new position c which is the midpoint between a and b . After collapsing an edge, we measure the dihedral angle between the neighboring newly produced faces and if overlapping occurs, the edge collapse is canceled and the original surface is recovered. This process is repeated until the desired coarsened surface is produced.

After remeshing, data which are available on the Gauss points of the old mesh are transferred to the Gauss points of the new mesh. For this we first use global smoothing, i.e. continuously interpolated, piecewise linear fields are determined which fit the integration point data best in a least squares sense. These data are subsequently interpolated at the new nodal coordinates and finally, using the element shape functions, the new Gauss point data are retrieved. In order to ensure a robust transfer, only a minimum set of data is transferred and the remaining data are reconstructed using the constitutive equations. This operation, which is an indispensable ingredient of the remeshing algorithm, is explained in detail in Ref. [28]. After transfer and reconstruction, balancing iterations are done to restore global equilibrium in the finite element sense. Since these iterations are not representative for any physical deformation, the material behavior is assumed to be elastic in order to guarantee convergence. Finally, because of the elastic nature of the balancing iterations, it is checked if the stress state obtained by them is on or inside the yield surface; otherwise the yield surface is corrected to restore yield consistency.

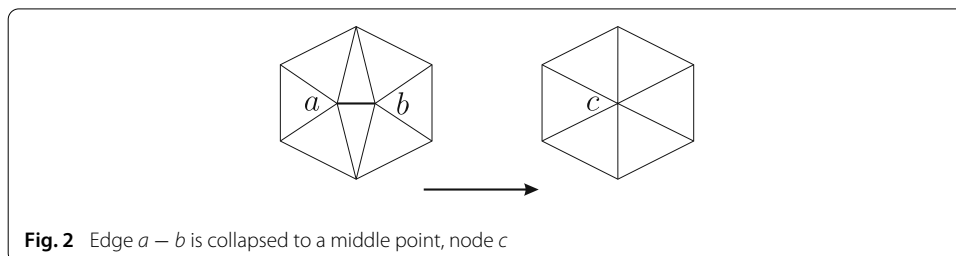


Fig. 2 Edge $a - b$ is collapsed to a middle point, node c

Crack propagation

In computational fracture mechanics, a critical stress intensity factor or J-integral value is typically used as a criterion for crack growth. In addition, a maximum hoop stress (MHS), minimum strain energy density or maximum energy release rate criterion is used to determine the crack growth direction. A different approach is employed in this study, where the evolution of the continuum damage variable, ω_p , governs the propagation of a crack. This has the advantage that crack initiation and propagation can be dealt with using the same (continuum) equations and no separate fracture criteria are necessary. Once the crack has been initiated, it follows the damage evolution ahead of it wherever the damage has become critical, i.e. $\omega_p = 1$. This concept has been successfully applied to 2D crack growth simulations in shear dominated problems like blanking [17,22].

This section summarizes the required steps for extending these algorithms to 3D problems. In 2D, the crack-front is a point, whereas in 3D it is a curve. For each node lying on this curve, a growth direction is determined in a plane perpendicular to the front. By using the nonlocal damage driving variable field in this plane, a direction vector is computed for all nodes lying on the crack-front. Using all these vectors, the extended crack surface is constructed. We discuss the numerical treatment of crack initiation in the next section, since it employs concepts developed here for crack propagation.

Crack propagation direction and distance

Contrary to the 2D case, where a crack is ending in a point called crack tip, here it is delimited by a curve, the crack-front. The crack-front is either a closed loop or it has two ends called the crack-front corners, see Fig. 3.

At each converged loading increment, the damage at a point lying on the crack-front is compared to the critical damage, $\omega_p^c = 0.99$, on the basis of which the crack is extended (or not). This value has been found to be sufficiently close to the theoretical value of $\omega_p^c = 1$

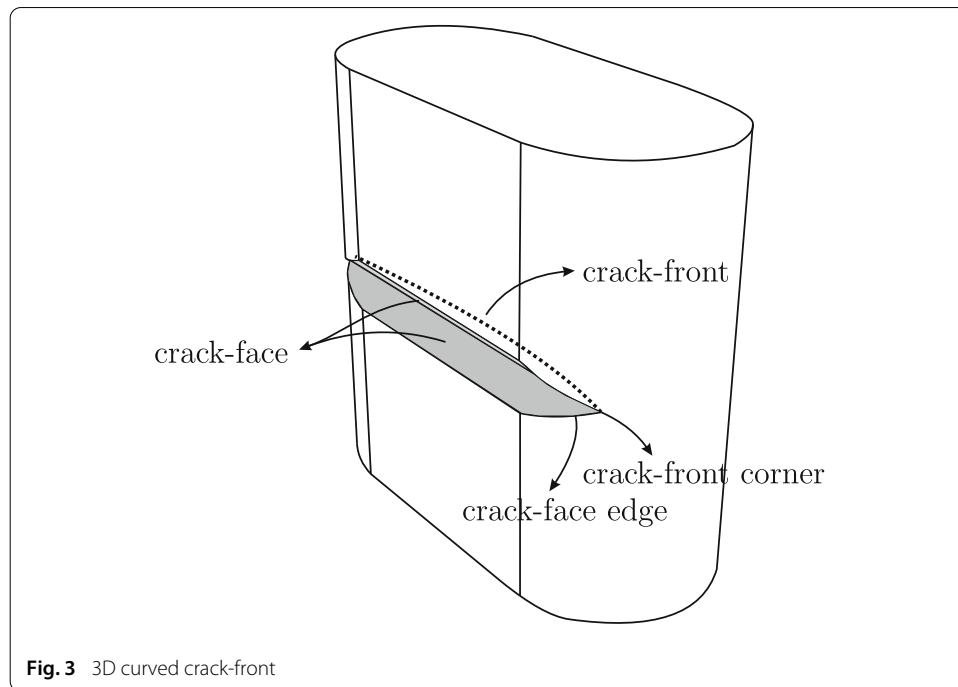


Fig. 3 3D curved crack-front

to ensure that most of the energy dissipation due to damage growth has taken place and the stress level has dropped sufficiently for it to not to be affected significantly by the insertion of a new crack segment. For a detailed study of the influence of these numerical parameters, in 2D, we refer to Ref. [22].

Using a tetrahedral discretisation of the 3D geometry, crack-front points coincide with finite element nodes. The damage variable is extrapolated from Gauss points to these nodes using a global smoothing procedure, i.e. a continuous, piecewise linear field is determined which fits the integration point data best in a least squares sense. The crack is predicted to grow over a distance which depends on the damage field ahead of the considered crack node and its direction is evaluated differently at the crack-front compared to the crack-front corners. Both cases are therefore explained separately below, followed by the distance by which the crack is extended.

Propagation of a crack-front node

For each crack-front node, a corresponding growth direction and distance must be determined. For each node, a reference plane is defined in which the direction and distance of the crack growth will be computed. The tangent to the crack-front at the desired point, o in Fig. 4, is used as the normal to this reference plane. This normal is determined from the discretised crack-front as follows.

As shown in Fig. 4a, for the crack-front point o , the vectors \vec{v}_1 and \vec{v}_2 are the vectors connecting the considered crack-front vertex to its neighboring vertices in the discretised geometry. The tangent vector is then computed as

$$\vec{n} = \frac{\vec{v}_1/\|\vec{v}_1\| - \vec{v}_2/\|\vec{v}_2\|}{\|\vec{v}_1/\|\vec{v}_1\| - \vec{v}_2/\|\vec{v}_2\|\|} \tag{18}$$

where $\|\vec{v}\|$ is the L^2 norm of a vector \vec{v} . Having obtained the (normal to the) reference plane for each node reduces the problem to a 2D crack propagation (direction and distance) problem, similar to the one dealt with by Mediavilla et al. [22]. The reference plane intersects the crack faces along two curves as shown in Fig. 4c. Motivated by the 2D procedure of Mediavilla et al. [22], the nonlocal damage driving variable \bar{z} is sampled in N points in a semi-circle located in the reference plane. A comparison has shown

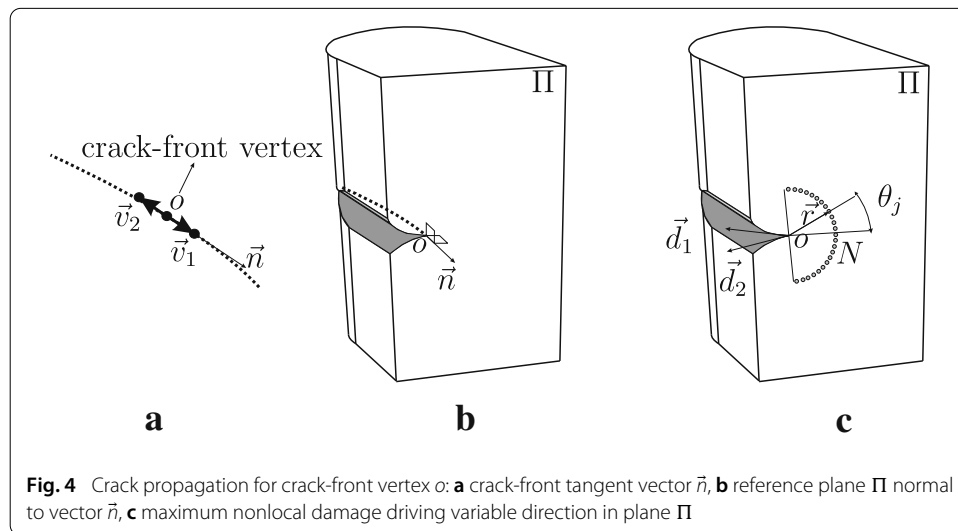


Fig. 4 Crack propagation for crack-front vertex o : **a** crack-front tangent vector \vec{n} , **b** reference plane Π normal to vector \vec{n} , **c** maximum nonlocal damage driving variable direction in plane Π

that using the nonlocal damage driving variable instead of the damage variable as used by Mediavilla et al. [22] avoids abrupt changes in the crack growth direction due to small local (numerical) variations between adjacent nodes. Vectors \vec{d}_1 and \vec{d}_2 in Fig. 4c are obtained from the intersection of the reference plane with the tetrahedral crack face edges of the discretised geometry. These two vectors are used to compute the vector \vec{d} that sets the central direction of the considered semi circle via

$$\vec{d} = -\frac{\vec{d}_1/\|\vec{d}_1\| + \vec{d}_2/\|\vec{d}_2\|}{\|\vec{d}_1/\vec{d}_1\| + \vec{d}_2/\|\vec{d}_2\|\|} \tag{19}$$

The position of a sampling point with respect to the crack-front vertex is given by the vector

$$\vec{r}_{ij} = r_i \cos(\theta_j) \vec{d} + r_i \sin(\theta_j) (\vec{n} \times \vec{d}), \quad -\frac{\pi}{2} < \theta_j < \frac{\pi}{2} \tag{20}$$

where four radii $r_1 = \frac{3}{4} \Delta a$, $r_2 = \Delta a$, $r_3 = \frac{3}{2} \Delta a$, $r_4 = 2 \Delta a$ are used. Δa is the maximum crack growth distance which is typically chosen to be a few times the smallest element edge. The nonlocal variable is sampled in N different angles ranging from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. For each r_i an angle θ_i is defined as (Fig. 4c)

$$\theta_i = \arg \max_{\theta_j} \bar{z}(r_i, \theta_j) \tag{21}$$

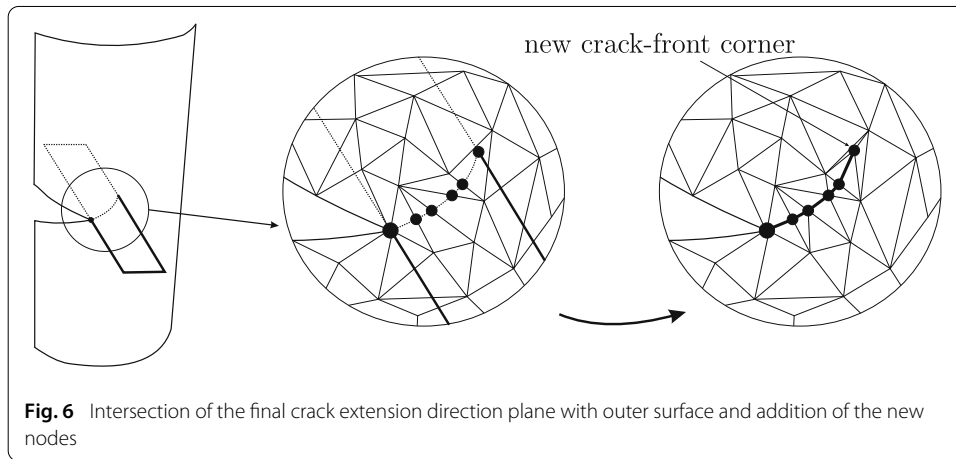
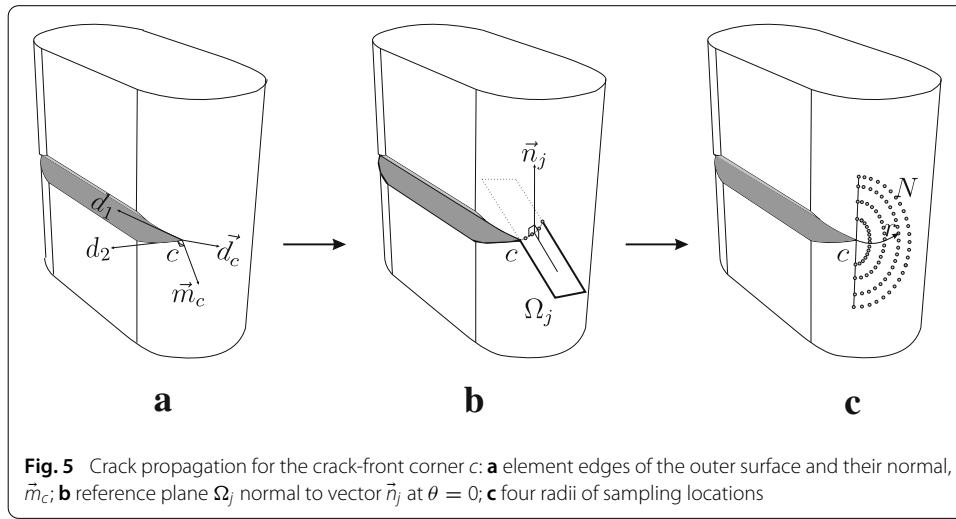
θ_i thus represents the angle at which the nonlocal damage driving variable has its maximum at a given distance r_i . Using θ_i and Eq. (20), the crack growth direction vector \vec{r}_i is obtained for each sampling distance. In order to ensure that the crack direction does not fluctuate due to local variations, the obtained crack growth direction vectors are averaged, yielding the following crack propagation direction \vec{R} for that node.

$$\begin{aligned} \vec{r}_{avg} &= \frac{\vec{r}_1}{\|\vec{r}_1\|} + \frac{\vec{r}_2}{\|\vec{r}_2\|} + \frac{\vec{r}_3}{\|\vec{r}_3\|} + \frac{\vec{r}_4}{\|\vec{r}_4\|} \\ \vec{R} &= \frac{\vec{r}_{avg}}{\|\vec{r}_{avg}\|} \end{aligned} \tag{22}$$

Propagation of a crack-front corner

Crack-front corners are the crack-front nodes located on the outer surface of the body. In order to ensure that the crack-front corner remains on the outer surface, the direction of crack growth has to be identified from the distribution of the nonlocal damage driving variable on this outer surface. Note that this surface is not necessarily planar and its geometry is available only in terms of the triangular surface mesh. The crack direction is computed in a similar fashion as for crack-front vertices, albeit on the discretised outer surface rather than the plane Π . Instead of a semi-circular set of sampling points in the plane Π , we therefore consider a set of planes intersecting the outer surface of the body to establish the potential growth directions. Each of these planes contains the crack-front corner node and has a normal \vec{n}_j , see Figs. 5 and 6. To determine \vec{n}_j , we first define the corner vector \vec{d}_c according to Eq. (19), where \vec{d}_1 and \vec{d}_2 are now the vectors along the element edge at the intersection of the outer surface and the two faces of the crack (Fig. 5a). We also define a corner vector \vec{m}_c perpendicular to vectors \vec{d}_1 and \vec{d}_2 :

$$\vec{m}_c = \frac{\vec{d}_1 \times \vec{d}_2}{\|\vec{d}_1 \times \vec{d}_2\|} \tag{23}$$



Finally vectors \vec{d}_c and \vec{m}_c are used to compute a set of plane normals as follows

$$\vec{n}_j = \cos(\theta_j)\vec{d}_c + \sin(\theta_j)(\vec{m}_c \times \vec{d}_c), \quad -\frac{\pi}{2} < \theta_j < \frac{\pi}{2} \tag{24}$$

Ω_j is the plane defined by the crack-front corner and the normal vector \vec{n}_j . Figure 5b shows this plane for $\theta = 0$. N different angles θ_j , ranging from $-\pi/2$ to $\pi/2$ are selected. A piecewise linear curve is obtained for each of these planes by intersecting it with the outer surface. Along these curves the nonlocal damage driving variable \bar{z} is sampled at four different distances r_i measured along the piecewise linear curve. For each r_i (the same sampling distances are used as in the previous section but then relative to the crack-front corner) there exists a plane with normal vector \vec{n}_i , where \bar{z} has its maximum on the intersection line of this plane with the external surface—cf. Eq. (21). Finally using Eq. (22), the average of these normals constitutes the growth direction for the crack-front corner.

Directional smoothing

Having obtained the averaged growth direction for all crack-front nodes and corners independently, these directions are again smoothed (relative to the neighboring ones) in order to damp possible numerically induced crack roughness. The direction vector of a

node k on the crack-front is combined with that of the adjacent nodes using the following smoothing operation:

$$\begin{aligned} \vec{r}_k &:= (\vec{r}_{k-1} + 2\vec{r}_k + \vec{r}_{k+1}) / 4 \\ \vec{R}_k &= \frac{\vec{r}_k}{\|\vec{r}_k\|} \end{aligned} \tag{25}$$

This filtering is only applied to the crack-front nodes and not the corners.

Growth distance

Smoothing the direction of the crack growth paves the path for obtaining a growth distance. At each node k at which the critical damage value ω_p^c is exceeded, the crack is assumed to grow in the computed direction over a distance L_k until the damage drops below $\omega_p = 0.97 \omega_p^c$. To obtain a smoother crack surface for more stable (re)meshing and computation, we furthermore set a minimum and maximum growth distance as follows: $L_{min} = 0.1 \Delta a$; $L_{max} = \Delta a$. This implies that for a point p_k^o on the old crack-front, the corresponding position on the new crack-front p_k^n is obtained as follows:

$$\vec{p}_k^o \vec{p}_k^n = L_k \vec{R}_k \tag{26}$$

Before constructing the crack surface and although the crack direction has already been smoothed in Eq. (25), the new crack-front is further smoothed by filtering all of its crack-front positions as follows

$$p_k^n := (p_{k-1}^n + 2p_k^n + p_{k+1}^n) / 4 \tag{27}$$

Note that this filtering influences mainly the obtained growth distances and has little influence on the direction, which was already smoothed in (25).

Construction of the new crack surface

The propagation direction and distance have now been computed for all nodes on the crack-front. Next step is to construct a new segment of the crack surface, along which the crack will be opened. First, the intersection of the new crack segment with the outer surface is determined. This procedure, which is schematically shown in Fig. 6, ensures that the crack surface remains properly connected to the outer surface.

In order to modify the surface, the computed crack extension direction plane for the crack-front corner is intersected with the triangular outer surface elements. Starting from the old crack-front corner, surface elements are split along the direction plane until the predicted growth distance has been reached. Triangle edges which are cut by the direction plane are split by adding a node and the triangle is divided into two triangles, see Fig. 6. If the intersection point is within a certain distance (namely a tolerance which here is 0.1 times the element edge) of an existing edge or node, the node or edge is mapped onto the crack direction plane. This avoids the creation of excessively small surface elements. If the crack extension direction exactly passes through a node or an already available edge, then no modification is made. This process is repeated until the predicted growth distance is reached. If the new crack-front corner is inside a triangle, then this triangle is divided into three triangles and the node is stored as the new crack-front corner.

The new crack-front is now obtained using the two new crack-front corner nodes on the boundary and the propagation direction and distance of the old crack-front nodes in the body's interior. Having modified the outer surface and computed the new crack-front,

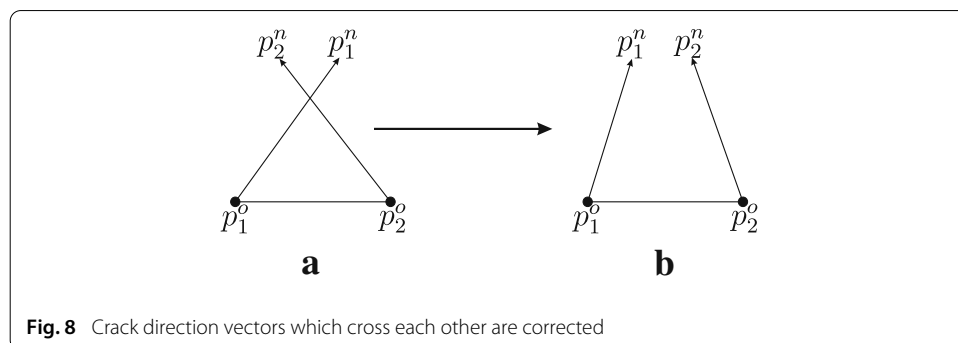
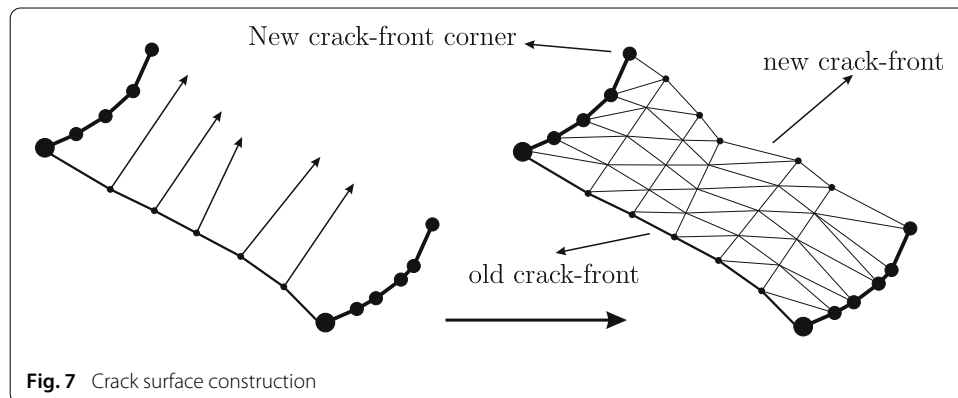
paves the way for the crack surface reconstruction. This is done by triangulation of the 3D surface which contains the old crack-front, new crack-front and the two crack-front corner traces as its boundaries, see Fig. 7.

There are some special cases where the above mentioned algorithm cannot be directly applied. One case is when two crack direction vectors are crossing each other. In this case, the points at which these directional vectors are pointing are swapped. Figure 8a shows how the directional vectors of p_1^o and p_2^o are crossing and their corresponding points p_1^n and p_2^n in Fig. 8b are swapped. Another case is when a vector ends outside of the (discretised) geometry. In this case this vector is discarded and the average crack vector of its neighboring nodes is used instead.

Meshing of the new geometry

The constructed crack surface based on the crack propagation distances and directions is now used to discretise the geometry. The geometrical description consists of the outer surface of the volume, possibly including parts of the already existing crack surface, and an inner surface which defines the new crack growth segment. The new crack surface is treated as an internal boundary by the mesher, so that tetrahedral elements are generated on both sides of the surface without intersecting it.

In order to properly model the opening of the crack surface, a topological data structure is needed. This data structure is built using the connectivity of the elements and geometry of the discretised domain. Using this data structure, the elements connected to each node and their position with respect to the crack surface are identified. Details of this data structure are given in the Appendix.



Crack opening

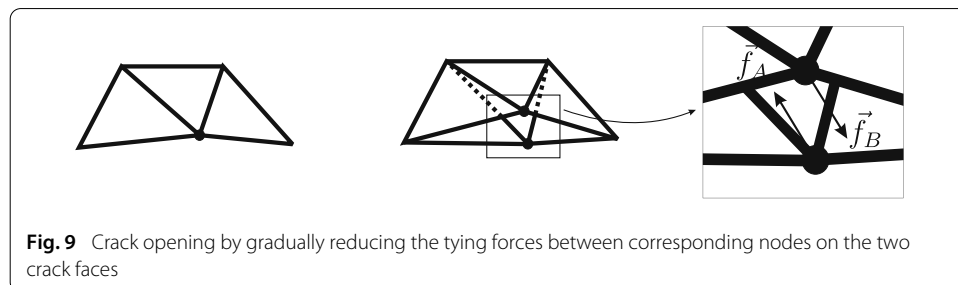
The mechanical insertion of the new crack surface is done by splitting the nodes generated on the new crack surface by the volumetric mesher. This implies that for each node, a corresponding node with the same coordinates is generated. The nodal connectivity of elements located on both sides of the crack is preserved, whereby the new node is used for the connectivity of the elements for one of the sides.

The two newly created surfaces are temporarily tied together by creating a dependency between their displacement degrees of freedom. While the crack is still closed, data from the last converged state is consistently transferred to the new mesh. Elastic equilibrium iterations are done in order to recover global consistency. During this iterative process the closed crack is treated as a new surface for Eqs. (15.3) and (16). However, the degrees of freedom for the pressure and the nonlocal damage driving variable are not tied. This improves stability of the simulation in the sense that the residual forces related to these two equations become zero in the elastic equilibrium iterations and artificial damage growth is prevented. This artificial damage growth, which is observed if all degrees of freedom are tied, may be caused by the sudden change in the boundary conditions for the nonlocal averaging Eq. (9).

Since the new crack surface is kept closed during the elastic equilibrium iterations using displacement tyings acting on the crack faces, a reaction force appears on these nodes. To mechanically open the crack, these reaction forces are first applied as external forces when the tyings are removed, and they are subsequently gradually released in a number of sub-steps, see Fig. 9 for an illustration (in which \vec{f}_A and \vec{f}_B represent these forces for one particular couple of nodes) and Ref. [22] for a more detailed description.

Crack initiation

Having established the algorithms to deal with crack propagation, we now turn our attention to the initiation of cracks based on the computed damage field. In a continuum damage mechanics approach, a crack is initiated when the scalar damage ω_p reaches a critical threshold. At this point in time, an already degenerated (softened) continuum material is split locally by introducing a discontinuity. For this purpose, the simulation is stopped, an initial crack surface is generated and, together with the outer skin of the geometry, is given as input to the 3D mesh generator. Cracks can start either inside the body (not connected to the outer surface) or from the geometry’s surface. Each of these situations is addressed separately in the following sections.



Internal crack initiation

The initiation points for cracks are the locations where the damage exceeds a predefined critical magnitude. To identify these points, all elements with damage values higher than the critical value are extracted. They constitute a 3D cloud of elements which are not necessarily interconnected. Groups of interconnected elements that are connected to an already existing crack are discarded since they contribute to crack propagation and not initiation. The element clouds connected to the outer surface form surface cracks, which are treated in the next section.

Figure 10a shows a cloud of elements with damage values higher than a critical level at the center of a body. The center point of the cloud is calculated using

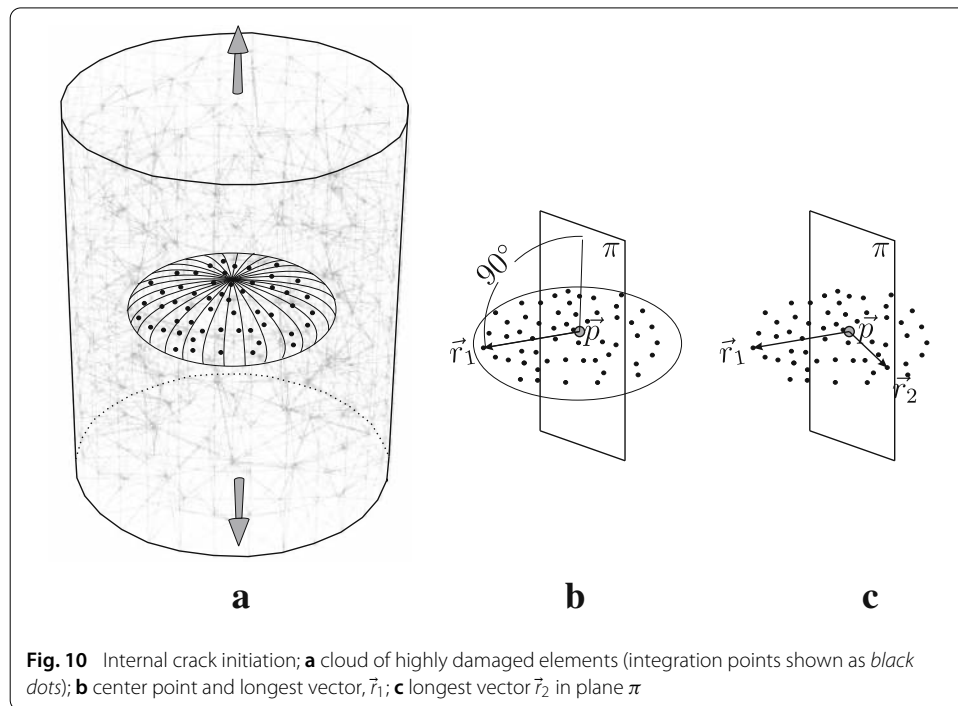
$$\vec{p} = \sum_{i=1}^n \frac{M_i}{\sum M_i} \vec{x}_i \tag{28}$$

where \vec{x}_i are the centers of elements within the cloud,

$$M_i = \frac{-V_i}{\log(\omega_p^i)} \tag{29}$$

is a damage-dependent weight factor and V_i is the volume of each element in the cloud; ω_p^i is its damage value (constant damage elements are used). The weight factor M_i ensures that larger elements with higher damage values contribute more to the calculation of the center point than small elements or elements with low levels of damage.

Starting from the center point \vec{p} , a vector \vec{r}_1 is computed, which is the longest vector connecting point \vec{p} to any other node in the cloud. A plane (π in Fig. 10b) is defined in point \vec{p} and normal to \vec{r}_1 . This plane intersects a set of elements in the cloud. All vectors from point \vec{p} to any node in this set are projected on the plane and vector \vec{r}_2 is then defined as the longest projected vector. Once \vec{r}_1 and \vec{r}_2 have been determined, they are mirrored



to obtain \vec{r}'_1 and \vec{r}'_2 . Together, these four vectors form a polygon with four sides lying in the same plane, see Fig. 11. The geometrical description of this plane, together with a point wise element size indicator (obtained from the damage rate) is given as input to the 3D tetrahedral mesher and is treated as an internal boundary for meshing.

Surface crack initiation

In some cases, a cloud of interconnected damaged elements contains nodes lying on the exterior surface of the geometry. If this is the case, a crack should nucleate from the exterior surface and propagate into the geometry with a proper propagation direction. For this purpose, the triangulated surface is modified to embed the new crack surface.

First, all damaged elements are identified that are in contact with the external surface and these are separated from the cloud. The center point of these surface elements (only a fraction of the original cloud) is obtained using Eq. (28). The closest node on the surface to this point is singled out as the surface center, \vec{P}_s in Fig. 12b. The original cloud of elements is used to determine the direction of the crack. The center point of this cloud is also computed using Eq. (28). The connection line between the surface center and the cloud center provides the vector \vec{r}_1 . To define a crack initiation plane, a second vector is needed. This vector is obtained by calculating the longest vector from the center \vec{P}_s to all surface nodes in the cloud, \vec{r}_2 in Fig. 12b. A plane normal is finally defined using the following equation:

$$\vec{N}_s = \frac{\vec{r}_1 \times \vec{r}_2}{\|\vec{r}_1 \times \vec{r}_2\|} \tag{30}$$

The intersection of this plane with the surface elements located in the cloud forms a curve on the triangularized exterior of the geometry and defines the crack-front. The crack propagation algorithm is then used to propagate the front into the body.

Crack opening

A crack surface has been defined at this stage for the cracked topology. This surface should be opened to recover equilibrium first. The applied methodology is explained here for cracks located inside the body.

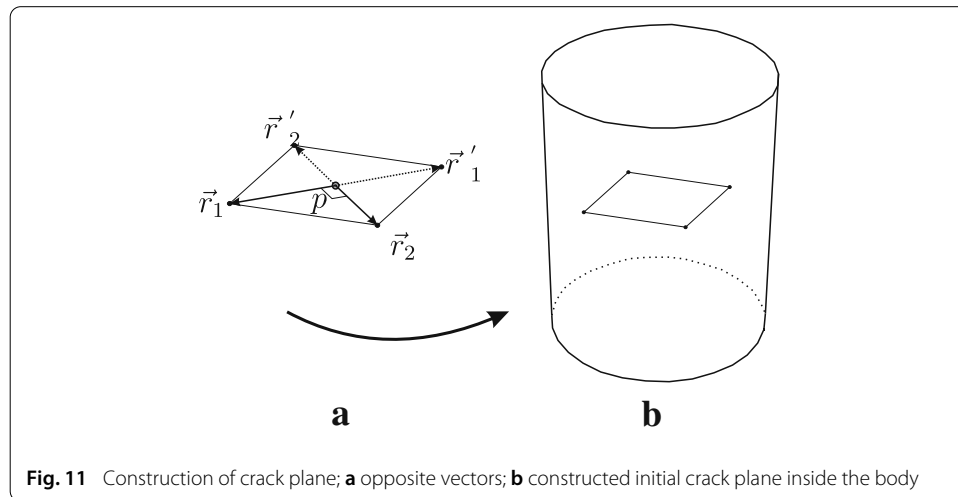
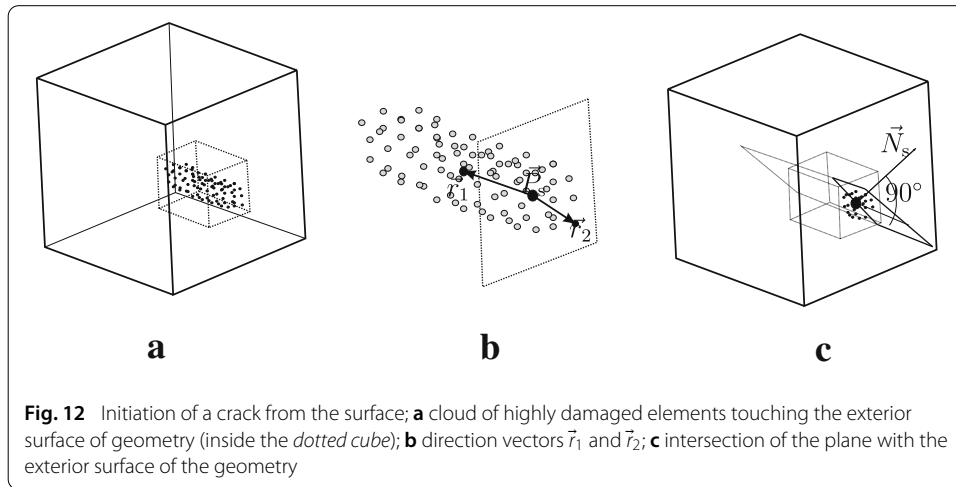


Fig. 11 Construction of crack plane; **a** opposite vectors; **b** constructed initial crack plane inside the body

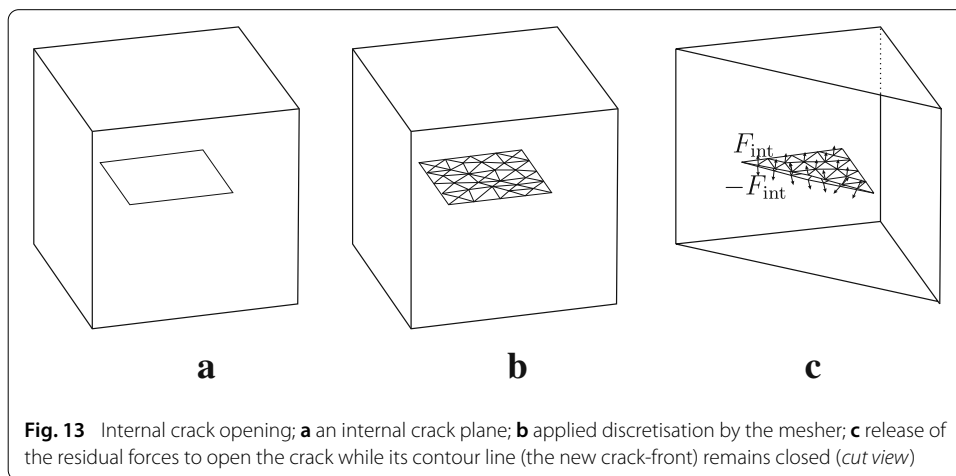


Crack opening is done in two steps. In the first step, the geometrical description of the internal crack surface is provided as input to the mesher, Fig. 13a. Next, the geometry is discretised accommodating this new interior boundary, Fig. 13b. Finally, all nodes located on the crack surface (discarding nodes on the contour of the surface) are decoupled and all internal forces acting in the nodes of the connected elements are applied as external forces, Fig. 13c. An automatic sub-incrementation procedure is then used to gradually release these forces to zero, resulting in an opened crack [22].

A similar technique is applied to open cracks in contact with the boundary of the geometry. The difference here is that the new crack front residing on the boundary is also opened.

Examples

The developed algorithm has been employed for studying crack initiation and propagation in two examples. These examples have been selected in order to assess the performance of the methods developed above in dealing with crack initiation and propagation.



A nonlinear material hardening is used throughout, in which the current hardening modulus is defined as

$$h_\epsilon = h + (\tau_{y\infty} - \tau_{y0}) \exp[-\alpha \epsilon_p] \quad \text{with } \alpha > 0 \tag{31}$$

with elasto-plastic-damage material parameters as shown in Table 1.

The described constitutive law is implemented using a locking free mixed formulation of the tetrahedral element [25,26], while a constant damage variable ω_p is used per element. In both examples, a vertical displacement is applied on the top surface of the model while the bottom surface is fixed. Frequent remeshing is used to maintain the quality of the elements and the damage rate $\dot{\omega}_p$ is employed as a point wise indicator for element size. Hence, the mesh is more refined in regions with a rapid evolution of damage.

Crack initiation in a rectangular bar

In this section, we present the results of a rectangular sample which is subjected to tension until necking and failure. The geometry and boundary conditions are shown in Fig. 14. A vertical displacement is prescribed to the top surface and the bottom surface is fully constrained. Therefore, necking is expected in the middle of the specimen.

Table 1 Material properties used [24, 29, 35]

Shear modulus G	80.19 GPa
Bulk modulus K	164.21 GPa
Initial flow stress τ_{y0}	0.45 GPa
Residual flow stress $\tau_{y\infty}$	0.715 GPa
Linear hardening coefficient h	0.129 GPa
Saturation exponent α	16.93
Damage initiation threshold κ_i	0
Critical value of history parameter κ_c	0.4
Intrinsic length ℓ	1 mm
Damage parameter A	3.9
Damage parameter B	0.63

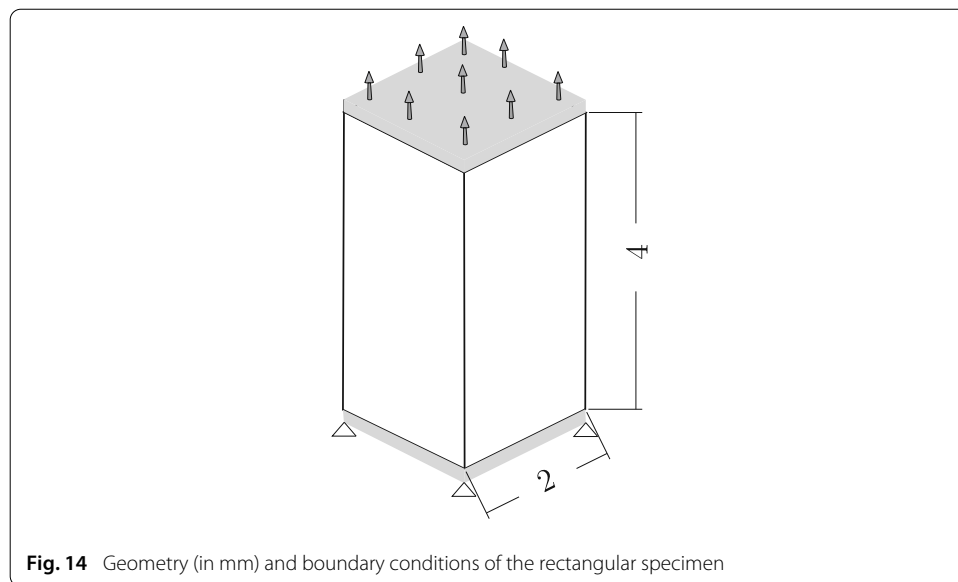


Fig. 14 Geometry (in mm) and boundary conditions of the rectangular specimen

Figure 15 shows the damage distribution as it evolves through different stages of remeshing and mesh refinement. Damage is maximal in the center of the specimen, where the hydrostatic stress is high.

As the necking progresses in the middle section of the specimen, a cloud of connected elements reveal a damage value higher than the critical level ω_p^c , as shown in Fig. 16a. The internal crack initiation algorithm is used to introduce a crack plane internally, see

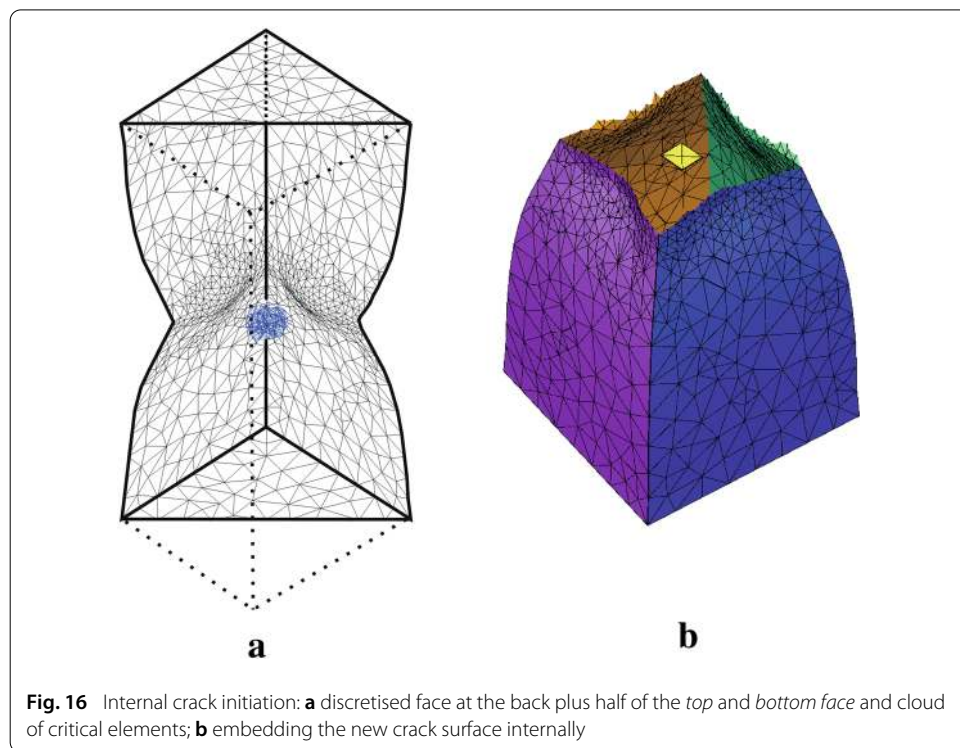
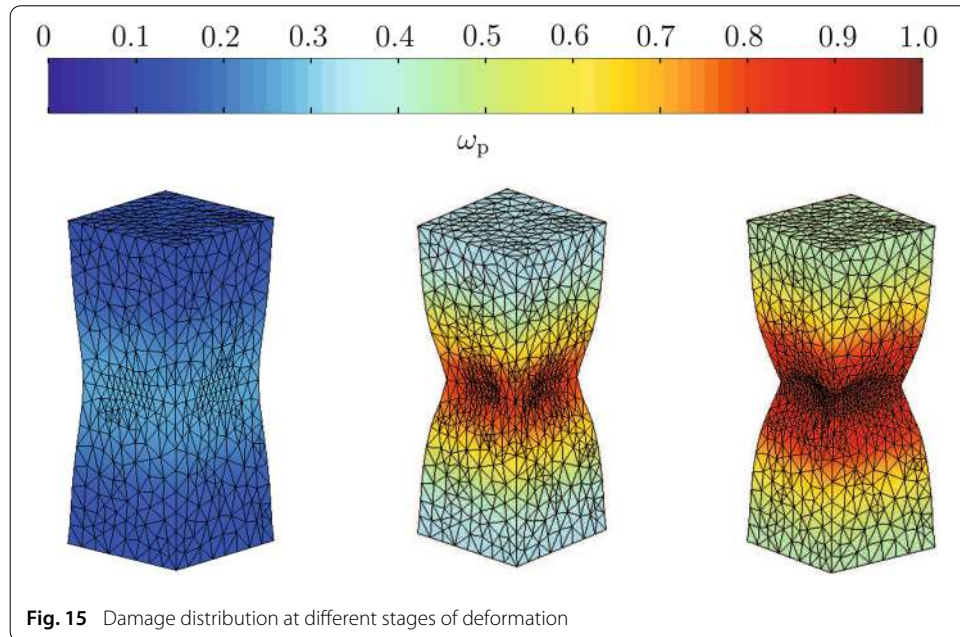


Fig. 16b. The geometry is therefore remeshed and by releasing the crack surface forces, a first crack appears inside the geometry.

Due to the concentration of the damage growth in the neck, a rapid crack growth is observed. Figure 17 shows an instant of crack propagation towards the outer surface. Since the mesh refinement is a function of the damage rate, a refined mesh appears around the crack, see Fig. 17b.

The force versus displacement response obtained for this problem is shown in Fig. 18. The diagram shows that the simulation may be continued until full failure, i.e. until the reaction force vanishes. The minor jumps on particularly the descending part of the curve are due to the transfer of state variables following remeshing steps. This transfer results in a slight inconsistency of the deformation and stress state on the new discretisation compared with that on the old. The effect is more pronounced where (and when) these fields fluctuate strongly—which explains why they are more prominent at later stages of the failure process, when the deformation is highly localised.

Surface crack initiation and propagation in a double notch specimen

The example of a double notched specimen is used to investigate the crack initiation on the surface and have a more substantial amount of crack growth. The geometry and boundary conditions are shown in Fig. 19. The deformation is imposed using displacement control and the front and back face of the geometry are free. The amount of damage growth during an increment is used as a point-wise indicator for mesh refinement. The geometry undergoes large deformations before crack initiation and propagation starts. As a result of this large deformation, elements may become distorted and compromise the required accuracy. To avoid this problem, frequent remeshing is used, after a predefined applied displacement. The number of sampling locations N and the crack increment length Δa are 50 and 0.3 mm respectively.

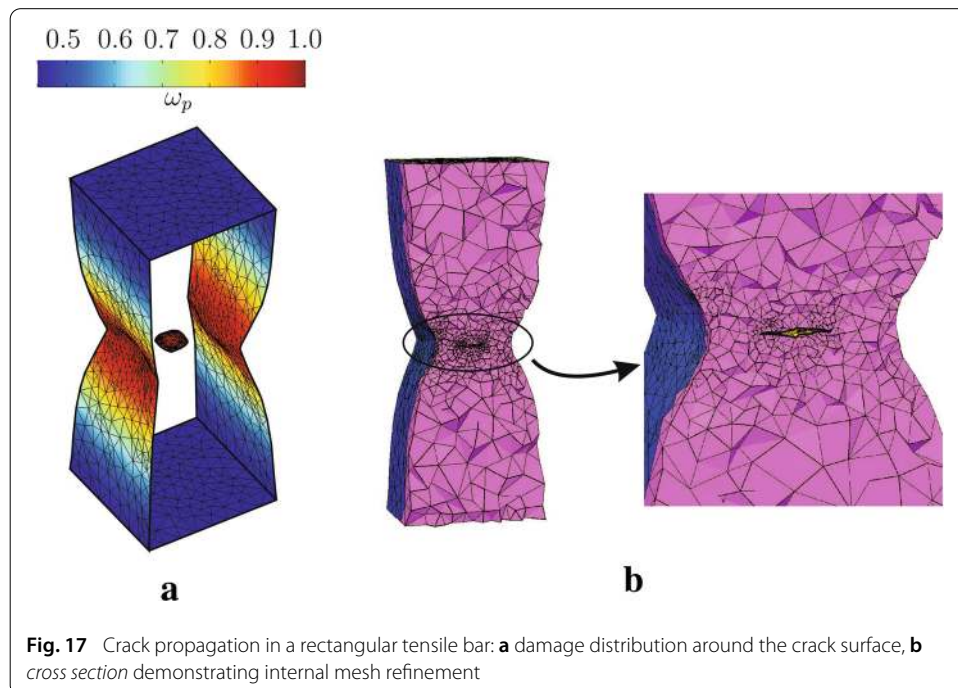


Fig. 17 Crack propagation in a rectangular tensile bar: **a** damage distribution around the crack surface, **b** cross section demonstrating internal mesh refinement

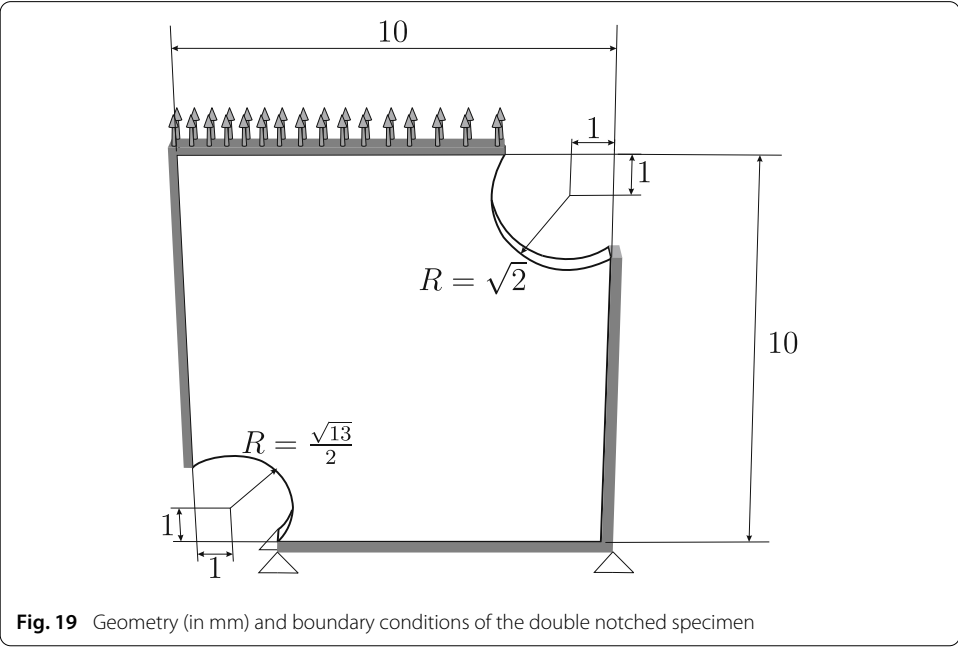
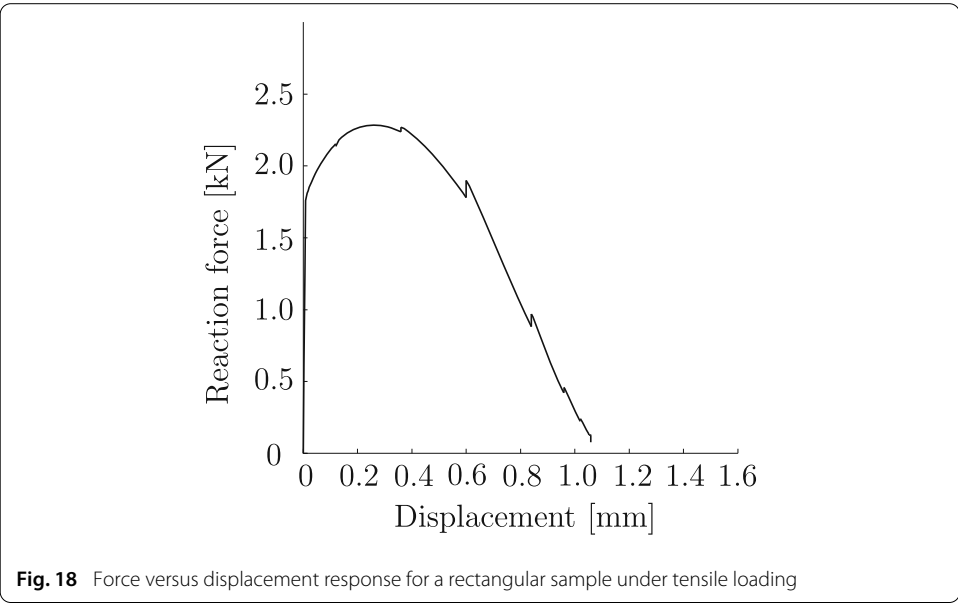
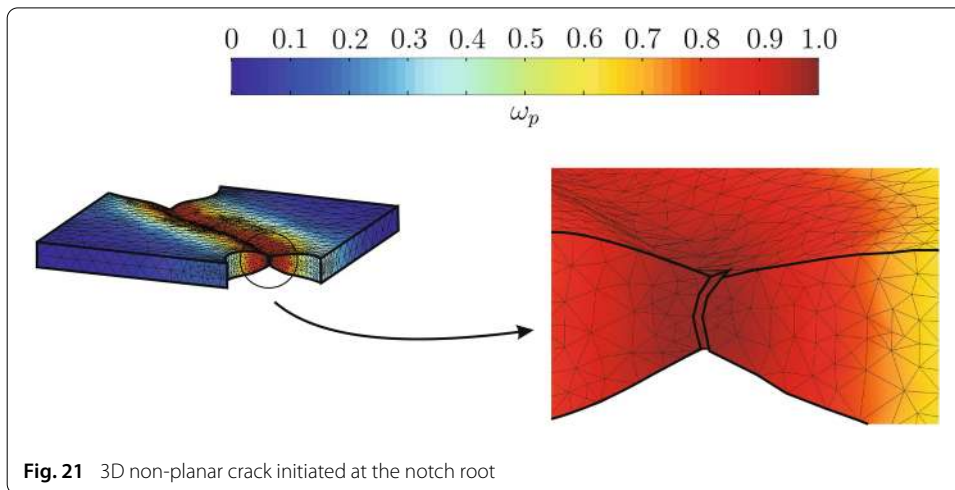
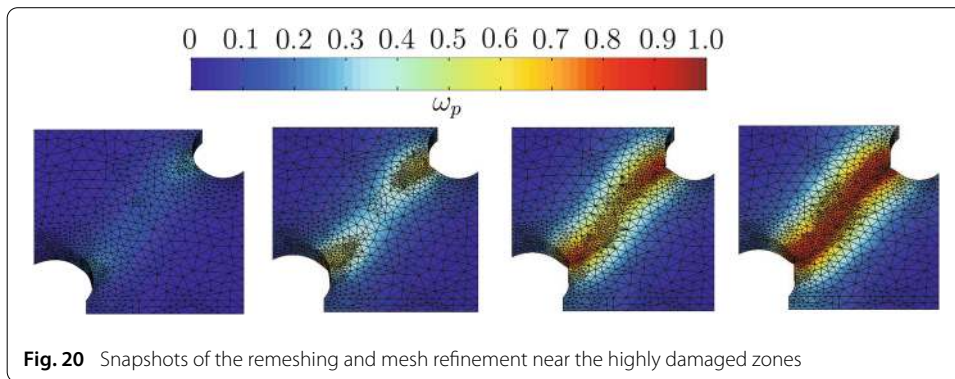


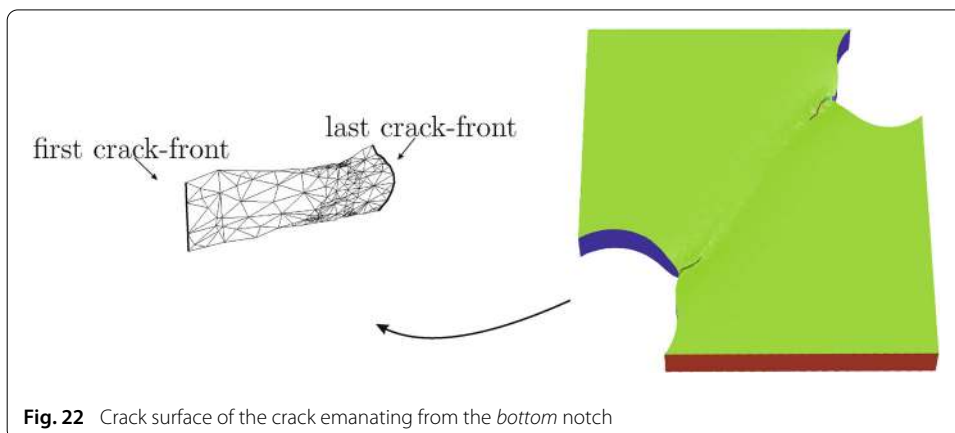
Figure 20 shows the damage distribution as it evolves through the different remeshing steps before reaching a critical value for inserting the first crack segment. As can be seen from this figure, the damage grows considerably faster at the notches, especially at the top right hand corner.

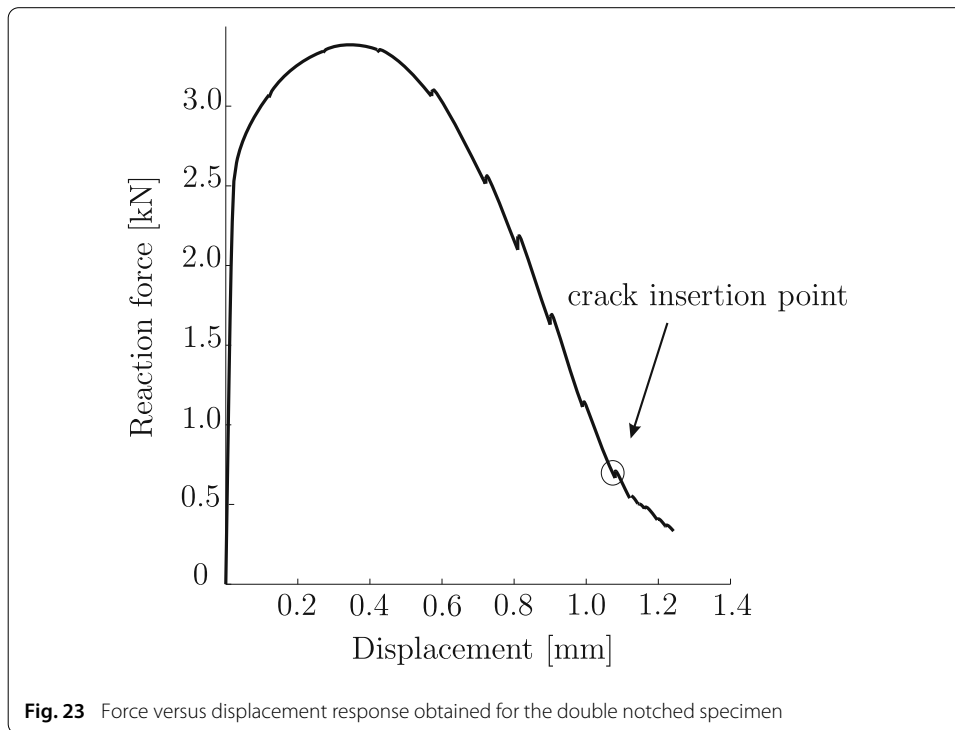
Figure 21 shows how the specimen is necking across its thickness along a curve connecting the two notches. Damage has the highest value where the mid-plane of the specimen intersects the notch root, since the hydrostatic stress and consequently the stress triaxiality is higher at this point relative to the front and back face of the specimen. Also shown in Fig. 21 is the first crack segment when it is opened and all residual forces have been released.



As the applied displacement increases, the crack which was initiated at the top right corner grows towards the bottom. After a while the damage at the bottom left corner also reaches the critical value and a second crack starts growing from there. While the second crack propagates, the crack propagation at the top is arrested. Since the crack tends to grow faster in the mid-plane of the specimen thickness, the crack-front is curved instead of a straight line, as shown in Fig. 22 for the crack growing from the bottom-left notch.

Figure 23 shows the force versus displacement response obtained from this simulation. As before, the jumps observed in this curve are due to the remeshing and transfer





between two different discretisations. Note that the first crack insertion occurs only when the mechanical strength of the specimen has already been significantly degraded by the damage evolution.

Conclusion

A large deformation 3D methodology has been developed to simulate the initiation and propagation of a crack in a ductile material, based on an underlying ductile damage mechanics formulation and a remeshing strategy.

An approach is presented to initiate a crack in 3D bodies undergoing large plastic deformations. Cracks start either internally or at the surface of the geometry, whereby a procedure is proposed for each case. In contrast to a traditional fracture mechanics approach, the size and direction of crack initiation and propagation are solely governed by the underlying damage model, and no extra criterion is therefore required.

Once a crack has been nucleated, it may propagate according to the damage field ahead of the crack tip. For each of the nodes on the current crack-front, a propagation direction and distance is computed. Depending upon the location of the node on the crack-front (corner or mid nodes), a slightly different method is used to identify the propagation vector. These propagation vectors, together with the old crack-front, are assembled to construct a new crack surface segment. The geometry is then discretised and refined in critical locations based on the damage rate.

The performance of the proposed method is shown by two examples where both cases for initiation/propagation of a crack (at the surface or internally) are demonstrated. Our results show that the method is promising in studying phenomena like internal fracture and other relevant applications. The characteristics of the proposed algorithm renders it promising for modelling 3D cracks in applications where remeshing is unavoidable.

It presents two essential advantages over a conventional fracture mechanics approach: first, it uses only a single criterion (damage model) for both crack initiation and propagation (distance and direction) and, second, the mechanical strength of the structure has been already degraded by the damage, making it more convenient to introduce a crack.

Authors' contributions

HRJ carried out most of the study, including development of the methodology and its implementation, and drafted the manuscript. RHJP and MGDG conceived the study, participated in its design and coordination and critically reviewed the manuscript. All authors read and approved the final manuscript.

Author details

¹Materials innovation institute M2i, PO Box 5008, 2600 GA Delft, The Netherlands, ²Department of Mechanical Engineering, Eindhoven University of Technology, PO Box 513, 5600 MB Eindhoven, The Netherlands.

Acknowledgements

This research was carried out under project number MC2.05205c in the framework of the Research Program of the Materials innovation institute M2i (<http://www.m2i.nl>).

Competing interests

The authors declare that they have no competing interests.

Appendix: Data structure of the discretised geometry

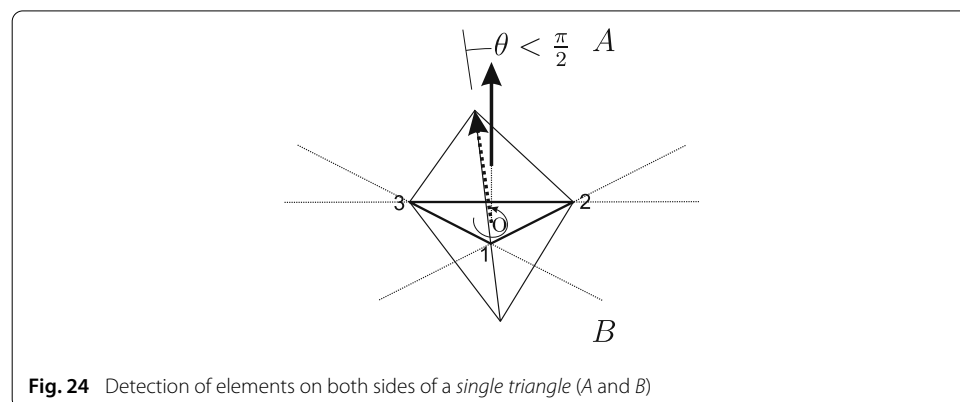
A data structure is needed in order to identify the elements spanning the crack surface. The 3D geometry is discretised using tetrahedral elements and it contains an internal triangulated surface to which the tetrahedral volume mesh conforms. In order to open the crack surface, a search algorithm is used to locate the elements on both sides of the crack surface.

Crack face elements

The first task is to find the elements which are on both sides of a triangle of the crack surface. Figure 24 shows a triangle with node numbering {1, 2, 3} and two tetrahedra connected to it. An element located on side *A* is detected by analyzing the angle between the direction of the triangle normal (obtained from the counterclockwise triangle connectivity) and the vector to the remaining node of the tetrahedron. As shown in the figure, if that angle is less than 90 degrees, then this tetrahedron is located on side *A* of the triangle.

Crack nodes

In order to open a crack at a node, it is necessary to identify all elements (not only crack face elements) at each side of the crack surface. This operation is schematically shown in



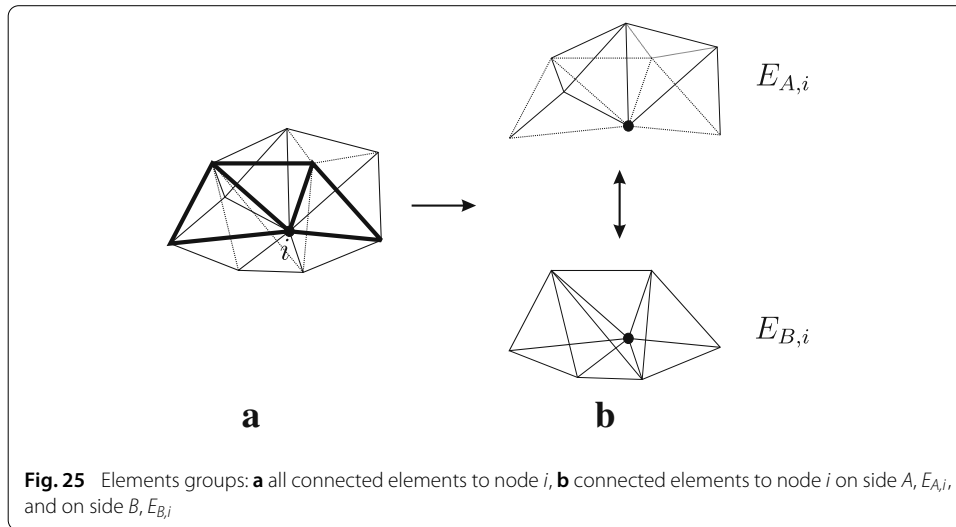


Fig. 25. E_i is the list of tetrahedral elements connected to node i . It must be subdivided into sets $E_{A,i}$ and $E_{B,i}$ associated with the respective sides of the crack surface, for which we have $E_i = E_{A,i} \cup E_{B,i}$ and $E_{A,i} \cap E_{B,i} = \emptyset$. Furthermore, the nodes connected to the elements in the subsets $E_{A,i}$ and $E_{B,i}$ are denoted as $N_{A,i}^E$ and $N_{B,i}^E$ respectively.

First step in identifying $E_{A,i}$ and $E_{B,i}$ is to identify the triangles connected to node i that are part of the crack surface (three bold triangles in Fig. 25a), called star triangles. The list of node numbers related to the star triangles, including node i (five nodes here), is called $N_{S,i}$. Each triangle is connected to two tetrahedra, one on each side, which are identified by the algorithm explained in “Conclusion” section. The related element numbers are removed from set E_i and constitute the first entries of $E_{A,i}$ and $E_{B,i}$. In order to assign all tetrahedra in E_i to $E_{A,i}$ and $E_{B,i}$ we define a node list $N_{A,i}$, which contains all nodes of elements connected to the crack face on side A , but are not on the crack face itself. Mathematically, this implies $N_{A,i} = \{N_{A,i} \subseteq N_{A,i}^E \text{ and } N_{A,i} \cap N_{S,i} = \emptyset\}$ and similarly $N_{B,i} = \{N_{B,i} \subseteq N_{B,i}^E \text{ and } N_{B,i} \cap N_{S,i} = \emptyset\}$. The remaining element numbers in $E_{A,i}$ and $E_{B,i}$ are recovered by iteratively checking if any node of the elements belongs to list $N_{A,i}$ or $N_{B,i}$. This element is then assigned to the proper set. This is repeated until all components of E_i have been visited. This technique only relies on the element connectivity, whereby no geometrical features are involved. Therefore, the crack surface complexity does not compromise the identification of $E_{A,i}$ and $E_{B,i}$.

Received: 9 February 2016 Accepted: 9 May 2016
 Published online: 07 June 2016

References

1. Gullerud AS, Dodds RH Jr, Hampton RW, Dawicke DS. Three-dimensional modeling of ductile crack growth in thin sheet metals: computational aspects and validation. Eng Fract Mech. 1999;63(4):347–74.
2. Simo JC, Oliver J, Armero F. An analysis of strong discontinuities induced by strain-softening in rate-independent inelastic solids. Comput Mech. 1993;12:277–96.
3. Armero F, Garikipati K. An analysis of strong discontinuities in multiplicative finite strain plasticity and their relation with the numerical simulation of strain localization in solids. Int J Solid Struct. 1996;33(20–22):2863–85.
4. Oliver J. Modelling strong discontinuities in solid mechanics via strain softening constitutive equations. Part 2: numerical simulation. Int J Numer Methods Eng. 1996;39:3601–23.
5. Melenk JM, Babuska I. The partition of unity finite element method: basic theory and applications. Comput Methods Appl Mech Eng. 1996;139(1–4):289–314.
6. Garikipati K, Hughes TJR. A study of strain localization in a multiple scale framework-the one-dimensional problem. Comput Methods Appl Mech Eng. 1998;159(3–4):193–222.

7. Belytschko T, Black T. Elastic crack growth in finite elements with minimal remeshing. *Int J Numer Methods Eng.* 1999;45(5):601–20.
8. Moës N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *Int J Numer Methods Eng.* 1999;46:131–50.
9. Wells GN, Sluys LJ. A new method for modelling cohesive cracks using finite elements. *Int J Numer Methods Eng.* 2001;50:2667–82.
10. Moës N, Belytschko T. Extended finite element method for cohesive crack growth. *Eng Fract Mech.* 2002;69(7):813–33.
11. Gravouil A, Moës N, Belytschko T. Non-planar 3D crack growth by extended finite elements and level sets-part I: mechanical model. *Int J Numer Methods Eng.* 2002;53:2549–68.
12. Gravouil A, Moës N, Belytschko T. Non-planar 3D crack growth by extended finite elements and level sets-part II: level set update. *Int J Numer Methods Eng.* 2002;53:2569–86.
13. Colombo D, Massin P. Fast and robust level set update for 3D non-planar X-FEM crack propagation modelling. *Comput Methods Appl Mech Eng.* 2011;200:2160–80.
14. Seabra MRR, Šuštarčič P, Cesar de Sa JMA, Rodič T. Damage driven crack initiation and propagation in ductile metals using XFEM. *Comput Mech.* 2013;52(1):161–79.
15. Dodds RH Jr, Tang M, Anderson TL. Numerical procedures to model ductile crack extension. *Eng Fract Mech.* 1993;46(2):253–64.
16. Bittencourt TN, Wawrzynek PA, Ingraffea AR, Sousa JL. Quasi-automatic simulation of crack propagation for 2D LEFM problems. *Eng Fract Mech.* 1996;55(2):321–34.
17. Brokken D, Brekelmans WAM, Baaijens FPT. Numerical modelling of the metal blanking process. *J Mater Process Technol.* 1998;83(1–3):192–9.
18. Brokken D, Brekelmans WAM, Baaijens FPT. Predicting the shape of blanked products: a finite element approach. *J Mater Process Technol.* 2000;103(1):51–6.
19. Bouchard PO, Bay F, Chastel Y, Tovina I. Crack propagation modelling using an advanced remeshing technique. *Comput Methods Appl Mech Eng.* 2000;189(3):723–42.
20. Carter B, Wawrzynek P, Ingraffea A. Automated 3D crack growth simulation. *Int J Numer Methods Eng.* 2000;47:229–53.
21. Cavalcante Neto JB, Wawrzynek PA, Carvalho MTM, Martho LF, Ingraffea AR. An algorithm for three-dimensional mesh generation for arbitrary regions with cracks. *Eng Comput.* 2001;17:75–91.
22. Mediavilla J, Peerlings RHJ, Geers MGD. An integrated continuous-discontinuous approach towards damage engineering in sheet metal forming processes. *Eng Fract Mech.* 2006;73(7):895–916.
23. Feld-Payet S. Amorçage et propagation de fissures dans les milieux ductiles non locaux. PhD thesis, École Nationale Supérieure des Mines de Paris; 2010.
24. Mediavilla J, Peerlings RHJ, Geers MGD. A nonlocal triaxiality-dependent ductile damage model for finite strain plasticity. *Comput Methods Appl Mech Eng.* 2006;195(33–36):4617–34.
25. Javani HR, Peerlings RHJ, Geers MGD. Three dimensional modelling of non-local ductile damage: element technology. *Int J Mater Form.* 2009;2:923–6.
26. Javani HR. A computational damage approach towards three-dimensional ductile fracture. PhD thesis, Eindhoven University of Technology; 2011.
27. Javani HR, Peerlings RHJ, Geers MGD. A remeshing strategy for three dimensional elasto-plasticity coupled with damage applicable to forming processes. *Int J Mater Form.* 2010;3:915–8.
28. Javani HR, Peerlings RH, Geers MG. Consistent remeshing and transfer for a three dimensional enriched mixed formulation of plasticity and non-local damage. *Comput Mech.* 2014;53(4):625–39.
29. Simo JC. Algorithms for static and dynamic multiplicative plasticity that preserve the classical return mapping schemes of the infinitesimal theory. *Comput Methods Appl Mech Eng.* 1992;99(1):61–112.
30. Goijaerts AM, Govaert LE, Baaijens FPT. Evaluation of ductile fracture models for different metals in blanking. *J Mater Process Technol.* 2001;110(3):312–23.
31. Peerlings RHJ, de Borst R, Brekelmans WAM, Geers MGD. Localisation issues in local and nonlocal continuum approaches to fracture. *Eur J Mech A Solids.* 2002;21(2):175–89.
32. Si H. Tetgen A. A quality tetrahedral mesh generator and three-dimensional delaunay triangulator. 2007. <http://www.tetgen.berlios.de/>.
33. Heckbert PS, Garland M. Optimal triangulation and quadric-based surface simplification. *Comput Geom.* 1999;14(1–3):49–65.
34. Hoppe H. Progressive meshes. In: SIGGRAPH. 1996;96:99–108.
35. Geers MGD. Finite strain logarithmic hyperelasto-plasticity with softening: a strongly non-local implicit gradient framework. *Comput Methods Appl Mech Eng.* 2004;193(30–32):3377–401.