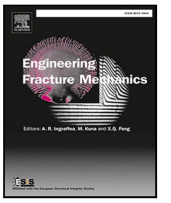
Engineering Fracture Mechanics 310 (2024) 110463

Contents lists available at ScienceDirect

Engineering Fracture Mechanics

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



# Efficient BFGS quasi-Newton method for large deformation phase-field modeling of fracture in hyperelastic materials

Aimane Najmeddine [a,*], Maryam Shakiba [b]

[a] Department Civil and Environmental Engineering, Princeton University, USA
[b] Smead Department of Aerospace Engineering Sciences, the University of Colorado Boulder, USA

ARTICLE INFO

*Keywords:*
Phase-field
Quasi-Newton
BFGS
Large deformation
Hyperelasticity
Abaqus/Standard
User-element subroutine

ABSTRACT

The prediction of crack propagation in materials is a crucial problem in solid mechanics, with many practical applications ranging from structural integrity assessment to the design of advanced materials. The phase-field method has emerged as a powerful tool for modeling crack propagation in materials, due to its ability to accurately capture the propagation of cracks. However, current phase-field algorithms suffer from the elevated computational cost associated with the so-called staggered solution scheme, which requires extremely small time increments to advance the crack due to its inherent conditional stability. In this paper, we present, for the first time, a quantitative analysis detailing the numerical implementation and comparison of two common solution strategies for the coupled large-deformation solid-mechanics-phase-field problem, namely the quasi-Newton based Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) and the full Newton based alternating minimization (or staggered) (AM/staggered) algorithm. We demonstrate that the BFGS algorithm is a more efficient and advantageous alternative to the traditional AM/staggered approach for solving coupled large-deformation solid-mechanics-phase-field problems. Our results highlight the potential of the quasi-Newton BFGS algorithm to significantly reduce the computational cost of predicting crack propagation in hyperelastic materials while maintaining the accuracy and robustness of the phase-field method.

## 1. Introduction

The phase-field approach was established based on the diffuse representation of localized discontinuity and was initially employed to simulate brittle fracture [1–3]. Its capacity to capture complex crack geometries – including arbitrary nucleation, propagation, and branching within a unified framework – has broadened its application across various fracture problems. This method's versatility has led to its expansion into a diverse range of scenarios, including multi-physics simulations [4–6], fatigue analysis [7–10], and the characterization of fracture behavior in elastomeric materials involving large deformations [11–18].

The phase-field method is based on a variational approach to fracture wherein crack initiation and propagation are the direct results of the minimization of the energy potential functional. The functional is defined as the sum of a strain energy density function integrated over the volume of a body and a crack surface energy component integrated over the crack region of the body. While the phase-field approach to fracture has proven vastly effective in capturing complex fracture patterns in brittle and quasi-brittle materials, a significant drawback to its efficiency is the computational cost associated with solving the required minimization problem. This issue is due to two factors. On the one hand, extremely fine meshes are often necessary to resolve the damage gradient due to the non-uniformly distributed phase-field. On the other hand, the non-convexity of the energy potential functional with

* Corresponding author.
*E-mail address:* an3801@princeton.edu (A. Najmeddine).

https://doi.org/10.1016/j.engfracmech.2024.110463
Received 11 April 2024; Received in revised form 25 August 2024; Accepted 3 September 2024
Available online 12 September 2024

**Nomenclature**

| | |
|---|---|
| $\Omega_0$ | Initial configuration of the body |
| $\Gamma_0$ | Boundary of the initial configuration |
| $\mathbf{m}$ | Unit outward normal vector |
| $\mathbf{X}$ | Position vector in the initial configuration |
| $\mathbf{x}$ | Position vector in the current configuration |
| $\chi$ | Motion mapping |
| $\mathbf{u}$ | Displacement field |
| $\mathbf{F}$ | Deformation gradient tensor |
| $\mathcal{S}_0$ | Sharp crack in the initial configuration |
| $d$ | Phase-field variable |
| $\mathcal{B}$ | Localization band |
| $\mathbf{b}_0$ | Body force per unit volume |
| $\mathbf{t}_0$ | Prescribed traction |
| $\Gamma_0^N$ | Natural boundary |
| $\Pi$ | Potential energy functional |
| $\Psi$ | Degraded strain energy density function |
| $\omega(d)$ | Phase-field degradation function |
| $G_c$ | Fracture energy |
| $\gamma$ | Crack surface density function |
| $l_c$ | Length scale parameter |
| $\alpha(d)$ | Crack geometric function |
| $\Gamma_0^D$ | Essential boundary |
| $\mathbf{P}$ | First Piola-Kirchhoff stress tensor |
| $\mathcal{H}$ | Local crack driving force |
| $\psi_{mech}$ | Intact strain energy density function |
| $\mu$ | Shear modulus |
| $K_B$ | Bulk modulus |
| $J$ | Determinant of the deformation gradient |
| $\lambda_i$ | Principal stretches |
| $\psi_{mech}^+$ | Tensile component of strain energy density |
| $\psi_{mech}^-$ | Compressive component of strain energy density |
| $\mathbf{w}_u$ | Arbitrary vector field for displacement |
| $w_d$ | Arbitrary scalar field for phase-field |
| $N_A$ | Shape function for node A |
| $\mathbf{R}_u^A$ | Element-level residual for displacement at node A |
| $R_d^A$ | Element-level residual for phase-field at node A |
| $\mathbf{K}_{uu}^{AB}$ | Element-level tangent for displacement-displacement coupling |
| $\mathbf{K}_{ud}^{AB}$ | Element-level tangent for displacement-phase-field coupling |
| $\mathbf{K}_{du}^{AB}$ | Element-level tangent for phase-field-displacement coupling |
| $\mathbf{K}_{dd}^{AB}$ | Element-level tangent for phase-field-phase-field coupling |
| $\bar{\mathbf{u}}$ | Vector of nodal displacements |
| $\bar{\mathbf{d}}$ | Vector of nodal phase-field values |
| $\mathbf{z}^e$ | Vector of element unknown nodal values |
| $\bar{\mathbf{F}}$ | F-bar deformation gradient |
| $\bar{\mathbf{F}}_{pe}$ | Deviatoric part of the F-bar deformation gradient |
| $\mathbf{F}_0$ | Deformation gradient at the element centroid |
| $\hat{\mathbf{P}}$ | Stress tensor evaluated using $\bar{\mathbf{F}}$ |
| $\bar{\mathsf{S}}$ | Voigt form of the F-bar first Piola-Kirchhoff stress |
| $\mathbb{A}$ | Fourth-order tangent modulus tensor |
| $\mathbb{Q}_R$ | Fourth-order tensor for F-bar formulation |
| $\mathbb{Q}_{R0}$ | Fourth-order tensor for F-bar formulation at element centroid |

respect to the unknowns (i.e., displacement and phase-field variables) makes any attempt at solving the fully coupled field problem (i.e., simultaneously solving for the two variables) using a standard Newton-based monolithic solver a rather cumbersome task, resulting in poor performance and divergence of solution.

Attempts have been made to bypass monolithic solvers and to propose alternative solution methods that provide more robustness. The alternating minimization (AM) or staggered solver [19,20] has been one such approach that is robust enough to solve the coupled solid mechanics phase-field governing equations sequentially (i.e., in a staggered manner). However, the AM solver is slow and oftentimes requires extremely fine time increments to achieve convergence at critical incremental steps due to its unconditional stability. Quasi-Newton-based methods such as the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm have been proposed as alternative algorithms aimed at achieving both a robust and an efficient solution scheme for the coupled solid mechanics phase-field problem [21–25]. However, all existing efforts have focused on linear elastic problems, and the application of these methods in highly nonlinear configurations has remained unexplored.

Several researchers have sought to incorporate the phase-field approach within a large-deformation setting to simulate the fracture behavior of elastomers, hydrogels, and soft bio-materials using hyperelastic constitutive equations [18,26–33]. Some of these works have used the AM/staggered approach, which, as mentioned earlier, is slow and usually requires extremely small time increments to achieve accurate response [6,34–38], while others proposed higher-order schemes or adaptive mesh strategies to help improve computational efficiency [33,39–44], but the implementation of such strategies within finite element (FE) software such as ABAQUS [45] is usually highly complex. However, the use of quasi-Newton-based methods to solve the nonlinear system of coupled equations has not been investigated in the context of large deformation kinematics.

This work presents several novel contributions to phase-field damage of localized failure in hyperelastic materials undergoing large deformations. First, we extend the application of the BFGS algorithm beyond linear elastic problems to solve the coupled large-deformation solid mechanics and phase-field problem for hyperelastic materials. This represents a significant advancement in computational efficiency for modeling complex fracture patterns in nonlinear elastic materials. Second, we provide a detailed quantitative comparison of the computational efficiency between the BFGS algorithm and the traditional alternating minimization (AM)/staggered approach for hyperelastic materials, demonstrating substantial time savings. Third, we present a novel implementation of the BFGS algorithm within a user-defined element (UEL) subroutine in Abaqus/Standard, specifically tailored for hyperelastic phase-field problems. Fourth, we demonstrate the effectiveness of the BFGS algorithm for capturing complex fracture patterns in hyperelastic materials through multiple benchmark examples, including single-edge notch specimens under uniaxial tension and pure shear, as well as a notched beam under three-point bending. Lastly, we incorporate tension–compression decomposition of the hyperelastic strain energy density function within the BFGS framework, further extending its applicability to a wider range of loading scenarios. These contributions collectively represent a significant step forward in the efficient and accurate modeling of fracture in hyperelastic materials undergoing large deformations.

## 2. Governing differential equations for the coupled large deformation solid mechanics and phase-field problem

### 2.1. Notation

Tensorial notation is used in this work. Bold letters are used to signify vectors or tensors. Inner vector products are represented by the dot " $\cdot$", while inner tensor products are denoted by the colon " $:$" operators. For instance, given vectors $\mathbf{a}$ and $\mathbf{b}$ and tensors $\mathbf{A}$ and $\mathbf{B}$, we express $\mathbf{a}\cdot\mathbf{b}=\mathbf{a}^{\mathsf{T}}\mathbf{b}$ and $\mathbf{A}:\mathbf{B}=\mathrm{tr}(\mathbf{A}^{\mathsf{T}}\mathbf{B})=\mathrm{tr}(\mathbf{A}\mathbf{B}^{\mathsf{T}})$, where the superscript $^{\mathsf{T}}$ denotes the transpose of a vector or tensor.

Furthermore, we use Div and div to represent the material and spatial divergence operators, respectively. For gradient operations, $\nabla_{\mathbf{X}}(.)=\nabla(.)=\frac{\partial(.)}{\partial\mathbf{X}}$ signifies the material gradient operator, while $\nabla_{\mathbf{x}}(.)=\frac{\partial(.)}{\partial\mathbf{x}}$ represents the spatial gradient operator.

### 2.2. Kinematics

Consider an elastomeric body in its initial configuration (at time $t=0$) that occupies a bounded domain $\Omega_0$, with boundary $\Gamma_0$ and unit outward normal $\mathbf{m}$ as shown in Fig. 1(a). We denote by $\mathbf{X}$ the vector containing the location of an arbitrary point in $\Omega_0$. After deformation, at a given time $t\in\mathbb{R}_+$, the position of the point at the current configuration becomes $\mathbf{x}=\chi(\mathbf{X},t)$, where $\chi$ is a motion mapping from $\Omega_0$ to the current configuration $\Omega_t$. The displacement field $\mathbf{u}(\mathbf{X},t)$ is then defined as the difference of the position vector in the initial configuration from the position vector at the current configuration (i.e., $\mathbf{u}(\mathbf{X},t)=\mathbf{x}-\mathbf{X}$). The gradient of the motion mapping with respect to the initial configuration gives the deformation gradient tensor $\mathbf{F}=\nabla\chi(\mathbf{X},t)$.

Moreover, consider that the body $\Omega_0$ contains a sharp crack $\mathcal{S}_0$ as shown in Fig. 1(a). The sharp crack is regularized through a phase-field variable which varies both temporally and spatially within the body $\Omega_0$. A continuous scalar field, $d(\mathbf{X},t)$, is then assigned to every point in the material domain, ranging between 0 (intact material) to 1 (fully fractured material), thus allowing for a gradual transition from intact to damaged regions over a localization band $\mathcal{B}\subseteq\Omega_0$ as illustrated in Fig. 1(b).

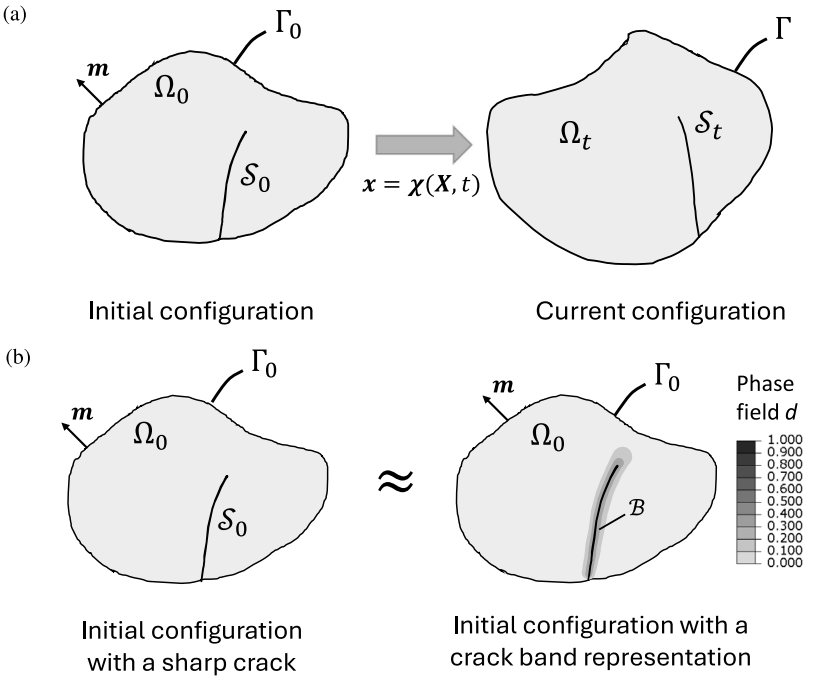



**Fig. 1.** (a) Geometric description of solid body undergoing fracture propagation in the initial (undeformed) and the current (deformed) configurations. (b) Substitution of a sharp crack discontinuity surface in a solid body with a crack band representation of a continuous scalar phase-field, $d$, to describe the growth of fracture within the initial configuration.

### *2.3. The variational phase-field theory and governing differential equations*

Assuming that the body is subject to a body force per unit volume $\mathbf{b}_0$ over $\Omega_0$ and prescribed tractions $\mathbf{t}_0$ on the natural boundary $\Gamma_{0N}$, the potential energy functional $\Pi$ can be expressed in terms of the displacement field and the phase-field variable as

$$\Pi(\mathbf{u}, d) = \int_{\Omega_0} \Psi(\mathbf{F}, \omega(d)) \mathrm{d}V + \int_{\Omega_0} G_c \gamma(d, \nabla d) \mathrm{d}V - \int_{\Omega_0} \mathbf{b}_0 \cdot \mathbf{u} \mathrm{d}V - \int_{\Gamma_{0N}} \mathbf{t}_0 \cdot \mathbf{u} \mathrm{d}A, \quad \mathbf{u} = \mathbf{u}_0 \text{ on } \Gamma_{0D} \tag{1}$$

where $\Psi(\mathbf{F}, \omega(d))$ is the degraded strain energy density function defined in terms of $\mathbf{F}$ and the phase-field degradation function $\omega(d)$, and $G_c$ is the fracture energy required to propagate a crack per unit area. The term $\gamma(d, \nabla d)$ represents the crack surface density function and is defined as $\gamma(d, \nabla d) = \frac{1}{c_\alpha}\left[\frac{1}{l_c}\alpha(d) + l_c \nabla d \cdot \nabla d\right]$ where $c_\alpha = 4\int_0^1 \sqrt{\alpha_{(\beta)}} \mathrm{d}\beta$, and $\nabla d$ is the gradient of the phase-field variable with respect to the initial configuration. The term $\alpha(d)$ is a characteristic function for the phase-field formulation called the crack geometric function, and $l_c$ is the length scale that regularizes the diffuse damage band. It is important to note that in the limit as $l_c$ approaches 0, the original Griffith theory for fracture is recovered. Finally, $\mathbf{u}_0$ is the prescribed displacement on the essential boundary $\Gamma_{0D}$. Note that $\Gamma_{0D} \bigcap \Gamma_{0N} = \varnothing$ and $\Gamma_{0D} \bigcup \Gamma_{0N} = \Gamma_0$.

**Remark 1.** Different versions of the crack geometric function $\alpha(d)$ have been explored in the literature. In this study, we have opted for the straightforward form $\alpha(d) = d^2$ due to its simplicity and ease of implementation. However, it is essential to note that the discussions presented in this manuscript remain applicable to all other versions of this function.

By performing variational operations on the total energy functional $\Pi$, the set of governing differential equations for the displacement field and phase-field variable can be derived with respect to the initial configuration as follows

$$\mathrm{Div}\,(\mathbf{P}) + \mathbf{b}_0 = \mathbf{0} \text{ in } \Omega_0 \qquad \text{and} \qquad \mathbf{P}\mathbf{m} = \mathbf{t}_0 \text{ on } \Gamma_{0N} \tag{2}$$

$$\frac{G_c}{c_\alpha}\left(\frac{\alpha'_{(d)}}{l_c} - 2l_c \nabla d \cdot \nabla d\right) + \frac{\partial \Psi(\mathbf{F}, \omega(d))}{\partial d} = 0 \text{ in } \mathcal{B} \subseteq \Omega_0 \quad \text{and} \quad \frac{2l_c G_c}{c_\alpha} \nabla d \cdot \mathbf{m} = 0 \text{ on } \Gamma_0 \tag{3}$$

where $\mathbf{P} = \frac{\partial \Psi(\mathbf{F}, \omega(d))}{\partial \mathbf{F}}$ is the degraded first Piola–Kirchhoff stress tensor. In the following section, we present the expression for the hyperelastic strain energy density function upon which the material's constitutive response is based. In addition, we also highlight our approach for considering the tension-compression asymmetry in the coupled large-deformation solid mechanics and phase-field constitutive framework.

## 3. Constitutive theory for hyperelastic materials considering tension-compression asymmetry

A neo-Hookean hyperelastic strain energy density function was used in this work. In the absence of phase-field, this function can be expressed in terms of the deformation gradient as follows

$$\psi_{mech}(\mathbf{F}) = \frac{\mu}{2}\left[tr(\mathbf{F}^{\mathsf{T}}\mathbf{F}) - 3 - 2\ln\left[\det\mathbf{F}\right]\right] + \frac{K_B}{2}\left[\ln(\det\mathbf{F})\right]^2 \tag{4}$$

where $\mu$ and $K_B$ represent the rubber (i.e., shear) and bulk moduli of the material, respectively.

In order to express the degraded hyperelastic strain energy density function, the different behaviors of fracture in tension and compression must be taken into account. Treatment of such intricacies is particularly challenging for problems dealing with large deformations. In this work, we follow the method proposed by Tang et al. [17]. As such, we first rewrite the intact strain energy density as a function of principal stretches, $\lambda_i$ where $i = 1$, 2, and 3

$$\psi_{mech} = \frac{1}{2}\mu\sum_{i=1}^{3}(\lambda_i^2 - 1 - 2\ln\lambda_i) + \frac{1}{2}K_B(\ln J)^2 \tag{5}$$

where $J = \det\mathbf{F}$ and can be calculated as $J = \lambda_1\lambda_2\lambda_3$. The strain energy density function can then be expressed by introducing $\psi^+_{mech}$ and $\psi^-_{mech}$, the energies associated with tensile and compression, respectively. These two quantities are defined as

$$\psi^+_{mech} = \psi_{mech}(\lambda_i^+, J^+) \quad \text{where} \quad \lambda_i^+ = \begin{cases}\lambda_i & \text{for } \lambda_i > 1\\ 1 & \text{for } \lambda_i \le 1\end{cases} \quad J^+ = \begin{cases}J & \text{for } J > 1\\ 1 & \text{for } J \le 1\end{cases} \tag{6}$$

and

$$\psi^-_{mech} = \psi_{mech}(\lambda_i^-, J^-) \quad \text{where} \quad \lambda_i^- = \begin{cases}\lambda_i & \text{for } \lambda_i < 1\\ 1 & \text{for } \lambda_i \ge 1\end{cases} \quad J^- = \begin{cases}J & \text{for } J < 1\\ 1 & \text{for } J \ge 1\end{cases} \tag{7}$$

The degraded strain energy density function then can be obtained by applying the degradation function $\omega(d)$ to the tensile component $\psi^+_{mech}$ of the energy [46,47]

$$\Psi(\mathbf{F}, \omega(d)) = \omega(d)\psi^+_{mech} + \psi^-_{mech} \tag{8}$$

Here, we select $\omega(d) = (1-\kappa)(1-d)^2+\kappa$ as the degradation function, with $\kappa$ being a parameter of a very small value that is introduced to avoid computational difficulty when the phase-field approaches 1. Note that in the absence of phase-field (i.e., $d = 0$), the free energy decomposition above reduces to $\Psi = \psi^+_{mech} + \psi^-_{mech}$ which constitutes a constraint to be satisfied by $\psi^+_{mech}$ and $\psi^-_{mech}$. The former refers to the energy with tensile stretching while the latter to the energy with compression, at least in one direction [17].

With the strain energy decomposition defined in Eq. (8) and substituting the assumed form of $\alpha(d)$, Eq. (3) becomes

$$2(1-\kappa)(1-d)\mathcal{H} - \frac{G_c}{l_c}\left(d - l_c^2\nabla d\cdot\nabla d\right) = 0 \text{ in } \mathcal{B}\subseteq\Omega_0 \quad \text{and} \quad \nabla d\cdot\mathbf{m} = 0 \text{ on } \Gamma_0 \tag{9}$$

Here, the local crack driving force $\mathcal{H}(\mathbf{X}, t)$ is introduced to ensure the irreversibility condition $\dot{d} \ge 0$ is enforced and to capture dependence on deformation history. This function has the form $\mathcal{H}(\mathbf{X}, t) = \max_{s\in[0,t]}\psi^+_{mech}(\mathbf{x}, s)$ and represents the maximum value of the tensile strain energy density ever reached [47]. This approach suppresses the negative damage values and ensures that the history variable approach is capable of obtaining the correct homogeneous solution.

The next step is to derive the expression of the first Piola–Kirchhoff stress tensor that arises from the presented degraded strain energy density function. The standard approach follows naturally from its derivation as $\mathbf{P} = \frac{\partial\Psi}{\partial\mathbf{F}} = \omega(d)\frac{\partial\psi^+_{mech}}{\partial\mathbf{F}} + \frac{\partial\psi^-_{mech}}{\partial\mathbf{F}}$. However, as argued in [3], such formulation introduces numerical difficulty due to the non-linearity associated with the decomposition of the strain energy density function in the case of large-deformation problems. An alternative formulation that overcomes this drawback – which was used here – is to derive the first Piola–Kirchhoff stress tensor based on the whole expression of the strain energy density function, i.e., $\mathbf{P} = \frac{\partial\Psi}{\partial\mathbf{F}}$ where $\Psi(\mathbf{F}, \omega(d)) = \omega(d)\psi_{mech}(\mathbf{F})$. Thus, using Eq. (4), we can write

$$\mathbf{P} = \omega(d)\frac{\partial\psi_{mech}}{\partial\mathbf{F}} = [(1-\kappa)(1-d)^2+\kappa]\left[\mu[\mathbf{F} - \mathbf{F}^{-\mathsf{T}}] + K_B\ln(\det\mathbf{F})\mathbf{F}^{-\mathsf{T}}\right] \tag{10}$$

## 4. Finite element formulation

This section outlines the FE formulation of the coupled large-deformation solid-mechanics phase-field problem. First, we present the weak forms. Subsequently, we explore the FE discretization and piecewise approximation techniques pertaining to both the displacement and phase-field functions. Lastly, we present the resulting discrete equations derived from these methods, which will be solved using specific numerical algorithms. Note that the FE formulation is performed with respect to the initial configuration.

### 4.1. Weak forms

In accordance with standard practice, by considering arbitrary vector and scalar fields $\mathbf{w_u}$ and $w_d$, whose components vanish at the corresponding essential displacement and phase-field boundary segments, respectively, the weak form corresponding to the displacement field (i.e., Eq. (2)) and the phase-field (i.e., Eq. (9)) governing differential equations can be written as

$$\begin{aligned}
&\int_{\Omega_0}\left(\frac{\partial \mathbf{w_u}}{\partial \mathbf{X}}\right)^{\mathsf{T}}\mathbf{P}\mathrm{dV}=\int_{\Omega_0}\mathbf{w_u}^{\mathsf{T}}\mathbf{b}_0\mathrm{dV}+\int_{\Gamma_{0N}}\mathbf{w_u}^{\mathsf{T}}\mathbf{t}_0\mathrm{dA}\qquad \forall \mathbf{w_u}\ \text{ with }\ \mathbf{w_u}=\mathbf{0}\ \text{ on }\ \Gamma_{0D}\\
&\int_{\Omega_0}\left[-G_c l_c\left(\frac{\partial w_d}{\partial \mathbf{X}}\right)^{\mathsf{T}}\nabla d+\frac{G_c d}{l_c}w_d\right]\mathrm{dV}+\int_{\Omega_0}w_d\left[-2(1-\kappa)(1-d)\mathcal{H}\right]\mathrm{dV}=0\qquad \forall w_d\ \text{ with }\ w_d=0\ \text{ on }\ \Gamma_0
\end{aligned}\tag{11}$$

where $\frac{\partial \mathbf{w_u}}{\partial \mathbf{X}}$ and $\frac{\partial w_d}{\partial \mathbf{X}}$ are the partial derivatives of the arbitrary vector fields $\mathbf{w_u}$ and $w_d$ with respect to the reference coordinates $\mathbf{X}$, respectively.

### 4.2. Finite element discretization

The weak forms (11) are discretized using multi-field finite elements; that is, the body $\Omega_0$ is discretized into $N_e$ finite elements $\Omega_0^e$ such that $\Omega_0=\bigcup_{i=1}^{N_e}\Omega_0^e$. Without loss of generality, in what follows, we consider the case of two-dimensional (2-D) problems, with the assumption that the three-dimensional (3-D) formulation extends in a straightforward manner.

In 2-D problems, each element node has three nodal degrees of freedom (dofs): two for the displacement field and one for the phase-field variable. Piecewise approximations for the displacement field $\mathbf{u}(\mathbf{X})$ and phase-field $d(\mathbf{X})$ functions are then established inside each element $\Omega_0^e$. These approximations are carried out by employing FE shape functions $N^A_{(\mathbf{X})}$ in terms of the corresponding nodal displacements $\mathbf{u}^A$ and nodal phase-field $d^A$ with the index $A=1,2,\dots,N_{node}$ denoting the nodes of an element. Thus, the approximate displacement vector field $\mathbf{u}^{\mathbf{e}}(\mathbf{X})$ and approximate scalar phase-field $d^e(\mathbf{X})$ in the interior of an element $\Omega_0^e$ with $N_{node}$ nodal points are given by the following equations

$$\mathbf{u}^{\mathbf{e}}(\mathbf{X})=\sum_{A=1}^{N_{node}}N^A_{(\mathbf{X})}\mathbf{u}^A\qquad\text{and}\qquad d^e(\mathbf{X})=\sum_{A=1}^{N_{node}}N^A_{(\mathbf{X})}d^A\tag{12}$$

where $\mathbf{u}^A=\begin{bmatrix}u_1\\u_2\end{bmatrix}^A$ is the vector containing the x- and y- components of the displacement nodal values for node $A$. Then, using a standard Galerkin approach, the arbitrary vector and scalar fields $\mathbf{w}^{\mathbf{e}}_{\mathbf{u}}(\mathbf{X})$ and $w^e_d(\mathbf{X})$, respectively, are interpolated by the same shape functions as $\mathbf{u}^{\mathbf{e}}(\mathrm{X})$ and $d^e(\mathbf{X})$

$$\mathbf{w}^{\mathbf{e}}_{\mathbf{u}}(\mathrm{X})=\sum_{A=1}^{N_{node}}N^A_{(\mathbf{X})}\mathbf{w}^A_{\mathbf{u}}\ \text{ and }\ w^e_d(\mathbf{X})=\sum_{A=1}^{N_{node}}N^A_{(\mathbf{X})}w^A_d\tag{13}$$

Plugging Eq. (13) into (11), we obtain the following element-level equations for each of the finite elements $\Omega_0^e$

$$\begin{cases}\int_{\Omega_0^e}\frac{\partial N^A}{\partial \mathbf{X}}\mathbf{P}\mathrm{dV}=\int_{\Omega_0^e}N^A\mathbf{b}_0\mathrm{dV}+\int_{\Gamma^e_{0N}}N^A\mathbf{t}_0\mathrm{dA}\\ \int_{\Omega_0^e}N^A\left[2(1-\kappa)(1-d^e)\mathcal{H}-\frac{G_c d^e}{l_c}\right]\mathrm{dV}-\int_{\Omega_0^e}G_c l_c\left(\frac{\partial N^A}{\partial \mathbf{X}}\right)^{\mathsf{T}}\nabla d^e\mathrm{dV}=0\end{cases}\tag{14}$$

where $\Gamma^e_{0N}$ is the natural boundary segment in element $\Omega_0^e$. This system of coupled equations can be solved iteratively using an appropriate Newton-type iteration procedure by defining the following element-level residuals for the displacement and phase-field

$$\begin{cases}\mathbf{R}^A_{\mathbf{u}}=-\int_{\Omega_0^e}\frac{\partial N^A}{\partial \mathbf{X}}\mathbf{P}\mathrm{dV}+\int_{\Omega_0^e}N^A\mathbf{b}_0\mathrm{dV}+\int_{\Gamma^e_{0N}}N^A\mathbf{t}_0\mathrm{dA}=\mathbf{0}\\ \mathbf{R}^A_d=\int_{\Omega_0^e}N^A\left[2(1-\kappa)(1-d^e)\mathcal{H}-\frac{G_c d^e}{l_c}\right]\mathrm{dV}-\int_{\Omega_0^e}G_c l_c\left(\frac{\partial N^A}{\partial \mathbf{X}}\right)^{\mathsf{T}}\nabla d^e\mathrm{dV}=0\end{cases}\tag{15}$$

and using the corresponding tangents for the iterative Newton solution procedure

$$\begin{aligned}\mathbf{K}^{AB}_{\mathbf{uu}}&=-\frac{\partial \mathbf{R}^A_{\mathbf{u}}}{\partial \mathbf{u}^B}\ ,\ \mathbf{K}^{AB}_{\mathbf{u}d}=-\frac{\partial \mathbf{R}^A_{\mathbf{u}}}{\partial d^B}\\ \mathbf{K}^{AB}_{d\mathbf{u}}&=-\frac{\partial \mathbf{R}^A_{d}}{\partial \mathbf{u}^B}\ ,\ \mathbf{K}^{AB}_{dd}=-\frac{\partial \mathbf{R}^A_{d}}{\partial d^B}\end{aligned}\tag{16}$$

where $A$ and $B=1,2,\dots,N_{node}$ denoting the nodes of an element.

The finite-element procedures have been incorporated into ABAQUS/Standard [45] through the utilization of the UEL subroutine feature. Within an analysis, the user subroutine UEL is invoked during each iteration within a given increment. The initial nodal coordinates, along with the current guesses of nodal displacements and phase-field, are provided as inputs to the subroutine. In turn, the subroutine requires the computation of nodal residuals (15) and consistent tangents (16) as outputs. In the next section, we present detailed descriptions of two algorithms for solving the nonlinear system of equations. In particular, we discuss differences

**Table 1**
Voigt transformation used for fourth-order tensors [50].

| $m$ | $(i\ ,\ j)$ | $n$ | $(k\ ,\ l)$ |
|---|---|---|---|
| 1 | (1,1) | 1 | (1,1) |
| 2 | (2,1) | 2 | (2,1) |
| 3 | (1,2) | 3 | (1,2) |
| 4 | (2,2) | 4 | (2,2) |

in how the update of the consistent tangent matrix is obtained as it pertains to the standard AM/staggered algorithm and the quasi-Newton based BFGS algorithm.

Note that in this work, we have developed user-defined elements for four-node isoparametric quadrilateral plane-strain elements. Additionally, to circumvent issues associated with volumetric-locking, we employed the *F-bar* method introduced by de Souza Neto et al. [48] for fully integrated elements. Appendix A provides details of the procedure for implementing the *F-bar* method for 2-D plane-strain elements. Additionally, in the AM/staggered UEL subroutine developed, the approach detailed in the work of Molnár and Gravouil [49] was used to implement the numerical integration of the AM iterative process.

**Remark 2.** Eqs. (15) and (16) correspond to the element-level residuals and tangents associated with the coupled large-deformation solid-mechanics phase-field problem discussed in this work. These quantities are implemented in a UEL subroutine using *specific matrix forms*. For the four-node isoparametric quadrilateral plane-strain element developed in this work, the element displacement residual vector and tangent matrix are given by

$$\begin{cases} \mathbf{R_u} = -\int_{\Omega_0^e} \mathsf{G}^{\mathsf{T}}\mathsf{P}\mathrm{dV} + \int_{\Omega_0^e} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{b}_0\mathrm{dV} + \int_{\Gamma^e_{0\,N}} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{t}_0\mathrm{dA} \\ \mathbf{K_{uu}} = \int_{\Omega_0^e} \mathsf{G}^{\mathsf{T}}\mathsf{A}\mathsf{G}\mathrm{dV} \end{cases} \tag{17}$$

where $\mathsf{N}^u = \begin{bmatrix} N^1 & 0 & N^2 & 0 & \dots & N^{N_{node}} & 0 \\ 0 & N^1 & 0 & N^2 & \dots & 0 & N^{N_{node}} \end{bmatrix}$ is the standard shape functions array with $N_{node}$ being the total number of nodes per element, and $\mathsf{P}$, $\mathsf{A}$, and $\mathsf{G}$, are the Voigt-vector form of the first Piola–Kirchhoff stress, Voigt-matrix form of the tangent modulus, and modified matrix form of the shape functions derivatives, respectively, defined as follows

- The first Piola–Kirchhoff stress vector $\mathsf{P} = [P_{11}\ P_{21}\ P_{12}\ P_{22}]^{\mathsf{T}}$ expressed in Voigt notation
- The tangent modulus $\mathsf{A}$ written in Voigt notation following the transformation presented in Table 1 where $\mathsf{A}_{mn} = \mathbb{A}_{ijkl} = \left(\frac{\partial \mathbf{P}}{\partial \mathbf{F}}\right)_{ijkl}$. Appendix A provides the exact form of the fourth-order tangent modulus tensor, $\mathbb{A}_{ijkl}$, from which $\mathsf{A}$ can be readily determined.
- G-matrix given by the following array

$$\mathsf{G} = \begin{bmatrix} \frac{\partial N^1}{\partial X_1} & 0 & \frac{\partial N^2}{\partial X_1} & 0 & \dots & \frac{\partial N^{N_{node}}}{\partial X_1} & 0 \\ 0 & \frac{\partial N^1}{\partial X_1} & 0 & \frac{\partial N^2}{\partial X_1} & \dots & 0 & \frac{\partial N^{N_{node}}}{\partial X_1} \\ \frac{\partial N^1}{\partial X_2} & 0 & \frac{\partial N^2}{\partial X_2} & 0 & \dots & \frac{\partial N^{N_{node}}}{\partial X_2} & 0 \\ 0 & \frac{\partial N^1}{\partial X_2} & 0 & \frac{\partial N^2}{\partial X_2} & \dots & 0 & \frac{\partial N^{N_{node}}}{\partial X_2} \end{bmatrix} \tag{18}$$

On the other hand, the phase-field residual and tangent matrix are defined as

$$\begin{cases} \mathbf{R}_d = -\int_{\Omega_0^e} \left\{ \left[ \frac{G_c}{l_c}\mathbf{N}^d\bar{\mathbf{d}}^{\mathsf{T}} - 2(1-\kappa)(1-\mathbf{N}^d\bar{\mathbf{d}}^{\mathsf{T}})\mathcal{H} \right](\mathbf{N}^d)^{\mathsf{T}} + G_c l_c (\mathbf{B}^d)^{\mathsf{T}}\mathbf{B}^d\bar{\mathbf{d}}^{\mathsf{T}} \right\}\mathrm{dV} \\ \mathbf{K}_{dd} = \int_{\Omega_0^e} \left\{ \left[ \frac{G_c}{l_c} + 2\mathcal{H} \right](\mathbf{N}^d)^{\mathsf{T}}\mathbf{N}^d + G_c l_c (\mathbf{B}^d)^{\mathsf{T}}\mathbf{B}^d \right\}\mathrm{dV} \end{cases} \tag{19}$$

where $\mathbf{N}^d = [N^1\ \dots\ N^{N_{node}}]$ is the vector of the shape functions, $\mathbf{B}^d = \begin{bmatrix} \frac{\partial N^1}{\partial X_1} & \frac{\partial N^2}{\partial X_1} & \dots & \frac{\partial N^{N_{node}}}{\partial X_1} \\ \frac{\partial N^1}{\partial X_2} & \frac{\partial N^2}{\partial X_2} & \dots & \frac{\partial N^{N_{node}}}{\partial X_2} \end{bmatrix}$ is the array of shape function derivatives, and $\bar{\mathbf{d}} = [d^1\ d^2\ \dots d^{N_{node}}]$ is the vector containing phase-field nodal values within each element.

## 5. Numerical algorithms for solving the coupled large deformation solid mechanics phase-field problem

The process for solving the system of nonlinear Eqs. (15) typically follows an incremental approach. This involves discretizing the total time interval $[0, T]$ into discrete time increments $[t_n, t_{n+1}]_{n\in[0,N-1]}$ of length $\Delta t := t_{n+1} - t_n$ such that $t_0 = 0$ and $t_N = T$, and evaluating all state variables within the increments. With all state variables known at the instant $t_n$ (i.e., the deformation gradient

$\mathbf{F}_n$, the phase-field variable, $d_n$, and the history term $\mathcal{H}_n = \mathcal{H}(\mathbf{X}, t_n)$), the system of equations is then solved for the nodal unknowns $\mathbf{u}_{n+1}$ and $d_{n+1}$ at each incremental step until the final increment $t_N = T$ is reached.

This section outlines the procedure for implementing the BFGS algorithm as a quasi-Newton-based approach that combines the robustness of the AM/staggered algorithm with the efficiency of the full monolithic approach. We begin by first presenting how the AM/staggered approach is typically implemented and then outline the framework for implementing the BFGS approach as a more efficient alternative to its staggered counterpart.

### 5.1. AM/staggered algorithm

The energy functional in Eq. (1) is non-convex with respect to the unknown displacement and phase-field when considered simultaneously. However, when one variable is held fixed, the functional becomes convex with respect to the other. As a result, the system of nonlinear coupled Eqs. (15) can be decoupled by fixing one of the dofs (either phase-field or displacement) and then solving for the other in an alternating manner, resulting in the so-called AM or staggered scheme. The AM/staggered approach involves the following steps, which are embedded in an iterative procedure.

1. At the $k$th iteration of a specific time increment $[t_n, t_{n+1}]$, the element nodal displacement vector $\bar{\mathbf{u}}_k = \{\mathbf{u}_k^A\}_{A\in[1,N_{node}]}$ is solved with the phase-field dofs fixed as obtained from the previous iteration; that is, the residual in Eq. (17) is evaluated as

$$\mathbf{R}_{\mathbf{u}} = -\int_{\Omega_0^e} \mathsf{G}^{\mathsf{T}}\mathsf{P}_k \mathrm{dV} + \int_{\Omega_0^e} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{b}_0 \mathrm{dV} + \int_{\Gamma_{0N}^e} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{t}_0 \mathrm{dA} = \mathbf{0}, \qquad \mathsf{P}_k = \mathsf{P}(\bar{\mathbf{u}}_k, \bar{\mathbf{d}}_{k-1}) \tag{20}$$

   Eq. (20) can then be solved using a Newton–Raphson iteration scheme by defining the following linearized equation $\mathbf{K}_{\mathbf{uu}}\Delta\bar{\mathbf{u}} = \mathbf{R}_{\mathbf{u}}$, where $\Delta\bar{\mathbf{u}} = \bar{\mathbf{u}}_k - \bar{\mathbf{u}}_{k-1}$ is the difference between the displacement solution at the previous iteration and the solution at the current iteration, and $\mathbf{K}_{\mathbf{uu}}$ is the tangent matrix defined in Eq. (17).
2. Then, the nodal phase-field vector at the $k$th iteration, $\bar{\mathbf{d}}_k$, is solved with the updated nodal displacements $\bar{\mathbf{u}}_k$; that is, the residual in Eq. (19) is evaluated as

$$\mathbf{R}_d = -\int_{\Omega_0^e} \left\{ \left[ \frac{G_c}{l_c}\mathbf{N}^d\bar{\mathbf{d}}_k^{\mathsf{T}} - 2(1-\kappa)(1-\mathbf{N}^d\bar{\mathbf{d}}_k^{\mathsf{T}})\mathcal{H}_k \right](\mathbf{N}^d)^{\mathsf{T}} + G_c l_c (\mathbf{B}^d)^{\mathsf{T}}\mathbf{B}^d\bar{\mathbf{d}}_k^{\mathsf{T}} \right\} \mathrm{dV} = \mathbf{0} \tag{21}$$

   which can be solved with the corresponding linearized form $\mathbf{K}_{dd}\Delta\bar{\mathbf{d}} = \mathbf{R}_d$, where similarly, $\Delta\bar{\mathbf{d}}$ is the difference between the phase-field solution at the previous iteration and the solution at the current iteration, and $\mathbf{K}_{dd}$ is the tangent defined in Eq. (19).

This procedure is repeated until the final solution converges.

### 5.2. BFGS quasi-Newton algorithm

In the introductory section, it was highlighted that the BFGS quasi-Newton algorithm has demonstrated superior efficiency in solving the coupled solid mechanics phase-field problem under linear elastic conditions [23]. However, other than Liu et al. [51], who proposed a quasi-Newton scheme for the solution of the coupled governing equations in large-deformation solid shells within linear elastic finite elements, there has been no detailed investigation into its application in scenarios involving hyperelastic large-deformation kinematics for elastomeric/polymeric materials. The following highlights the main considerations involved in implementing the BFGS method to solve large-deformation solid mechanics-phase-field problems.

Let $\mathbf{z}^e$ to be the vector of element unknown nodal values, defined as $\mathbf{z}^e = [\bar{\mathbf{u}}, \bar{\mathbf{d}}]^{\mathsf{T}}$ where $\bar{\mathbf{u}}$ and $\bar{\mathbf{d}}$ are the element nodal displacement and phase-field values as defined earlier. Eq. (15) can then be written in a compact form as

$$\mathbf{g}(\mathbf{z}^e) = \mathbf{0} \tag{22}$$

Eq. (22) can be linearized with respect to $\mathbf{z}^e$ using the Newton–Raphson method

$$\mathbf{g} + \frac{\partial \mathbf{g}}{\partial \mathbf{z}^e}\delta\mathbf{z}^e = \mathbf{g} - \mathbf{K}^{\mathbf{e}}\delta\mathbf{z}^e = \mathbf{0} \quad \text{where} \quad \delta\mathbf{z}^e = [\delta\bar{\mathbf{u}}, \delta\bar{\mathbf{d}}]^{\mathsf{T}} \tag{23}$$

The tangent matrix $\mathbf{K}^{\mathbf{e}}$ (oftentimes referred to as Jacobian matrix) is generally not symmetric since the off-diagonals in Eq. (16) may not be equal (i.e., $\mathbf{K}_{\mathbf{ud}}^{AB} \neq \mathbf{K}_{d\mathbf{u}}^{AB}$) due to the presence of the history variable $\mathcal{H}$ in Eq. (15). However, for weakly coupled problems, the off-diagonal block matrices can be omitted [23], and Eq. (23) then reduces to

$$\begin{bmatrix} \mathbf{K}_{\mathbf{uu}} & 0 \\ 0 & \mathbf{K}_{dd} \end{bmatrix} \begin{bmatrix} \delta\bar{\mathbf{u}} \\ \delta\bar{\mathbf{d}} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{\mathbf{u}} \\ \mathbf{R}_d \end{bmatrix} \tag{24}$$

where the relevant tangent matrices and residual vectors are used as defined in Eqs. (17) and (19).

In a quasi-Newton procedure, an approximation to the secant tangent between two successive iterations for a specific time step $t_n$ is estimated. The process involves generating a sequence of refined approximations to the Jacobian matrix at iteration $k$, denoted as $\tilde{\mathbf{K}}_k^e$, which adhere to the secant condition

$$\mathbf{g}(\mathbf{z}_k^e) - \mathbf{g}(\mathbf{z}_{k-1}^e) = \tilde{\mathbf{K}}_k^e(\mathbf{z}_k^e - \mathbf{z}_{k-1}^e) \quad \text{or} \quad \delta\mathbf{g}_k = \tilde{\mathbf{K}}_k^e\delta\mathbf{z}_k^e \tag{25}$$

**Table 2**
Material properties used to conduct simulations for each case investigated.

| | μ (MPa) | $K_B$ (MPa) | $l_c$ (mm) | $G_c$ (N/mm) | $\kappa$ |
|---|---|---|---|---|---|
| Single-edge notch (Uniaxial tension) (Section 6.1.1) | 80 770 | 175 000 | 0.024 | 2.7 | $10^{-8}$ |
| Single-edge notch (Pure shear) (Section 6.1.2) | 80 770 | 175 000 | 0.015 | 2.7 | $10^{-8}$ |
| Notched beam under three-point bending (Section 6.2) | 11 539 | 25 000 | 2.5 | 0.228 | $10^{-8}$ |

where $\delta\mathbf{g}_k = \mathbf{g}_k - \mathbf{g}_{k-1}$ and $\delta\mathbf{z}^e_k = \mathbf{z}^e_k - \mathbf{z}^e_{k-1}$ so that $\tilde{\mathbf{K}}^e_k$ approaches $\mathbf{K}^e_k$ as the iterations proceed.

The BFGS method provides a computationally efficient approach to create a series of approximations to $[\tilde{\mathbf{K}}^e_k]^{-1}$ that fulfill Eq. (25) while preserving the symmetry and positive definiteness of $\tilde{\mathbf{K}}^e_k$. This is achieved by updating $[\tilde{\mathbf{K}}^e_{k-1}]^{-1}$ to $[\tilde{\mathbf{K}}^e_k]^{-1}$ using a "product plus increment" formula

$$[\tilde{\mathbf{K}}^e_k]^{-1} = \left(\mathbf{I} - \frac{\delta\mathbf{z}^e_k \delta\mathbf{g}_k^{\mathsf{T}}}{\delta\mathbf{z}^{e^{\mathsf{T}}}_k \delta\mathbf{g}_k}\right)[\tilde{\mathbf{K}}^e_{k-1}]^{-1}\left(\mathbf{I} - \frac{\delta\mathbf{z}^e_k \delta\mathbf{g}_k^{\mathsf{T}}}{\delta\mathbf{z}^{e^{\mathsf{T}}}_k \delta\mathbf{g}_k}\right)^{\mathsf{T}} + \frac{\delta\mathbf{z}^e_k \delta\mathbf{z}^{e^{\mathsf{T}}}_k}{\delta\mathbf{z}^{e^{\mathsf{T}}}_k \delta\mathbf{g}_k} \tag{26}$$

where $\mathbf{I}$ is the second-order identity tensor.

In the actual implementation of this version of the BFGS method, the algorithm does not store each $[\tilde{\mathbf{K}}^e_k]^{-1}$ individually. Instead, it employs a "kernel" matrix, denoted as $[\tilde{\mathbf{K}}^e_m]^{-1}$, and updates it by pre-multiplying and post-multiplying it with the terms $\left(\mathbf{I} - \frac{\delta\mathbf{z}^e_j \delta\mathbf{g}_j^{\mathsf{T}}}{\delta\mathbf{z}^{e^{\mathsf{T}}}_j \delta\mathbf{g}_j}\right)$ and $\left(\mathbf{I} - \frac{\delta\mathbf{z}_j \delta\mathbf{g}_j^{\mathsf{T}}}{\delta\mathbf{z}^{e^{\mathsf{T}}}_j \delta\mathbf{g}_j}\right)^{\mathsf{T}}$, respectively for $j = m+1, m+2, \ldots, k$. This approach involves inner products of vectors and scaling of vectors by constants, which makes the method computationally efficient. However, it is advisable to limit the number of such operations to maintain computational efficiency. Typically, a new kernel matrix is created and stored after several iterations. In the ABAQUS/Standard implementation, this kernel is the actual Jacobian matrix and is regenerated after a specified number of iterations without achieving convergence, with a default limit of 8 iterations. ABAQUS/Standard does not recreate the kernel unless this limit is exceeded, allowing the same kernel to be used for multiple increments if BFGS updates are successful. It should be noted that the local convergence rate of quasi-Newton techniques falls short of the quadratic convergence rate seen with full Newton–Raphson iterations, which employ an algorithmically consistent Jacobian matrix [52,53]. Nevertheless, quasi-Newton methods can offer greater computational efficiency as they eliminate the need to construct and invert a new Jacobian matrix during each iteration. Furthermore, these methods can help avoid issues related to the singularity of the global tangent stiffness matrix.

The BFGS algorithm is activated in an ABAQUS job by the following prompt

```
*SOLUTION TECHNIQUE, TYPE=QUASI-NEWTON, REFORM KERNEL=25
```

where the option "reform kernel" is used to set the number of iterations above which the Jacobian matrix $[\tilde{\mathbf{K}}^e_m]^{-1}$ is reformed. In this work, this number was set to 25. Note that the ABAQUS-specific forms of the residual and tangent matrix required for the BFGS UEL implementation are provided in Appendix B. Additionally, an algorithm detailing the procedure followed to solved the coupled large-deformation phase-field solid mechanics problem using the two approaches discussed in this work (i.e., BFGS vs. AM/Staggered) is provided in Appendix C.

## 6. Representative numerical examples — results and discussion

This section investigates the computational efficiency of the BFGS algorithm in solving phase-field problems within the large deformation context. We compare the BFGS approach to the AM/staggered approach through a series of benchmark examples. In particular, we consider three representative numerical examples of elastomers undergoing deformation in various modes: a single-edge notch sample under uniaxial tension (SNUT), a single-edge notch sample under pure shear (SNPS), and a notched beam under three-point bending (3PB). These examples aim to demonstrate the efficiency of the BFGS algorithm in solving large-deformation solid mechanics problems with varying complexity and fracture propagation patterns. Note that the 3PB case was chosen to illustrate the efficacy of the tension-compression decomposition outlined in Section 3. The geometry and boundary conditions for each problem configuration are presented in Fig. 2, and the constitutive properties assigned to the materials used in each example are provided in Table 2. The material properties for the single-edge notch examples were taken from [47], whereas those corresponding to the 3PB were taken from [54].

The simulations using the AM/staggered solver were performed with varying time increments in order to thoroughly appreciate the differences in computational efficiency between the AM/staggered and BFGS solvers. In what follows, we use the notation below in reference to the type of simulation performed.

- BFGS-ATS, designating a simulation run with the BFGS solver enriched with an automated time-stepping algorithm (ATS) that is activated whenever the Jacobian of the finite element is negative. The ATS algorithm consists of reducing the time increment by half at each integration point.
- SNUT-N2, SNUT-N3, and SNUT-N4, designating SNUT simulations run with the AM/staggered solver with a single time increment of $\Delta t = 10^{-2}$ s, $\Delta t = 10^{-3}$ s, and $\Delta t = 10^{-4}$ s, respectively.

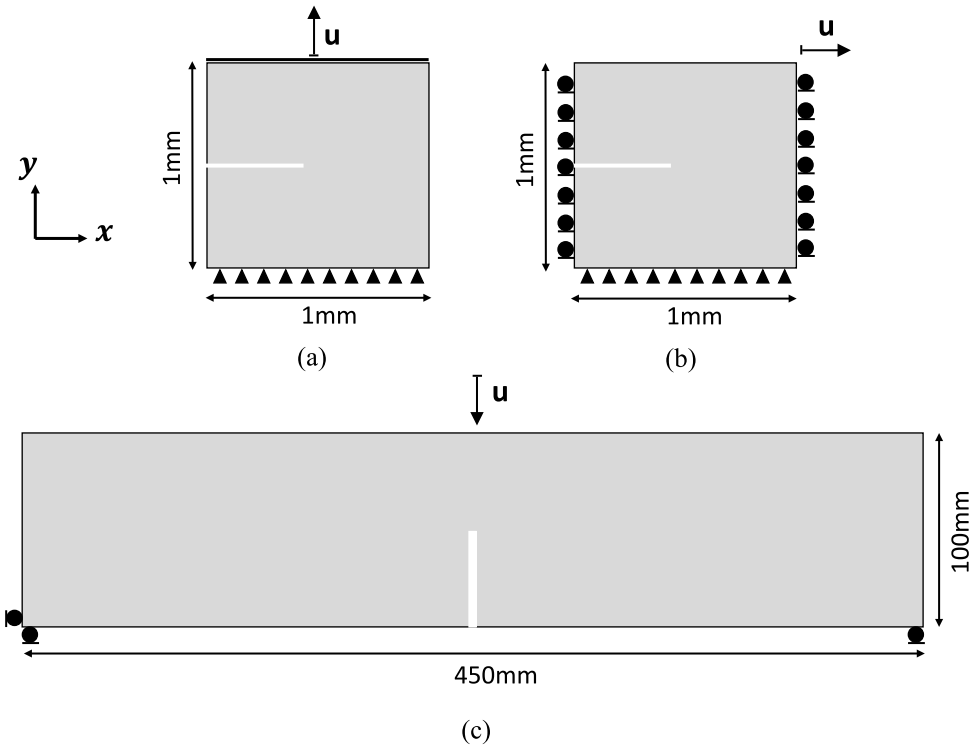



**Fig. 2.** Sample geometry and boundary conditions for the single-edge notch problem loaded by a prescribed displacement $u$ under (a) uniaxial tension and (b) pure shear. (c) Sample geometry and boundary conditions for the notched beam under three point bending loaded by a prescribed displacement $u$.

- SNUT-N2N4 and SNUT-N2N5, designating SNUT simulations run with the AM/staggered solver with double time-step incrementation: an initial time increment of $\Delta t = 10^{-2}$ s until the critical increment, then $\Delta t = 10^{-4}$ s or $\Delta t = 10^{-5}$ s, respectively, for all subsequent increments until complete fracture.
- SNPS-N2N3, SNPS-N2N4, and SNPS-N2N5, designating SNPS simulations run with the AM/staggered solver with double time-step incrementation: an initial time increment of $\Delta t = 10^{-2}$ s until the critical increment, then $\Delta t = 10^{-3}$ s, $\Delta t = 10^{-4}$ s, or $\Delta t = 10^{-5}$ s, respectively, for all subsequent increments until complete fracture.
- 3PB-N3, designating a 3PB simulation run with the AM/staggered solver with a single time increment of $\Delta t = 10^{-3}$ s.
- 3PB-N2N4 and 3PB-N2N5, designating 3PB simulations run with the AM/staggered solver with double time-step incrementation: an initial time increment of $\Delta t = 10^{-2}$ s until the critical increment, then $\Delta t = 10^{-4}$ s or $\Delta t = 10^{-5}$ s, respectively, for all subsequent increments until complete fracture.

A critical aspect of this study involved maintaining equivalent computational power across all simulations to ensure a fair assessment of the solutions' efficiency. To achieve this, we utilized the Cray supercomputer at the Advanced Research Computing (ARC) facility at Virginia Tech. This setup employed parallel computation across 128 AMD EPYC 7702 CPU cores, guaranteeing that any observed differences in performance were solely due to the algorithms themselves, rather than variations in computational resources.

### 6.1. Single-edge notch specimen

In this section, we examine the case of a single-edge notch elastomeric sample loaded on its top edge quasi-statically by a prescribed displacement in two directions: vertical, exemplifying a uniaxial tensile mode (SNUT), and horizontal, exemplifying a pure shear mode (SNPS), as shown in Fig. 2a and b, respectively. Various simulations were conducted, each featuring a unique numerical setup. This included different solver types (i.e., BFGS vs. AM/staggered), as well as different time-step incrementation methods.

#### 6.1.1. Uniaxial tension

First, we investigate uniaxial tension in the SNUT case to showcase a mode-I fracture. As mentioned in Section 4.2, we discretized the problem using linear plane-strain four-node quadrilateral elements. The critical path, where crack propagation was expected, was meshed using an element size that was at least six times smaller than the length-scale $l_c$, while the rest of the sample was meshed using a larger size, resulting in 14,205 elements as presented in Fig. 3(a). Various simulations were performed, each with a distinct

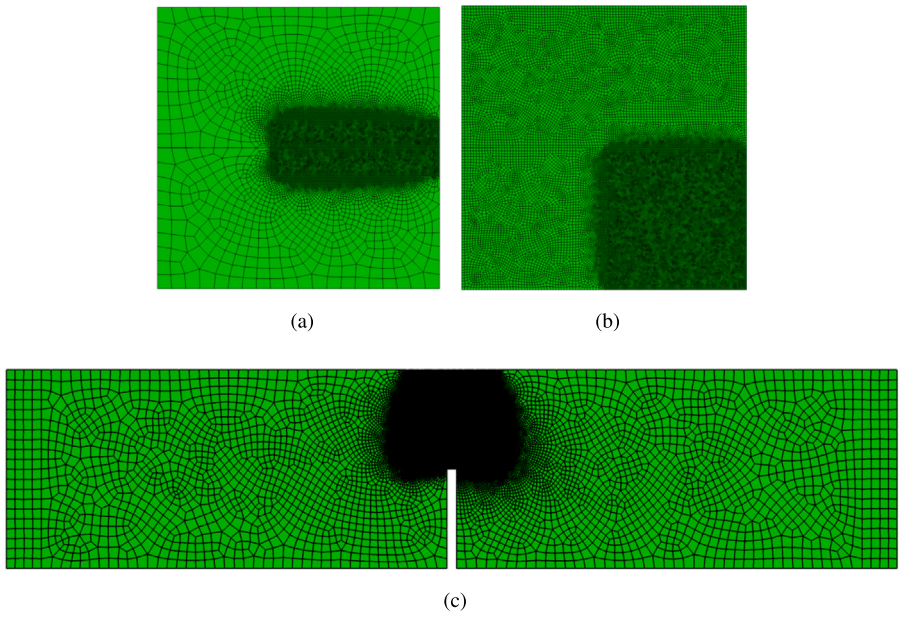



**Fig. 3.** FE mesh used for each of the three representative numerical examples. The element size was taken to be at least six times smaller than the length-scale (i.e., $l_c/6$). Only the region corresponding to the critical path along which crack was expected to propagate was meshed with the aforementioned size. The rest of the geometry was meshed using a relatively larger element size. (a) The single-edge notched sample under uniaxial tension (SNUT) consisting of 14 205 finite elements; (b) the single-edge notched sample under pure shear (SNUT) consisting of 30 769 finite elements; and (c) the notched beam sample under three-point bending (3PB) consisting of 60 695 finite elements.

**Table 3**
Computational time (in minutes) needed to solve the coupled displacement-phase-field problem for each of the simulations performed.

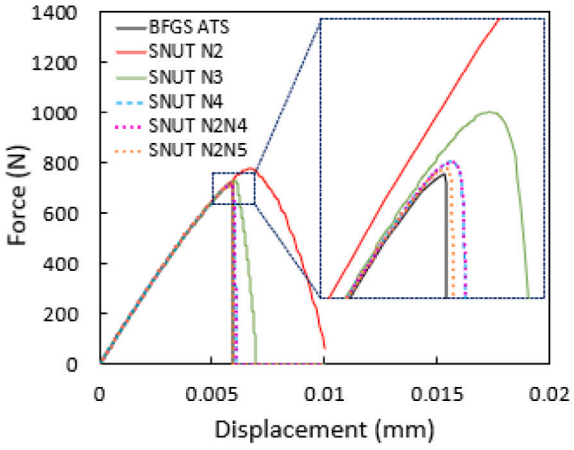
| | **AM/staggered** | | | | | **BFGS** |
|---|---|---|---|---|---|---|
| | SNUT-N2 | SNUT-N3 | SNUT-N4 | SNUT-N2N4 | SNUT-N2N5 | BFGS-ATS |
| SNUT | 8 | 74 | 683 | 304 | 3068 | 29 |
| | SNPS-N2N3 | SNPS-N2N4 | SNPS-N2N5 | | | BFGS-ATS |
| SNPS | 97 | 437 | 4055 | | | 356 |
| | 3PB-N3 | 3PB-N2N4 | 3PB-N2N5 | | | BFGS-ATS |
| 3PB | 224 | 1902 | 8640 | | | 136 |

numerical configuration: type of solver used (i.e., BFGS vs. AM/staggered), and type of time-step incrementation (i.e., BFGS-ATS, SNUT-N2, SNUT-N3, SNUT-N4, SNUT-N2N4, and SNUT-N2N5).

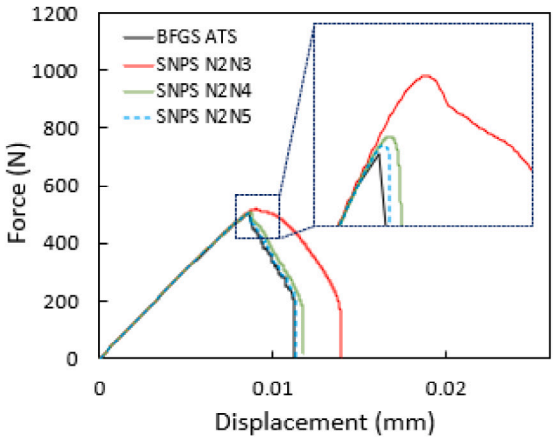
The force–displacement curves for the various simulations performed are presented in Fig. 4a. The computational times required to obtain these solutions are listed in Table 3. It was found that in order to capture the sudden drop in stress that is expected due to unstable crack propagation, the AM/staggered approach required extremely small time increments, i.e, $\Delta t = 10^{-4}$ s, resulting in a large number of increments and prolonged computational time. In contrast, the BFGS algorithm was able to capture the catastrophic failure with fewer increments and less computational time. This is evident in Fig. 4a, where it can be seen that increasingly smaller time increments were required for the AM/staggered solution to converge to the BFGS-based solution. Specifically, the BFGS solution was computed much faster, taking only 29 min compared to 683 min (11.4 h) for the AM/staggered approach (as shown by the curves corresponding to BFGS-ATS and SNUT-N4 in Fig. 4a). Note that even after using a larger time step for the first couple of increments leading to the critical time increment, i.e., SNUT-N2N4 configuration, the AM/staggered algorithm still required 4.5 more hours to arrive at the solution that the BFGS algorithm computed in 29 min (see Table 3). The solution computed by the SNUT-N2N5 solver took 3068 min (51 h), corresponding to a whopping 106 increase in computational time compared the BFGS solver. Fig. 5 shows the contours of the phase-field variable computed using the BFGS solver with ATS control and the AM/staggered solver using $\Delta t = 10^{-2}$, $10^{-3}$, and $10^{-4}$ s. Again, the phase-field profile calculated using the AM/staggered solver with the smallest time increment is seen to provide the closest match to the one computed using the BFGS solver.

### *6.1.2. Pure shear*

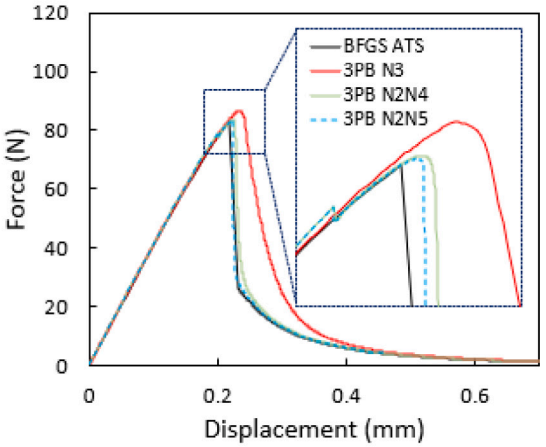
In the SNPS case, we applied a quasi-static horizontal displacement on the top-edge of the rubber specimen to simulate pure shear loading, as illustrated in Fig. 2b. Similar to the SNUT case, we used linear plane-strain quadrilateral elements and discretized




(a) Single-edge notch under uniaxial tension (SNUT)




(b) Single-edge notch under pure shear (SNPS)




(c) Notched beam under three-point bending (3PB)

**Fig. 4.** Force-displacement plots for the single-edge notch under uniaxial tension problem obtained using the BFGS and the AM/staggered algorithms. BFGS-ATS denotes the BFGS solver with an automated time-stepping algorithm, while the affix -N2, -N3, and -N4 refer to the AM/staggered solver with single time increments of $10^{-2}$ s, $10^{-3}$ s, and $10^{-4}$ s, respectively. The affix -N2N4 and -N2N5 refers to the AM/staggered solver with double time incrementation: an initial time increment of $10^{-2}$ s until the critical increment, followed by $10^{-4}$ s or $10^{-5}$ s for all subsequent increments until complete fracture, respectively.

the problem with an element size that is at least six times smaller than the length-scale in the critical region where crack was expected to propagate, resulting in a total of 30 769 finite elements as shown in Fig. 3(b).

Fig. 4b shows the force–displacement curves obtained for each particular case, and Table 3 lists the corresponding computational times required to solve each problem. As in the SNUT case, the AM/staggered solver required increasingly smaller time increments to produce the same response as the BFGS algorithm. More precisely, the AM/staggered algorithm required a reduction in time increment to $10^{-5}$ s after the critical region (i.e., SNPS-N2N5) to closely match the force–displacement curve of the BFGS solution. In terms of computational cost, the BFGS solver required only 356 min (5.93 h), in contrast to the AM/staggered solver (i.e., SNPS-N2N5), which needed 4055 min (67.6 h), making it 11.4 times as time-consuming as the BFGS. Fig. 6 illustrates the phase-field contours for each case examined in this section. The fracture profile computed by the BFGS solver is qualitatively most similar to the one obtained using the AM/staggered solver with the smallest time increment (i.e., SNPS-N2N5).

### 6.2. Beam under three-point bending

In this section, we examine the case of a notched elastomeric beam loaded under three-point bending as shown in Fig. 2c. We used plane-strain quadrilateral elements to discretize the problem into 60 695 finite elements, with an element size at least six times as small as the length-scale, as shown in Fig. 3(c). We performed the following simulations: BFGS-ATS, 3PB-N3, 3PB-N2N4, and 3PB-N2N5.

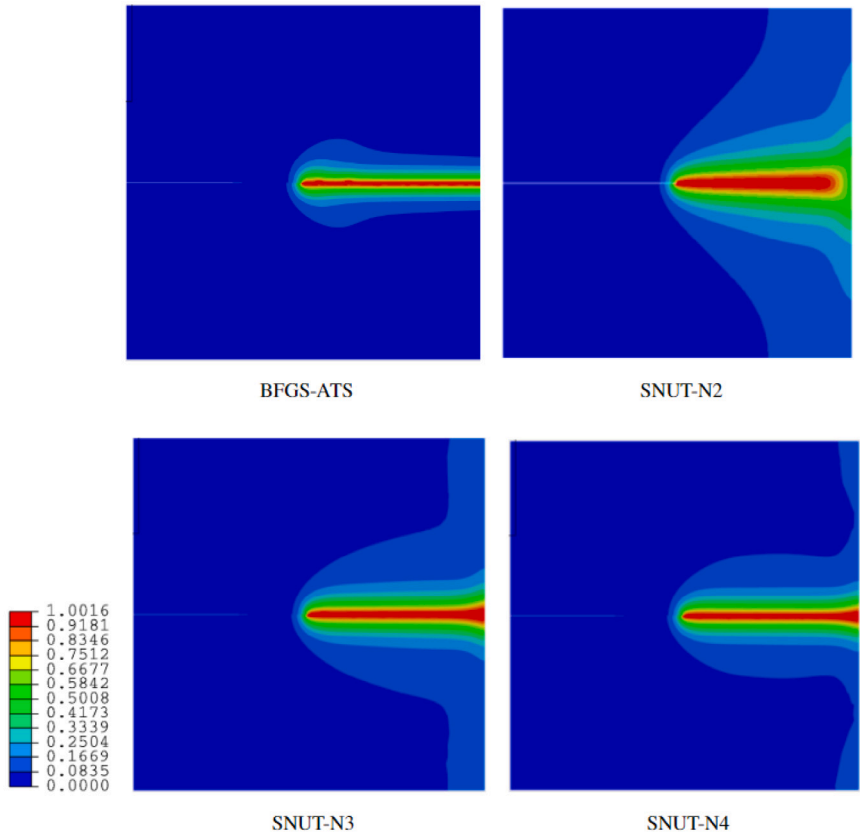



**Fig. 5.** Phase-field contours for the single-edge notch under uniaxial tension (SNUT) problem. Four out of the six simulations are displayed: BFGS-ATS, which denotes the BFGS solver with an automated time-stepping algorithm, and -N2, -N3, -N4, corresponding to the AM/staggered solver with varying time increments of $10^{-2}$ s, $10^{-3}$ s, and $10^{-4}$ s, respectively.

Fig. 4c presents the force–displacement curves for each case in the 3PB example, while Table 3 details the computational times needed for each configuration. Similar to the SNUT and SNPS examples, the AM/staggered solver required smaller and smaller time increments to match the force–displacement response achieved by the BFGS method. Notably, the closest match to the BFGS solver's result (specifically, 3PB-N2N5) took 8640 min (144 h) with the AM/staggered solver, compared to just 136 min (2.3 h) with the BFGS solver. This represents a significant 193 percent increase in computational time. Furthermore, Fig. 7 illustrates the phase-field contours for each case in this section. The phase-field profile calculated by the BFGS algorithm shows the closest qualitative resemblance to that obtained using the AM/staggered solver with the smallest time increment (i.e., 3PB-N2N5).

### 6.3. Discussion and future outlook

The results from previous sections clearly indicate that the BFGS algorithm offers superior efficiency over the AM/staggered approach in solving the system of nonlinear solid-mechanics phase-field governing Eqs. (2) and (3). This advancement is significant, considering the traditionally high computational demands of phase-field damage frameworks, which have been an obstacle in their development. Moreover, the implementation of the BFGS algorithm as a more elegant alternative to the AM/staggered solver has demonstrated its potential in accurately and efficiently simulating complex fracture patterns in hyperelastic materials, as exemplified in Section 6.2. Such capability marks a notable progression in addressing the complexities of fracture propagation and the accuracy required in predictive simulations. The findings of this study shift the narrative from grappling with computational burdens – which has so far been the major focus of recent developments – to harnessing the intrinsic strengths of the phase-field approach in capturing crack propagation in complex material systems. Consequently, this work enables the application of the phase-field approach to more accurately represent soft material behaviors under coupled mechanical–environmental conditions, such as thermo-oxidation [55], photo-oxidation [56], or other types of nonlinear dissipative heterogeneous systems such as asphalt concrete [57–59] or composites such as fiber-reinforced composites [60], thereby extending its utility across a wider spectrum of multi-physics problems. Furthermore, the framework can also be extended to account problems related to fatigue fracture in hyperelastic materials [7,61].

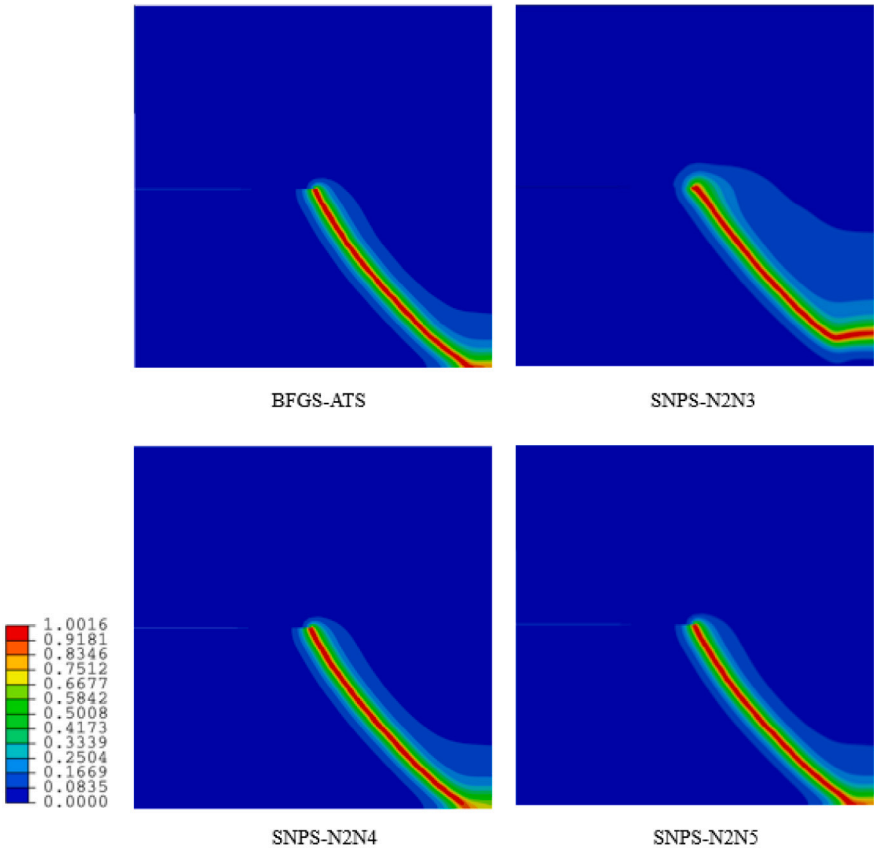



**Fig. 6.** Phase-field contours for the single-edge notch under pure shear (SNPS) problem. BFGS-ATS denotes the BFGS solver with an automated time-stepping algorithm, and -N2N3, -N2N4, -N2N5, refer to the AM/staggered solver with double time incrementation: an initial time increment of $10^{-2}$ s until the critical increment, followed by $10^{-3}$ s, $10^{-4}$ s, or $10^{-5}$ s for all subsequent increments until complete fracture, respectively.

## 7. Conclusions

In this work, we have introduced the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm as an alternative approach to solve the nonlinear system of governing equations resulting from large-deformation solid mechanics and phase-field problems in the context of hyperelastic materials. The phase-field approach, while capable of simulating complex fracture patterns, is hampered by the substantial computational cost required to solve the displacement-phase-field minimization problem. Here, we propose the BFGS algorithm as an alternative method for solving the associated weak forms, and numerically implement the framework in a user-defined element (UEL) subroutine within Abaqus/Standard. Differences pertaining to the numerical solution procedure of the BFGS solver were presented in detail in comparison with the more traditional alternating minimization (AM)/staggered algorithm.

Several benchmark examples were studied, exemplifying elastomers undergoing deformation in various modes: a single-edge notch sample under uniaxial tension (SNUT), a single-edge notch sample under pure shear (SNPS), and a notched beam under three point bending (3PB). These examples were intended to demonstrate the efficiency of the BFGS algorithm in solving large-deformation solid mechanics problems with varying complexity and fracture propagation patterns. The 3PB case, in particular, was chosen so as to illustrate the efficacy of the incorporated tension-compression decomposition of the associated hyperelastic strain energy density function. Across all three representative numerical examples, the BFGS algorithm consistently outperformed the AM/staggered method:

- For the SNUT case, BFGS was 23.5 times faster than the most accurate AM/staggered solution (29 min vs. 683 min).
- In the SNPS scenario, BFGS demonstrated an 11.4-fold speed increase (356 min vs. 4055 min).
- The 3PB example showed the most dramatic improvement, with BFGS computing the solution 63.5 times faster than the AM/staggered approach (136 min vs. 8640 min).

Importantly, our results demonstrate that the BFGS algorithm improves the computational efficiency in solving the coupled large deformation phase-field-displacement solid mechanics problems while reproducing the same fracture behavior and force–displacement responses as the AM/staggered solutions with the most refined time step. This combination of accuracy and efficiency

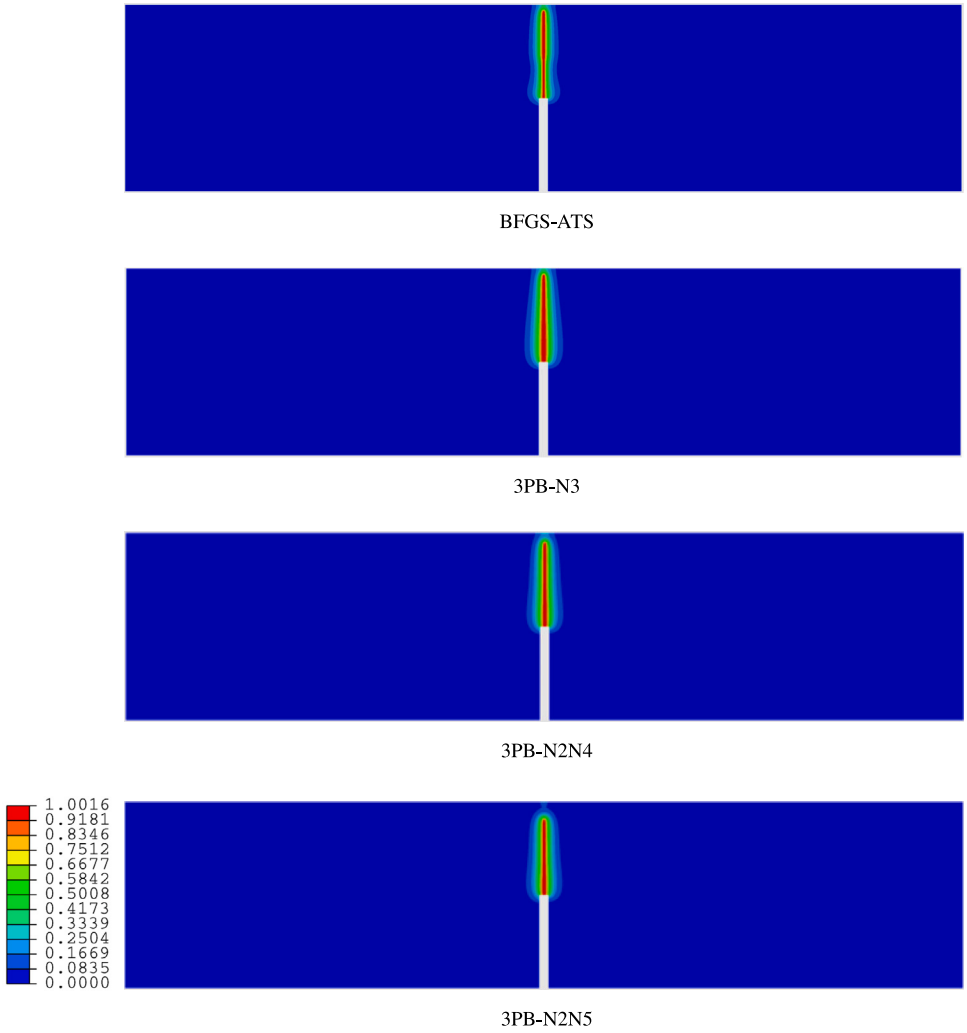



**Fig. 7.** Phase-field contours for the notched beam under three-point bending (3PB) problem. BFGS-ATS denotes the BFGS solver with an automated time-stepping algorithm, while 3PB-N3 refers to the AM/staggered solver with a single time increment of $10^{-3}$ s all throughout the simulation. 3PB-N2N4 and 3PB-N2N5 refer to the AM/staggered solver with double time incrementation: an initial time increment of $10^{-2}$ s until the critical increment, followed by $10^{-4}$ s or $10^{-5}$ s for all subsequent increments until complete fracture, respectively.

makes the BFGS approach a promising tool for advancing the practical application of phase-field models in simulating complex fracture patterns in hyperelastic materials.

The findings presented in this study have significant implications for the field of fracture mechanics in hyperelastic materials. The demonstrated efficiency of the BFGS algorithm paves the way for more widespread adoption of phase-field methods in studying complex fracture phenomena in soft materials. This is particularly relevant for scenarios involving multi-physics conditions, such as thermo-oxidation and photo-oxidation, where computational demands have typically been a limiting factor. Moreover, the enhanced computational efficiency offered by the BFGS algorithm opens up new avenues for investigating larger and more complex systems within reasonable timeframes. This advancement is particularly beneficial for studying composite materials and systems exhibiting various dissipation mechanisms, such as fatigue fracture.

## CRediT authorship contribution statement

**Aimane Najmeddine:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Maryam Shakiba:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Investigation, Funding acquisition.

## Declaration of competing interest

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

## Data availability

Data will be made available on request.

## Acknowledgments

The authors gratefully acknowledge the support from the National Science Foundation, United States under the award number CMMI-1914565, and the Air-Force Office of Scientific Research (AFOSR) Young Investigator Program (YIP) award #FA9550-20-1-0281. The authors also acknowledge Advanced Research Computing at Virginia Tech for providing computational resources and technical support that have contributed to the results reported within this paper. URL: https://arc.vt.edu/.

## Appendix A. F-bar method

This section provides details of the displacement residuals and tangents for 2-D *F-bar* plane-strain quadrilateral elements in the reference configuration according to de Souza Neto et al. [48].

First, the *F-bar* deformation gradient is defined as

$$\bar{\mathbf{F}} = \left[\begin{array}{cc|c} & & 0 \\ & \bar{\mathbf{F}}_{pe} & 0 \\ \hline 0 & 0 & 1 \end{array}\right] \quad \text{where} \quad \bar{\mathbf{F}}_{pe} = \left(\frac{\det \mathbf{F}_0}{\det \mathbf{F}}\right)^{1/2} \mathbf{F} \tag{27}$$

where $\mathbf{F}_0$ is the deformation gradient at the centroid of the element. The degraded first Piola–Kirchhoff stress then becomes

$$\mathbf{P}(\bar{\mathbf{F}}) = \left(\frac{\det \mathbf{F}_0}{\det \mathbf{F}}\right)^{-1/2} \hat{\mathbf{P}}(\bar{\mathbf{F}}) \tag{28}$$

where $\hat{\mathbf{P}}(\bar{\mathbf{F}})$ is the stress tensor in Eq. (10) evaluated using $\bar{\mathbf{F}}$ instead of $\mathbf{F}$. $\mathbf{P}(\bar{\mathbf{F}})$ can be written in Voigt notation as $\bar{\mathsf{S}} = \left(\frac{\det \mathbf{F}_0}{\det \mathbf{F}}\right)^{-1/2} [\hat{P}_{11}\ \hat{P}_{21}\ \hat{P}_{12}\ \hat{P}_{22}]^{\mathsf{T}}$. Hence, the matrix form of the displacement residual in Eq. (17) becomes

$$\mathbf{R}_{\mathbf{u}} = -\int_{\Omega^e} \mathsf{G}^{\mathsf{T}} \bar{\mathsf{S}} \mathrm{dV} + \int_{\Omega_0^e} \mathsf{N}^{u^T} \mathbf{b}_0 \mathrm{dV} + \int_{\Gamma_{0N}^e} \mathsf{N}^{u^T} \mathbf{t}_0 \mathrm{dA} \tag{29}$$

Subsequently, the matrix form for the displacement tangent is given by

$$\mathbf{K}_{\mathbf{uu}} = \int_{\Omega_0^e} \left[\mathsf{G}^{\mathsf{T}} \mathsf{A} \mathsf{G} + \left(\frac{\det \mathbf{F}_0}{\det \mathbf{F}}\right)^{-1/2} \mathsf{G}^{\mathsf{T}} (\mathsf{Q}_{R0} \mathsf{G}_0 - \mathsf{Q}_R \mathsf{G})\right] \mathrm{dV} \tag{30}$$

where $\mathsf{A}$, $\mathsf{Q}_R$, and $\mathsf{Q}_{R0}$ are the matrix forms of the fourth-order tensors

$$\begin{aligned} \mathbb{A} &= \frac{\partial \hat{\mathbf{P}}(\bar{\mathbf{F}})}{\partial \bar{\mathbf{F}}} \\ \mathbb{Q}_{R0} &= \frac{1}{2}\mathbb{A} : (\bar{\mathbf{F}} \otimes \mathbf{F}_0^{-\mathsf{T}}) - \frac{1}{2}\mathbf{P}(\bar{\mathbf{F}}) \otimes \mathbf{F}_0^{-\mathsf{T}} \\ \mathbb{Q}_R &= \frac{1}{2}\mathbb{A} : (\bar{\mathbf{F}} \otimes \mathbf{F}^{-\mathsf{T}}) - \frac{1}{2}\mathbf{P}(\bar{\mathbf{F}}) \otimes \mathbf{F}^{-\mathsf{T}} \end{aligned} \tag{31}$$

The fourth-order tangent modulus tensor $\mathbb{A}$ is given in index form as follows

$$\mathbb{A}_{ijkl} = \mu \delta_{ik} \delta_{jl} + [\mu - K_B \ln(\det \bar{\mathbf{F}})] \bar{F}_{li}^{-1} \bar{F}_{jk}^{-1} + K_B \bar{F}_{lk}^{-1} \bar{F}_{ji}^{-1} \tag{32}$$

from which the corresponding Voigt second-order tensor form can be derived using the transformation table in Remark 2.

## Appendix B. Abaqus element-level residual and tangent used in the BFGS uel subroutine: RHS and AMATRX

In ABAQUS, each UEL developed must contain two important outputs: the element-level residual denoted by RHS, and the element-level tangent denoted by AMATRX. For the BFGS UEL subroutine developed in this work, the following order of indices was used

- Residual (dimension $1 \times M$)
  $\mathsf{RHS} = \left[R_{u_1}^1 R_{u_2}^1 R_{u_1}^2 R_{u_2}^2 \dots R_{u_1}^M R_{u_2}^M R_d^1 R_d^2 \dots R_d^M\right]$

- Tangent (dimension $3M \times 3M$)

$$\text{AMATRX} = \begin{bmatrix} K^{11}_{u_1u_1} & K^{11}_{u_1u_2} & \dots & K^{1M}_{u_1u_1} & K^{1M}_{u_1u_2} & K^{11}_{u_1d} & \dots & K^{1M}_{u_1d} \\ K^{11}_{u_2u_1} & K^{11}_{u_2u_2} & \dots & K^{1M}_{u_2u_1} & K^{1M}_{u_2u_2} & K^{11}_{u_2d} & \dots & K^{1M}_{u_2d} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ K^{M1}_{u_1u_1} & K^{M1}_{u_1u_2} & \dots & K^{MM}_{u_1u_1} & K^{MM}_{u_1u_2} & K^{M1}_{u_1d} & \dots & K^{MM}_{u_1d} \\ K^{M1}_{u_2u_1} & K^{M1}_{u_2u_2} & \dots & K^{MM}_{u_2u_1} & K^{MM}_{u_2u_2} & K^{M1}_{u_2d} & \dots & K^{MM}_{u_2d} \\ K^{11}_{du_1} & K^{11}_{du_2} & \dots & K^{1M}_{du_1} & K^{1M}_{du_2} & K^{11}_{dd} & \dots & K^{1M}_{dd} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ K^{M1}_{du_1} & K^{M1}_{du_2} & \dots & K^{MM}_{du_1} & K^{MM}_{du_2} & K^{M1}_{dd} & \dots & K^{MM}_{dd} \end{bmatrix}$$

where the displacement components in RHS and AMATRX are given by Eqs. (29) and (30), respectively, whereas those corresponding to the phase-field are given by Eq. (19).

## Appendix C. Algorithm for solving coupled large deformation solid mechanics and phase-field problems

**Algorithm 1:** Algorithm for solving coupled large deformation solid mechanics and phase-field problems

**Input:** Material properties
**Output:** Converged solution for displacement $\mathbf{u}$ and phase-field $d$
Initialize variables and parameters; Choose solution method: Staggered or BFGS; **while** *not converged* **do**
  **if** *Staggered method* **then**
    **for** *each iteration* $k$ **do**
      Solve for displacement $\bar{\mathbf{u}}_k$ with fixed $\bar{\mathbf{d}}_{k-1}$:
        - Calculate residual: $\mathbf{R}_{\mathbf{u}} = -\int_{\Omega_0^e} \mathsf{G}^{\mathsf{T}}\mathsf{P}_k \mathrm{dV} + \int_{\Omega_0^e} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{b}_0 \mathrm{dV} + \int_{\Gamma_{0\,N}^e} \mathsf{N}^{u^{\mathsf{T}}}\mathbf{t}_0 \mathrm{dA}$ with $\mathsf{P}_k = \mathsf{P}(\bar{\mathbf{u}}_k, \bar{\mathbf{d}}_{k-1})$
        - Solve: $\mathbf{K}_{\mathbf{uu}}\Delta\bar{\mathbf{u}} = \mathbf{R}_{\mathbf{u}}$ with $\Delta\bar{\mathbf{u}} = \bar{\mathbf{u}}_k - \bar{\mathbf{u}}_{k-1}$

      Solve for phase-field $\bar{\mathbf{d}}_k$ with updated $\bar{\mathbf{u}}_k$:

        - Calculate residual: $\mathbf{R}_d = -\int_{\Omega_0^e} \left\{ \left[ \frac{G_c}{l_c}\mathbf{N}^d\bar{\mathbf{d}}_k^{\mathsf{T}} - 2(1-\kappa)(1-\mathbf{N}^d\bar{\mathbf{d}}_k^{\mathsf{T}})\mathcal{H}_k \right](\mathbf{N}^d)^{\mathsf{T}} + G_c l_c (\mathbf{B}^d)^{\mathsf{T}}\mathbf{B}^d\bar{\mathbf{d}}_k^{\mathsf{T}} \right\} \mathrm{dV}$
        - Solve: $\mathbf{K}_{dd}\Delta\bar{\mathbf{d}} = \mathbf{R}_d$

      Check convergence;
    **end**
  **else if** *BFGS method* **then**
    Compute initial approximation to the inverse of the Jacobian, $\tilde{\mathbf{K}}_0^{-1}$; Set initial $\mathbf{z}_0 = [\bar{\mathbf{u}}_0, \bar{\mathbf{d}}_0]^T$;
    **for** *each iteration* $k = 1, 2, \dots$ **do**
      Solve for the correction: $\mathbf{c} = -[\tilde{\mathbf{K}}_{k-1}]^{-1}\mathbf{g}(\mathbf{z}_{k-1})$, where $\mathbf{g} = [\mathbf{R}_u, \mathbf{R}_d]^T$;
      Update solution: $\mathbf{z}_k = \mathbf{z}_{k-1} + \mathbf{c}$;
      Compute the change in residual: $\delta\mathbf{g} = \mathbf{g}(\mathbf{z}_k) - \mathbf{g}(\mathbf{z}_{k-1})$;
      **if** *not converged* **then**
        Update the approximation of the inverse Jacobian using the BFGS formula:

$$[\tilde{\mathbf{K}}_k]^{-1} = \left(\mathbf{I} - \frac{\delta\mathbf{z}_k\delta\mathbf{g}_k^{\mathsf{T}}}{\delta\mathbf{z}_k^{\mathsf{T}}\delta\mathbf{g}_k}\right)[\tilde{\mathbf{K}}_{k-1}]^{-1}\left(\mathbf{I} - \frac{\delta\mathbf{z}_k\delta\mathbf{g}_k^{\mathsf{T}}}{\delta\mathbf{z}_k^{\mathsf{T}}\delta\mathbf{g}_k}\right)^{\mathsf{T}} + \frac{\delta\mathbf{z}_k\delta\mathbf{z}_k^{\mathsf{T}}}{\delta\mathbf{z}_k^{\mathsf{T}}\delta\mathbf{g}_k}$$

      **end**
      Check convergence;
    **end**
  **end**
**end**

## References


[1] Francfort GA, Marigo J-J. Revisiting brittle fracture as an energy minimization problem. J Mech Phys Solids 1998;46(8):1319–42.
[2] Miehe C, Welschinger F, Hofacker M. Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field FE implementations. Internat J Numer Methods Engrg 2010;83(10):1273–311.
[3] Ambati M, Gerasimov T, De Lorenzis L. A review on phase-field models of brittle fracture and a new fast hybrid formulation. Comput Mech 2015;55(2):383–405. http://dx.doi.org/10.1007/s00466-014-1109-y.

[4] Miehe C, Schaenzel L-M, Ulmer H. Phase field modeling of fracture in multi-physics problems. Part I. Balance of crack surface and failure criteria for brittle crack propagation in thermo-elastic solids. Comput Methods Appl Mech Engrg 2015;294:449–85.
[5] Chen W-X, Wu J-Y. Phase-field cohesive zone modeling of multi-physical fracture in solids and the open-source implementation in Comsol Multiphysics. Theor Appl Fract Mech 2022;117:103153.
[6] Najmeddine A, Shakiba M. Physics and chemistry-based phase-field constitutive framework for thermo-chemically aged elastomer. Int J Mech Sci 2023;108721.
[7] Li P, Li W, Li B, Yang S, Shen Y, Wang Q, Zhou K. A review on phase field models for fracture and fatigue. Eng Fract Mech 2023;289:109419.
[8] Lo Y-S, Borden MJ, Ravi-Chandar K, Landis CM. A phase-field model for fatigue crack growth. J Mech Phys Solids 2019;132:103684.
[9] Seiler M, Linse T, Hantschke P, Kästner M. An efficient phase-field model for fatigue fracture in ductile materials. Eng Fract Mech 2020;224:106807.
[10] Carrara P, Ambati M, Alessi R, De Lorenzis L. A framework to model the fatigue behavior of brittle materials based on a variational phase-field approach. Comput Methods Appl Mech Engrg 2020;361:112731.
[11] Miehe C, Schänzel L-M. Phase field modeling of fracture in rubbery polymers. Part I: Finite elasticity coupled with brittle failure. J Mech Phys Solids 2014;65:93–113.
[12] Raina A, Miehe C. A phase-field model for fracture in biological tissues. Biomech Model Mechanobiol 2016;15(3):479–96.
[13] Borden MJ, Hughes TJ, Landis CM, Anvari A, Lee IJ. A phase-field formulation for fracture in ductile materials: Finite deformation balance law derivation, plastic degradation, and stress triaxiality effects. Comput Methods Appl Mech Engrg 2016;312:130–66.
[14] Wu J, McAuliffe C, Waisman H, Deodatis G. Stochastic analysis of polymer composites rupture at large deformations modeled by a phase field method. Comput Methods Appl Mech Engrg 2016;312:596–634.
[15] Reinoso J, Paggi M, Linder C. Phase field modeling of brittle fracture for enhanced assumed strain shells at large deformations: formulation and finite element implementation. Comput Mech 2017;59(6):981–1001.
[16] Gültekin O, Dal H, Holzapfel GA. Numerical aspects of anisotropic failure in soft biological tissues favor energy-based criteria: A rate-dependent anisotropic crack phase-field model. Comput Methods Appl Mech Engrg 2018;331:23–52.
[17] Tang S, Zhang G, Guo TF, Guo X, Liu WK. Phase field modeling of fracture in nonlinearly elastic solids via energy decomposition. Comput Methods Appl Mech Engrg 2019;347:477–94.
[18] Najmeddine A, Gupta S, Moini R. Coupled large deformation phase-field and cohesive zone model for crack propagation in hard-soft multi-materials. 2024, Available at SSRN 4783033.
[19] Bourdin B, Francfort GA, Marigo J-J. Numerical experiments in revisited brittle fracture. J Mech Phys Solids 2000;48(4):797–826.
[20] Bourdin B, Francfort GA, Marigo J-J. The variational approach to fracture. J Elasticity 2008;91(1):5–148.
[21] Kristensen PK, Martínez-Pañeda E. Phase field fracture modelling using quasi-Newton methods and a new adaptive step scheme. Theor Appl Fract Mech 2020;107:102446.
[22] Wu J-Y, Huang Y. Comprehensive implementations of phase-field damage models in Abaqus. Theor Appl Fract Mech 2020;106:102440.
[23] Wu J-Y, Huang Y, Nguyen VP. On the BFGS monolithic algorithm for the unified phase field damage theory. Comput Methods Appl Mech Engrg 2020;360:112704.
[24] Mandal TK, Nguyen VP, Wu J-Y, Nguyen-Thanh C, de Vaucorbeil A. Fracture of thermo-elastic solids: Phase-field modeling and new results with an efficient monolithic solver. Comput Methods Appl Mech Engrg 2021;376:113648.
[25] Wu J-Y, Huang Y, Zhou H, Nguyen VP. Three-dimensional phase-field modeling of mode I+ II/III failure in solids. Comput Methods Appl Mech Engrg 2021;373:113537.
[26] Russ J, Slesarenko V, Rudykh S, Waisman H. Rupture of 3D-printed hyperelastic composites: Experiments and phase field fracture modeling. J Mech Phys Solids 2020;140:103941.
[27] Peng F, Huang W, Zhang Z-Q, Guo TF, Ma YE. Phase field simulation for fracture behavior of hyperelastic material at large deformation based on edge-based smoothed finite element method. Eng Fract Mech 2020;238:107233.
[28] Ye J-Y, Zhang L-W, Reddy J. Large strained fracture of nearly incompressible hyperelastic materials: Enhanced assumed strain methods and energy decomposition. J Mech Phys Solids 2020;139:103939.
[29] Li B, Bouklas N. A variational phase-field model for brittle fracture in polydisperse elastomer networks. Int J Solids Struct 2020;182:193–204.
[30] Swamynathan S, Jobst S, Keip M-A. An energetically consistent tension–compression split for phase-field models of fracture at large deformations. Mech Mater 2021;157:103802.
[31] Ciambella J, Lancioni G, Stortini N. An Ogden-like formulation incorporating phase-field fracture in elastomers: from brittle to pseudo-ductile failures. Philos Trans R Soc A 2022;380(2234):20210323.
[32] Zheng S, Huang R, Lin R, Liu Z. A phase field solution for modelling hyperelastic material and hydrogel fracture in ABAQUS. Eng Fract Mech 2022;108894.
[33] Xing C, Yu T, Sun Y, Wang Y. An adaptive phase-field model with variable-node elements for fracture of hyperelastic materials at large deformations. Eng Fract Mech 2023;281:109115.
[34] Mandal TK, Nguyen VP, Wu J-Y. A length scale insensitive anisotropic phase field fracture model for hyperelastic composites. Int J Mech Sci 2020;188:105941.
[35] Mandal TK, Gupta A, Nguyen VP, Chowdhury R, de Vaucorbeil A. A length scale insensitive phase field model for brittle fracture of hyperelastic solids. Eng Fract Mech 2020;236:107196.
[36] Yin S, Yang W, Kwon J, Wat A, Meyers MA, Ritchie RO. Hyperelastic phase-field fracture mechanics modeling of the toughening induced by Bouligand structures in natural materials. J Mech Phys Solids 2019;131:204–20.
[37] Tian F, Zeng J, Tang X, Xu T, Li L. A dynamic phase field model with no attenuation of wave speed for rapid fracture instability in hyperelastic materials. Int J Solids Struct 2020;202:685–98.
[38] Tian F, Zeng J, Zhang M, Li L. Mixed displacement–pressure-phase field framework for finite strain fracture of nearly incompressible hyperelastic materials. Comput Methods Appl Mech Engrg 2022;394:114933.
[39] Peng F, Huang W, Ma Y, Zhang Z-Q, Fu N. Fourth-order phase field model with spectral decomposition for simulating fracture in hyperelastic material. Fatigue Fract Eng Mater Struct 2021;44(9):2372–88.
[40] Ang I, Bouklas N, Li B. Stabilized formulation for phase-field fracture in nearly incompressible hyperelasticity. Internat J Numer Methods Engrg 2022.
[41] Zhang Z, Qiu Y, Hu Z, Ye H, Zhang H, Zheng Y. Explicit phase-field total Lagrangian material point method for the dynamic fracture of hyperelastic materials. Comput Methods Appl Mech Engrg 2022;398:115234.
[42] Tian F, Zhang M, Zeng J, Li B, Li L. Adaptive stabilized mixed formulation for phase field fracture modeling of nearly incompressible finite elasticity. Int J Mech Sci 2022;107753.
[43] Tian F, Tang X, Xu T, Li L. An adaptive edge-based smoothed finite element method (ES-FEM) for phase-field modeling of fractures at large deformations. Comput Methods Appl Mech Engrg 2020;372:113376.
[44] Valverde-González A, Reinoso J, Jha N, Merodio J, Paggi M. A phase field approach to fracture for hyperelastic and visco-hyperelastic materials applied to pre-stressed cylindrical structures. Mech Adv Mater Struct 2022;1–20.
[45] Abaqus AD. Dassault systemes. RI, USA: Providence; 2014.
[46] Borden MJ, Verhoosel CV, Scott MA, Hughes TJ, Landis CM. A phase-field description of dynamic brittle fracture. Comput Methods Appl Mech Engrg 2012;217:77–95.

[47] 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 Engrg 2010;199(45–48):2765–78.
[48] de Souza Neto E, Perić D, Dutko M, Owen D. Design of simple low order finite elements for large strain analysis of nearly incompressible solids. Int J Solids Struct 1996;33(20–22):3277–96.
[49] Molnár G, Gravouil A. 2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture. Finite Elem Anal Des 2017;130:27–38.
[50] Lee J, Lee S, Chester SA, Cho H. Finite element implementation of a gradient-damage theory for fracture in elastomeric materials. Int J Solids Struct 2023;279:112309.
[51] Liu Z, Reinoso J, Paggi M. Phase field modeling of brittle fracture in large-deformation solid shells with the efficient quasi-Newton solution and global–local approach. Comput Methods Appl Mech Engrg 2022;399:115410.
[52] Koutromanos Y. Nonlinear finite element analysis. In: Lecture notes, vol. 24060, Blacksburg, Va: Virginia Polytechnic Institute and State University, Department of Civil and Environmental Engineering; 2014.
[53] Koutromanos I. Fundamentals of finite element analysis: Linear finite element analysis. John Wiley & Sons; 2018.
[54] Wu J-Y, Chen W-X, Zhou H. A length scale insensitive phase-field model for fully coupled thermo-mechanical fracture in concrete at high temperatures. Int J Numer Anal Methods Geomech 2022;46(14):2725–53.
[55] Shakiba M, Najmeddine A. Physics-based constitutive equation for thermochemically aged elastomers based on crosslink density evolution. J Mech Mater Struct 2023;17(3):229–46.
[56] Najmeddine A, Xu Z, Liu G, Croft ZL, Liu G, Esker AR, Shakiba M. Physics and chemistry-based constitutive modeling of photo-oxidative aging in semi-crystalline polymers. Int J Solids Struct 2022;239:111427.
[57] Najmeddine A, Shakiba M. Impact of void morphology on the mechanical response of time-dependent heterogeneous media: A numerical investigation approach. J Mater Civ Eng 2020;32(7):04020191.
[58] Najmeddine A, Shakiba M. Micromechanical study of porosity effects on coupled moisture-mechanical responses of viscoelastic asphalt concrete. J Eng Mech 2021;147(9):04021059.
[59] Niazi S, Najmeddine A, Shakiba M. A coupled thermo-hydro-mechanical framework for simulating the failure response of asphalt concrete under freezing conditions. Cold Reg Sci & Technol 2023;104073.
[60] Niazi S, Najmeddine A, Shakiba M. The effects of curing process on the damage behavior of additively manufactured fiber-reinforced thermosetting composites. Applied Composite Materials 2023;30(4):1305–31.
[61] Loew PJ, Poh LH, Peters B, Beex LA. Accelerating fatigue simulations of a phase-field damage model for rubber. Comput Methods Appl Mech Engrg 2020;370:113247.