Deutsche Forschungsgemeinschaft
Priority Programme 1962

Optimal Control for Phase-Field Fracture: Algorithmic Concepts and Computations

Denis Khimin, Marc C. Steinbach, Thomas Wick

SPP 1962 | Non-smooth and Complementarity-based Distributed Parameter Systems: Simulation and Hierarchical Optimization

Preprint Number SPP1962-160

received on March 25, 2021

Edited by
SPP1962 at Weierstrass Institute for Applied Analysis and Stochastics (WIAS)
Leibniz Institute in the Forschungsverbund Berlin e.V.
Mohrenstraße 39, 10117 Berlin, Germany
E-Mail: spp1962@wias-berlin.de

World Wide Web: http://spp1962.wias-berlin.de/

Optimal control for phase-field fracture: algorithmic concepts and computations

Denis Khimin, Marc C. Steinbach, Thomas Wick

Abstract In this work, we present an algorithmic realization for computing optimal control problems with quasi-static phase-field fracture as a PDE constraint. The phase-field fracture problem is formulated in a quasi-monolithic approach resulting in a nonlinear forward problem. The optimization problem is formulated within a reduced approach, where the state variable is eliminated. To this end, a globalized reduced Newton algorithm is employed. Our algorithmic developments are substantiated with a numerical example.

1 Introduction

This work is devoted to algorithmic concepts of phase-field fracture based optimization. Overviews of current trends in phase-field fracture are provided in [3, 12] and the monograph [11]. The basic mathematical model of phase-field fracture opimization was developed and analyzed in [9, 10]. Based on known algorithms for PDE-constrained optimization (e.g., [2, 7, 5]), numerical computations were carried out in [6] using the open-source package DOpElib [4]. In this contribution, we summarize algorithmic concepts and substantiate our developments with a numerical example.

Khimin, D., Steinbach M. C., Wick T.,
Leibniz University Hannover, Institute of Applied Mathematics, Welfengarten 1, 30167 Hannover, Germany
e-mail: {khimin,mcs,thomas.wick}@ifam.uni-hannover.de

2 Problem statements

In this section, we first state the forward problem and then the optimization problem. The former is given by a regularized discrete time fracture problem which uses a penalty ansatz for the growth irreversibility constraint.

2.1 Phase-field fracture forward problem

To formulate the problem, we need to introduce some notation. We consider a bounded domain $\Omega \subset \mathbb{R}^2$. Its boundary is partitioned as $\partial\Omega = \Gamma_N \dot\cup \Gamma_D$ where Γ_D and Γ_N have nonzero Hausdorff measure. Next we define two function spaces, $V := H^1_D(\Omega;\mathbb{R}^2) \times H^1(\Omega)$ and $Q := L^2(\Gamma_N)$, where V denotes the space for the displacement-field and the phase-field and Q is the control space. In the next step we introduce a function $E^\gamma_\varepsilon(q;u,\varphi)$ from which we derive our forward problem. $E^\gamma_\varepsilon(q;u,\varphi)$ is defined as the sum of the regularized total energy of a crack plus a penalty term for our time dependent irreversibility constraint. The regularized total energy of a crack is given by

$$E_\varepsilon(q;u,\varphi) := \frac{1}{2}\left(g(\varphi)\mathbb{C}e(u), e(u)\right) - (q,u)_{\Gamma_N} + G_c\Gamma_\varepsilon(\varphi), \tag{1}$$

where q denotes a force that is applied in orthogonal direction to $\Gamma_N \subset \partial\Omega$, u is the vector-valued displacement field, φ is the phase-field, $\mathbb{C}$ is the elasticity tensor and $e(u)$ the symmetric gradient. The so-called degradation function $g(\varphi) := (1-\kappa)\varphi^2+\kappa$ helps to extend the displacements to the entire domain Ω. The term $G_c\Gamma_\varepsilon(\varphi) := \frac{1}{2\varepsilon}\|1-\varphi\| + \frac{\varepsilon}{2}\|\nabla\varphi\|^2$ is a regularized form of the Hausdorff measure [1].

So far the problem consists in finding a function $\mathbf{u}(t) = (u(t),\varphi(t))$ that minimizes the regularized total energy (1) subject to the irreversibility constraint $\varphi(t_2) \le \varphi(t_1)$ for $t_1 \le t_2$. In the sequel, the constraint is being replaced by a penalty term in the cost functional. To ensure differentiability up to second order [8] we define the penalty function as $R(\varphi^{i-1};\varphi^i) := \frac{1}{4}\|(\varphi^i-\varphi^{i-1})^+\|^4_{\mathcal{L}^4}$. Finally the regularized discrete forward problem consists in finding a vector

$$\mathbf{u} := (\mathbf{u}^i)_{i=1}^M = (u,\varphi) = (u^i,\varphi^i)_{i=1}^M = (u(t_i),\varphi(t_i))_{i=1}^M \in V^M$$

that solves the following optimization problem for given intial data $(u^0,\varphi^0) \in V$ and given control $q := (q^i)_{i=1}^M \in Q^M$:

$$\min_{\mathbf{u}} \; E^\gamma_\varepsilon(q^i,\varphi^{i-1};u^i,\varphi^i) := E_\varepsilon(q^i;u^i,\varphi^i) + \gamma R(\varphi^{i-1};\varphi^i), \tag{C$^\gamma$}$$

for $i = 1,\dots,M$ and $\gamma > 0$.

2.2 Optimization problem

We formulate the following regularized NLP. For given $(u_0, \varphi_0) \in V$ we seek a solution $(q, \mathbf{u}) \in (Q, V)^M$ to (NLP$^\gamma$):

$$\begin{aligned} \min_{q,\mathbf{u}} \quad & J(q,\mathbf{u}) = \frac{1}{2}\sum_{i=1}^{M}\|\varphi^i - \varphi_d^i\|^2 + \frac{\alpha}{2}\sum_{i=1}^{M}\|q^i\|_{\Gamma_N}^2 \\ \text{s.t.} \quad & (q^i, \mathbf{u}^i) \text{ solves (EL}^\gamma\text{) for each } i = 1, \dots, M, \end{aligned} \tag{NLP$^\gamma$}$$

where φ_d is some desired phase-field function. The existence of a global solution of (NLP$^\gamma$) has been shown in [9, Theorem 4.3] for functions that are non-negative and weakly semi-continuous. Here, $\mathbf{u}^i = (u^i, \varphi^i) \in V^M$ of (C$^\gamma$) solves the Euler-Lagrange equations for describing the phase-field fracture propagation, see e.g., [9],

$$\begin{aligned} \big(g(\varphi^i)\mathbb{C}e(u^i), e(v)\big) - \big(q^i, v\big)_{\Gamma_N} &= 0, \\ G_c\varepsilon\big(\nabla\varphi^i, \nabla\psi\big) - \frac{G_c}{\varepsilon}\big(1-\varphi^i, \psi\big) + (1-\kappa)\big(\varphi^i\mathbb{C}e(u^i) : e(u^i), \psi\big) & \\ + \gamma\big(((\varphi^i - \varphi^{i-1})^+)^3, \psi\big) &= 0, \end{aligned} \tag{EL$^\gamma$}$$

for every pair of test functions $(v, \psi) \in V$ and $i = 1, \dots, M$.

3 Reduced optimization problem

To formulate a reduced optimization problem, we assume the existence of a solution operator $S\colon Q^M \to V^M$ via equation (EL$^\gamma$). With this solution operator we can reduce the cost functional $J(q,u)$ to $j\colon Q^M \to \mathbb{R}$, $j(q) \coloneqq J(q, S(q))$. This yields the unconstrained reduced optimization problem

$$\min_q \; j(q), \quad q \in Q^M. \tag{NLP$^\gamma_{\text{red}}$}$$

In the next step we formulate the Lagrangian function for (NLP$^\gamma$) by means of a functional $a\colon (Q\times V)^M \to \mathbb{R}$ that combines the constraints (EL$^\gamma$). Given a Lagrange multiplier $\mathbf{z} = (\mathbf{z}^i)_{i=1}^M = (z, \varphi_z) = (z^i, \varphi_z^i)_{i=1}^M \in V^M$, that functional reads

$$\begin{aligned} a(q,\mathbf{u})(\mathbf{z}) \coloneqq \sum_{i=1}^{M}\Big[& \big(g(\varphi^i)\mathbb{C}e(u^i)), e(z^i)\big) - (q^i, z^i)_{|\Gamma_N} + G_c\varepsilon\big(\nabla\varphi^i, \nabla\varphi_z^i\big) \\ & - \frac{G_c}{\varepsilon}\big(1-\varphi^i, \varphi_z^i\big) + (1-\kappa)\big(\varphi^i\mathbb{C}e(u^i) : e(u^i), \varphi_z^i\big) + \gamma\big(((\varphi^i - \varphi^{i-1})^+)^3, \varphi_z^i\big)\Big]. \end{aligned} \tag{2}$$

The Lagrangian $\mathcal{L}\colon (Q \times V \times V)^M \to \mathbb{R}$ for (NLP$^\gamma$) is then defined as

$$\mathcal{L}(q, \mathbf{u}, \mathbf{z}) \coloneqq J(q,\mathbf{u}) - a(q,\mathbf{u})(\mathbf{z}). \tag{3}$$

We will use this Lagrangian to obtain computable representations of the first and second order derivatives of j. To find a respresentation for the first derivative $j'(q)(\tau q)$ in the direction τq, we have to solve two equations. First we need to find a solution $\boldsymbol{u} = S(q) \in V$ of the state equation $\mathcal{L}'_{\mathbf{z}}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\phi}) = 0 \Leftrightarrow a(q, \mathbf{u})(\boldsymbol{\phi}) = 0 \; \forall \boldsymbol{\phi} \in V^M$. Then we have to solve the adjoint equation $\mathcal{L}'_{\mathbf{u}}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\phi}) = 0$ for $\mathbf{z} \in V$. Finally we obtain the representation

$$j'(q)(\tau q) = \mathcal{L}'_q(q, \mathbf{u}, \mathbf{z})(\tau q). \tag{4}$$

For an analogous representation of $j''(q)(\delta q, \tau q)$ we first need to find a solution $\boldsymbol{\delta u} \in V$ of the tangent equation $\mathcal{L}''_{q\mathbf{z}}(q, \mathbf{u}, \mathbf{z})(\delta q, \boldsymbol{\phi}) + \mathcal{L}''_{\mathbf{uz}}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\delta u}, \boldsymbol{\phi}) = 0$. After that we solve the adjoint Hessian equation $\mathcal{L}''_{q\mathbf{u}}(q, \mathbf{u}, \mathbf{z})(\delta q, \boldsymbol{\phi}) + \mathcal{L}''_{\mathbf{uu}}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\delta u}, \boldsymbol{\phi}) + \mathcal{L}''_{\mathbf{zu}}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\delta} z, \boldsymbol{\phi}) = 0$ for $\boldsymbol{\delta z} \in V$. This results in the representation

$$\begin{aligned} j''(q)(\delta q, \tau q) = \mathcal{L}''_{qq}(q, \mathbf{u}, \mathbf{z})(\delta q, \tau q) + \mathcal{L}''_{\mathbf{u}q}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\delta u}, \tau q) \\ + \mathcal{L}''_{\mathbf{z}q}(q, \mathbf{u}, \mathbf{z})(\boldsymbol{\delta z}, \tau q). \end{aligned} \tag{5}$$

Additional theory for these representations can be found in [2, Theorem 2.1].

For the spatial discretization, we employ a classical Galerkin finite element method on a quadrilateral mesh. To this end, we use conforming finite element spaces $V_{h_1} \subset H^1_D(\Omega; \mathbb{R}^2)$, $V_{h_2} \subset H^1(\Omega)$, and Q_h for the control.

Now we are ready to formulate our discrete problem, where we seek a solution $(q_h, \boldsymbol{u}_h) \coloneqq (q_h(t_i), u_h(t_i))_{i=1}^M$ with $\varphi_h(0) = \varphi_{0,h}$ of the following NLP:

$$\begin{aligned} \min_{q_h, \mathbf{u}_h} \quad & J(q_h, \mathbf{u}_h) \\ \text{s.t.} \quad & \big(g(\varphi_h^i)\mathbb{C}e(u_h^i), e(v_h)\big) - \big(q_h^i, v_h\big)_{\Gamma_N} = 0 \, \forall (v_h, \psi_h) \in V_{h_1} \times V_{h_2}, i = 1, \dots, M, \\ & G_c \varepsilon\big(\nabla\varphi_h^i, \nabla\psi_h\big) - \frac{G_c}{\varepsilon}\big(1 - \varphi_h^i, \psi_h\big) + (1 - \kappa)\big(\varphi_h^i \mathbb{C}e(u_h^i) : e(u_h^i), \psi_h\big) \\ & + \gamma\big(((\varphi_h^i - \varphi_h^{i-1})^+)^3, \psi_h\big) = 0 \, \forall (v_h, \psi_h) \in V_{h_1} \times V_{h_2}, i = 1, \dots, M. \end{aligned}$$

The discrete reduced cost functional $j_h \colon Q_h^M \to \mathbb{R}$ is defined in terms of the discrete solution operator S_h as $j_h(q_h) \coloneqq J(q_h, S_h(q_h))$. This yields the (unconstrained) discrete reduced NLP

$$\min_{q_h} \; j_h(q_h), \quad q_h \in Q_h^M. \tag{6}$$

Next, we define a basis $\{\tau q_j \mid j = 1, \dots, n_q\}$ of our n_q-dimensional control space Q_h^M in order to solve (6) with Newton's method. In the unconstrained problems ($\text{NLP}^{\gamma}_{\text{red}}$) and (6) we seek an optimal control $q \in Q_h^M$ that satisfies the first order necessary optimality condition $j_h'(q)(\tau q) = 0 \, \forall \tau q \in Q_h^M$. The Newton increment δq of this condition is determined from the linear system

$$j_h''(q)(\delta q, \tau q) = -j_h'(q)(\tau q) \quad \forall \tau q \in Q_h^M. \tag{7}$$

Provided that $j_h''(q)$ is positive definite, (7) is also a necessary *and sufficient* condition for the strongly convex quadratic problem

$$\min_{\delta q} \; m(q,\delta q) \coloneqq j_h(q) + j_h'(q)(\delta q) + \frac{1}{2} j_h''(q)(\delta q, \delta q) \quad \forall \delta q \in Q_h^M. \tag{8}$$

Thus, any solution δq of (8) solves (7) and vice versa. Before we proceed with system (7), we define vectores and matrices that represent the derivatives of the discrete reduced cost functional j_h:

$$\begin{aligned} (\nabla j_h(q), \tau q)_{Q^M} &\coloneqq j_h'(q)(\tau q) \quad \forall \tau q \in Q_h^M, \\ \left(\nabla^2 j_h(q)\delta q, \tau q\right)_{Q^M} &\coloneqq j_h''(q)(\delta q, \tau q) \quad \forall \delta q, \tau q \in Q_h^M. \end{aligned}$$

With this notation (7) becomes:

$$\left(\nabla^2 j_h(q)\delta q, \tau q_j\right)_{Q^M} = -\left(\nabla j_h(q), \tau q_j\right)_{Q^M}, \quad j = 1, \dots, n_q. \tag{9}$$

We can reformulate (9) in vector notation by replacing $\nabla j_h(q) \in Q_h^M$ with the corresponding coefficient vector $\boldsymbol{f} \in \mathbb{R}^{n_q}$:

$$\left(\nabla j_h(q), \tau q_j\right)_{Q^M} = \sum_{i=1}^{n_q} \left(\boldsymbol{f}_i \tau q_i, \tau q_j\right)_{Q^M} = \sum_{i=1}^{n_q} \boldsymbol{f}_i (\tau q_i, \tau q_j)_{Q^M}.$$

Since that equation has to hold for every element τq_j, $\boldsymbol{f}$ can be computed as a solution of the linear system

$$\boldsymbol{G}\boldsymbol{f} = \left((\nabla j_h(q), \tau q_j)_{Q^M}\right)_{j=1}^{n_q}, \tag{10}$$

where $\boldsymbol{G}_{ji} \coloneqq \left(\tau q_i, \tau q_j\right)_{Q^M}$ denotes the Gramian matrix of the basis of Q_h^M. Having constructed the right hand side of (9), we do the same for the left hand side and replace δq by its coefficient vector $\boldsymbol{d}$:

$$\left(\nabla^2 j_h(q)\delta q, \tau q_j\right)_{Q^M} = \sum_{i=1}^{n_q} \boldsymbol{d}_i \left(\nabla^2 j_h(q)\tau q_i, \tau q_j\right)_{Q^M}.$$

Thus we compute $\boldsymbol{d}$ as solution of the linear system

$$\boldsymbol{K}\boldsymbol{d} = \left(\left(\nabla^2 j_h(q)\delta q, \tau q_j\right)_Q\right)_{j=1}^{n_q}, \tag{11}$$

where $\boldsymbol{K}_{ji} \coloneqq (\nabla^2 j_h(q)\tau q_i, \tau q_j)_{Q^M}$. Using (10) and (11), we finally replace (9) by the equivalent linear system $\boldsymbol{H}\boldsymbol{d} = -\boldsymbol{f}$ with coefficient matrix $\boldsymbol{H} \coloneqq \boldsymbol{G}^{-1}\boldsymbol{K}$. If the dimension n_q is large, constructing the entire matrix $\boldsymbol{H}$ becomes expensive. Instead we can just compute a coefficient vector for the product $\nabla^2 j_h(q)\delta q \in Q_h^M$ by solving $\left(\nabla^2 j_h(q)\delta q, \tau q_j\right)_{Q^M} = \sum_{i=1}^{n_q} \boldsymbol{h}_i \left(\tau q_i, \tau q_j\right)_{Q^M}$. To conclude this section,

we introduce the notation $\langle \boldsymbol{a}, \boldsymbol{b} \rangle := \boldsymbol{a}^T \boldsymbol{G} \boldsymbol{b}$ and $|\boldsymbol{a}| := \langle \boldsymbol{a}, \boldsymbol{a} \rangle^{\frac{1}{2}}$. Now $(\mathbb{R}^{n_q}, |\cdot|)$ is isometrically isomorphic to $(Q_h^M, \|\cdot\|_{Q^M})$, and the functional m from (8) can be written as:

$$m(q, \boldsymbol{d}) = j_h(q) + \langle \boldsymbol{f}, \boldsymbol{d} \rangle + \frac{1}{2} \langle \boldsymbol{H} \boldsymbol{d}, \boldsymbol{d} \rangle. \tag{12}$$

The conjugate gradient (CG) method can be used for minimizing m in step 4 of Algorithm (1) below.

4 Algorithmic realization

Algorithm (1) For a given control $q^0 = (q^0(t_i))_{i=1}^M \in Q_h^M$, iterate over $k = 0, 1, \ldots$

1. Calculate $\mathbf{u}^k = (\mathbf{u}^k(t_i))_{i=1}^M \in (V_{h_1} \times V_{h_2})^M$ from the discrete state equation: $\mathcal{L}'_{\mathbf{z}}(q^k, \mathbf{u}^k, \mathbf{z}^k)(\boldsymbol{\phi}) = 0 \Leftrightarrow a(q^k, \mathbf{u}^k)(\boldsymbol{\phi}) = 0$, for all $\boldsymbol{\phi} \in (V_{h_1} \times V_{h_2})^M$.
2. Calculate $\mathbf{z}^k = (\boldsymbol{z}^k(t_i))_{i=1}^M \in (V_{h_1} \times V_{h_2})^M$ from the discrete adjoint equation: $\mathcal{L}'_{\mathbf{u}}(q^k, \mathbf{u}^k, \mathbf{z}^k)(\boldsymbol{\phi}) = 0$ for all $\boldsymbol{\phi} \in (V_{h_1} \times V_{h_2})^M$.
3. Build the coefficient vector $\boldsymbol{f}$ of $\nabla j_h(q^k)$ by applying the representation (4) to $\tau q = \tau q_j$ for $j = 1, \ldots, n_q$: $(\nabla j_h(q^k))_j = \hat{\mathcal{L}}'_q(q^k, \mathbf{u}^k, \mathbf{z}^k)(\tau q_j)$. Then solve the following system:

$$\boldsymbol{G}\boldsymbol{f} = \left(j'_h(q^k)(\tau q_j)\right)_{j=1}^{n_q} \quad \text{where} \quad \boldsymbol{G}_{ij} = (\tau q_j, \tau q_i)_{Q^M}.$$

4. Calculate a solution $j''_h(q^k)(\delta q, \tau q_j) = -j'_h(q^k)(\tau q_j)$ with $\tau q_j \in Q_h^M$ via

$$\min_{\boldsymbol{d} \in \mathbb{R}^{n_q}} \; m(q^k, \boldsymbol{d}), \tag{13}$$

   using an iterative algorithm that only needs matrix-vector products with the Hessian. These products will be calculated in algorithm 2.
5. Choose a steplength ν_k by an Armijo backtracking method.
6. Set $q^{k+1} = q^k + \nu_k \delta q$.
7. Increase $k = k + 1$.

Repeat until $|\boldsymbol{f}| = \|\nabla j_h(q^k)\|_{Q^M} < \text{TOL}$.

Algorithm (2) Let $\boldsymbol{u}^k$ and $\boldsymbol{z}^k$ be given with respect to the current control q^k.

1. Calculate $\boldsymbol{\delta u}^k$ by solving the tangent equation.
2. Calculate $\boldsymbol{\delta z}^k$ by solving the adjoig Hessian equation.
3. Construct the coefficient vector for $\nabla^2 j_h(q)\delta q$ by using representation (5) with $\tau q = \tau q_j$ for $j = 1, \ldots, n_q$. Then solve the linear system

$$\boldsymbol{G}\boldsymbol{h} = \left(j''_h(q^k)(\delta q, \tau q_j)\right)_{j=1}^{n_q}.$$

Remark 1 A documentation and implementation of Newton's method for solving the inner nonlinear phase-field fracture problem can be found in [4].

5 Numerical example

We wish to find an optimal control that produces some given phase-field φ_d. Here we apply our force q in orthogonal direction to the top side $\Gamma_N \subset \partial\Omega$ of the domain $\Omega := (-1,1)^2$. Furthermore we split $\partial\Omega$ in three subsets, $\partial\Omega = \Gamma_N \cup \Gamma_D \cup \Gamma_{\text{free}}$, where $\Gamma_N := [-1,1]\times\{1\}$, $\Gamma_D := [-1,1]\times\{-1\}$, $\Gamma_{\text{free}} := \{\pm 1\}\times[-1,1]$. On Γ_N we apply our force q and on Γ_D we enforce Dirichlet boundary conditions $u = 0$.

In this example the control space Q_h is one-dimensional. The applied force q is constant in time, $q^i = q$, and the initial data is given by $\boldsymbol{u}^0 = (u(t_0), \varphi(t_0)) := (0; \varphi^0)$, where the inital phase-field φ^0 is defined as follows:

$$\varphi^0(x,y) := \begin{cases} 0, \; x \in (-0.1-h, 0.1+h) \text{ and } y \in (-h,h), \\ 1, \text{ else.} \end{cases} \tag{14}$$

This initial condition describes a horizontal crack between the points $(-0.1-h, 0)$ and $(0.1+h, 0)$ with a thickness of $2h$. Before we can solve (NLP$^\gamma$) we have to define some constants. The regularization parameter for the phase-field is $\varepsilon := 2h = 0.088$, where h denotes the diameter of the mesh elements. The parameter κ in the coefficient function $g(\varphi)$ is $\kappa := 10^{-10}$. The main regularization parameter α is 10^{-6}, and the penalty parameter γ is 10^8. The given phase-field φ_d continues the initial phase-field φ to the left side of the domain,

$$\varphi_d(x,y) := \begin{cases} 0, \; x \in (-1, 0.1+h) \text{ and } y \in (-h,h), \\ 1, \text{ else.} \end{cases} \tag{15}$$

Our results are presented in Table 1. Every row consists of 8 entries. The first value (Iter.) is the current Newton iteration, the second value (CG) is the number of CG iterations that were necessary for minimizing m. The remaining values are the relative and absolute Newton residuals (Rel. residual, Abs. residual), the cost functional J and its tracking part $\frac{1}{2}\sum_{i=1}^{5}\|\varphi(t_i) - \varphi_d(t_i)\|_2^2$ (Cost, Tracking), the maximal force $|q_{\max}|$ on Γ_N (Force), and the norm-square of the gradient of the reduced cost functional, $\|\nabla j(q)\|_2^2$ (Gradient). All values are rounded to three significant digits. Finally, solution plots are provided in Figure 1.

Table 1 Results of the numerical example.

Iter.	CG	Rel. residual	Abs. residual	Cost	Tracking	Force	Gradient
0	—	1.0	0.005 66	4.68	0.251	1000	1.00×10^{-6}
1	2	0.006 73	3.82×10^{-5}	0.165	0.165	26.2	4.15×10^{-11}
2	3	1.05×10^{-5}	5.93×10^{-8}	0.165	0.165	0.882	3.52×10^{-15}
3	18	7.36×10^{-9}	4.17×10^{-11}	0.165	0.165	0.000 215	1.74×10^{-21}
4	9	3.76×10^{-11}	2.13×10^{-13}	0.165	0.165	1.36×10^{-6}	4.52×10^{-26}

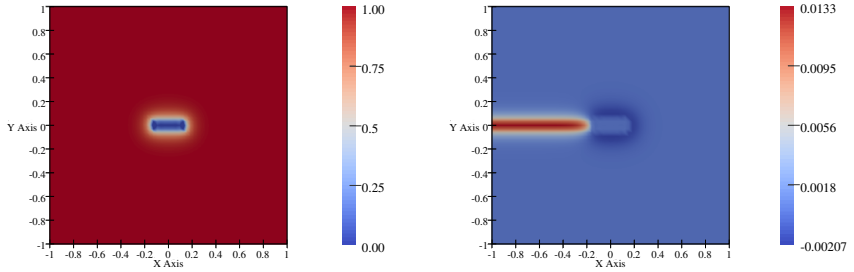

Fig. 1 Left: phase-field, right: adjoint phase-field.

Acknowledgements This work has been supported by the German Research Foundation, Priority Program 1962 (DFG SPP 1962) with in the subproject *Optimizing Fracture Propagation using a Phase-Field Approach* with the project number 314067056.

References

1. L. Ambrosio and V. Tortorelli. On the approximation of free discontinuity problems. *Boll. Un. Mat. Ital. B*, 6:105–123, 1992.
2. R. Becker, D. Meidner, and B. Vexler. Efficient numerical solution of parabolic optimization problems by finite element methods. *Optim. Methods Softw.*, 22(5):813–833, 2007.
3. B. Bourdin and G. A. Francfort. Past and present of variational fracture. *SIAM News*, 52(9), 2019.
4. C. Goll, T. Wick, and W. Wollner. DOpElib: Differential equations and optimization environment; A goal oriented software library for solving PDEs and optimization problems with PDEs. *Archive of Numerical Software*, 5(2):1–14, 2017.
5. M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*. Number 23 in Mathematical modelling: theory and applications. Springer, Dordrecht u.a., 2009.
6. D. Khimin. Numerische Modellierung von Optimalsteuerungsproblemen für Phasenfeld-Riss-Ausbreitung. Master's thesis, Leibniz University Hannover, Jun 2020.
7. D. Meidner. *Adaptive Space-Time Finite Element Methods for Optimization Problems Governed by Nonlinear Parabolic Systems*. PhD thesis, University of Heidelberg, 2008.
8. C. Meyer, A. Rademacher, and W. Wollner. Adaptive optimal control of the obstacle problem. *SIAM Journal on Scientific Computing*, 37(2):A918–A945, Jan. 2015.
9. I. Neitzel, T. Wick, and W. Wollner. An optimal control problem governed by a regularized phase-field fracture propagation model. *SIAM Journal on Control and Optimization*, 55(4):2271–2288, 2017.
10. I. Neitzel, T. Wick, and W. Wollner. An optimal control problem governed by a regularized phase-field fracture propagation model. Part II: The regularization limit. *SIAM Journal on Control and Optimization*, 57(3):1672–1690, 2019.
11. T. Wick. *Multiphysics Phase-Field Fracture: Modeling, Adaptive Discretizations, and Solvers*. De Gruyter, Berlin, Boston, 12 Oct. 2020.
12. J.-Y. Wu, V. P. Nguyen, C. Thanh Nguyen, D. Sutula, S. Bordas, and S. Sinaie. Phase field modelling of fracture. *Advances in Applied Mechanics*, 53, 09 2019.