

Topology optimization of structures undergoing brittle fracture

Jeet Desai, Grégoire Allaire, François Jouve

▶ **To cite this version:**

Jeet Desai, Grégoire Allaire, François Jouve. Topology optimization of structures undergoing brittle fracture. Journal of Computational Physics, 2022, 458, 10.1016/j.jcp.2022.111048 . hal-03277281

HAL Id: hal-03277281
https://hal.science/hal-03277281

Submitted on 2 Jul 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Topology optimization of structures undergoing brittle fracture

Jeet Desai [*1, 3], Grégoire Allaire [† 2], and François Jouve [‡ 1]

[1]Université de Paris and Sorbonne Université, CNRS, LJLL, F-75006 Paris, France
[2]CMAP, École Polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau, France
[3]IRT SystemX, Gif-sur-Yvette, France

July 2, 2021

Abstract

In the framework of the level-set method we propose a topology optimization algorithm for linear elastic structures which can exhibit fractures. In the spirit of Griffith theory, brittle fracture is modeled by the Francfort-Marigo energy model, with its Ambrosio-Tortorelli regularization, which can also be viewed as a gradient damage model. This quasi-static and irreversible gradient damage model is approximated using penalization to make it amenable to shape-differentiation. The shape derivative is determined using the adjoint method. The shape optimization is implemented numerically using a level-set method with body-fitted remeshing, which captures shapes exactly while allowing for topology changes. The efficiency of the proposed method is demonstrated numerically on 2D and 3D test cases. The method is shown to be efficient in conceiving crack-free structures.

1 Introduction

The integrity of a mechanical structure is of paramount importance to ensure safety. Structures subjected to a critical external load may undergo high stress, material damage, crack and an ultimate collapse. Shape and topology optimization to minimize damage can facilitate engineers to conceive robust structures, less susceptible to crack and hence with a longer life. The goal of the present work is to design a level-set based algorithm for optimizing structures which are not prone to fracture.

The first step is to choose a convenient fracture model. In the framework of brittle fracture mechanics, we consider the so-called Francfort-Marigo model [32], in the spirit of the pioneering energetic approach of Griffith [35]. This model relies on the quasi-static minimization of a total energy which is the sum of a linear elasticity energy and a fracture energy, under an irreversibility constraint. Its mathematical formulation was inspired by the Mumford-Shah energy functional for image segmentation [45], which is a simpler model with a scalar unknown and without time variable. The original Francfort-Marigo model belongs to the class of free discontinuity problems (the crack is an unknown of the problem), which are notoriously difficult to solve numerically. Therefore, the same authors propose to approximate their original model with a regularized damage model [16], following a Gamma-convergence process, first suggested by Ambrosio and Tortorelli for the Mumford-Shah functional [6]. This Ambrosio-Tortorelli regularization features a small regularizing parameter $\ell > 0$. In the case of the Mumford-Shah functional, by Gamma-convergence techniques it was proved in [6], [17] that, as ℓ converges to zero, the minimizers of the regularized model converge to those of the original free discontinuity problem. A similar convergence holds true for a time-discretized version of the Francfort-Marigo model [20, 34].

In the regularization of this free discontinuity problem, the parameter ℓ can be viewed as a purely mathematical artifact, used to approach the exact minimization problem. However, this

*jeet.desai@irt-systemx.fr
†gregoire.allaire@polytechnique.fr
‡francois.jouve@u-paris.fr

regularization has also a very clear physical interpretation as damage model with an internal length scale which is precisely the regularization parameter ℓ. Recall that a damage model features a damage variable α, measuring the extent of degradation of the material and varying continuously between 0 (no damage) and 1 (fully damage), and an elastic Hooke's tensor which is a decreasing function of α. Additional ingredients are, like in the fracture case, an irreversibility constraint and an energy criterion which determines the onset of damage. How the original fracture or free discontinuity problem is recast as a damage problem is precisely explained in [16] and [48]. This approach is also called sometimes a smeared interface approach or a phase-field approach. Its main advantage is its simplicity of numerical implementation using the finite element method. Furthermore, it can easily detect initiation, branching and coalescence of cracks without the need of meshing the crack path. In the sequel we shall indifferently call this model a damage or a fracture model.

The second step is to choose a shape and topology optimization method. There are mainly two classes of algorithms. The first class, that of so-called density methods, represents or approximates shapes by a continuous density function. The second class captures shapes by means of an auxiliary function which could be a characteristic function, a phase-field function or a level-set function. Here, we follow the level-set framework, introduced by Osher and Sethian [47]. The level-set approach was combined with the notion of shape derivative in [4, 51] to make it a successful method for shape and topology optimization of structures. We refer to [2] for a review of the level-set method in structural optimization and for further references. Combining the Francfort-Marigo fracture model with the level-set method, we propose a shape and topology optimization algorithm for preventing crack initiation and propagation in solid structures.

There has already been some works on this topic. In the level-set framework we are only aware of [52] which optimizes the configuration of composite materials in a phase-field based fracture model. We differ from [52] in many aspects: they do not use a level-set equation but rather a reaction-diffusion equation, they do not use a shape derivative but instead a topological gradient for a simplified model with a fixed damage field and finally they do not optimize the overall shape but just the inclusion's shape inside a composite structure. There are more works on topology optimization using SIMP (solid isotropic material with penalization) applied to various fracture models. Topology optimization using SIMP was performed for a fracture model in [40] or for a damage model to reinforce concrete in [8]. Path dependency of the damage model was taken into account in SIMP [38]. Phase-field model of fracture was considered for topology optimization using extended BESO (Bi-directional Evolutionary Structural Optimization) [26]. Fracture governed by the phase-field model is considered for maximizing the fracture resistance of periodic composite [26].

A common feature in all the previous works using SIMP is that several material properties must be approximated for the mixture of material and void, corresponding to the density variable. This is classical for the Young's modulus [14] but more delicate for other properties like, for example, the fracture toughness. Usually, these material properties are approximated by multiplying their values by the density raised to a certain exponent. This exponent is different for every property and ought to be chosen in an ad-hoc manner, ensuring numerical stability. If the optimized shape has intermediate densities, the interpretation of damage is quite artificial and may cause numerical difficulties. On the contrary, in the level-set framework, since the material properties are never approximated, such artificial damaged zones are avoided.

The content of this article is as follows. Section 2 is devoted to the presentation of the Francfort-Marigo damage model. Although this model has nice properties, it features an irreversibility constraint (a damage region cannot heal and be again undamaged) which makes it a variational inequality, instead of a more standard variational equality. Unfortunately, the adjoint method for computing sensitivities or derivatives of an objective function is extremely involved and not practical for variational inequalities since it involves the notion of conical derivative [44, 49]. Therefore, we penalize the irreversibility constraint to transform the variational inequality into a more convenient variational equality which is amenable to the adjoint method.

Section 3 states the optimization problem and delivers its shape derivative (see Proposition 1), relying on the adjoint method and the well-known Hadamard and Céa's methods. There is again a subtle point here. The state equation corresponds to the minimization of a total energy depending on two variables, the elastic displacement $\boldsymbol{u}$ and the damage variable α. It turns out that this energy is not convex with respect to $(\boldsymbol{u}, \alpha)$. As usual, the adjoint equation is the adjoint linearization of the optimality condition for this energy minimization. Therefore, the adjoint equation features a linear operator which is the transpose of the Hessian (or second-order derivative matrix) of this non-convex

energy. As such, it is not a coercive operator and it is not clear at all that the adjoint equation admits a solution. Nevertheless, under a technical assumption we are able to prove the existence and uniqueness of a solution of the time-discretized version of the adjoint problem (see Theorem 1).

Section 4 presents the discretization of the damage model and of the adjoint problem in space and time, and the level-set algorithm. A key ingredient for the sequel is that, the damage model being non-convex, it is solved with a so-called backtracking algorithm, due to [16], which is able to escape from local minima in the total energy minimization. Our finite element analyses are performed within the `FreeFEM` software [36]. For the finer 3D meshes we use the parallel computational capacities of `FreeFEM` which rely on domain decomposition and the `PETSc` package [10]. The level-set transport equation is solved with the `advect` library [18]. One original feature of our work is that we rely on body-fitted meshes at each iteration, using the remeshing software `MMG` [27]. Having a body-fitted mesh of the structure is crucial for an accurate evaluation of the damage. Indeed, there is no ersatz or weak material, so damage does not interact with this fictitious phase. Furthermore, since damage typically occurs in region of high stresses, those are more precisely computed with a body-fitted mesh. We also outline the details of our gradient-based algorithm for shape and topology optimization.

Finally, Section 5 is concerned with 2D and 3D numerical test cases. The objective function is the so-called total compliance (integrated in time), which is minimized under a volume constraint. These examples illustrate the efficiency of our proposed shape and topology optimization algorithm to obtain crack-free optimal structures. One specific difficulty in optimization of damage of fracture models is that these phenomena may be discontinuous: a small increase in the loadings, a small change in the structure's geometry may cause the sudden occurrence of a not-so-small crack and thus a large increase of the objective function (see the discussion in Remark 10). Nevertheless, our algorithm is able to sustain these large oscillations and, in the end, converge smoothly to optimal undamaged structures.

2 Fracture model

This section is devoted to a presentation of the Francfort-Marigo model [32] of brittle fracture which is relying on a mechanical energy minimization. Their original model was based on a representation of the fracture as a line (in 2D) or surface (in 3D) of displacement discontinuity but they proposed [16] a very efficient approximation using the Ambrosio-Tortorelli regularization, which can be viewed as a gradient damage model [48]. Over the last two decades, several researchers have worked on different aspects of this regularization, cited in the following presentation.

2.1 Governing laws and variational formulation

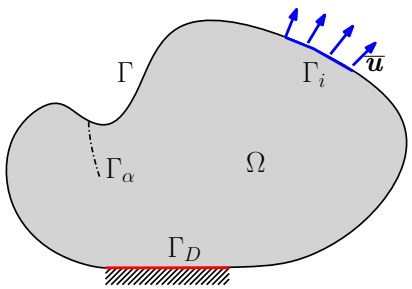


Figure 1: Boundary conditions and the unknown crack Γ_α

Let $\Omega \subset \mathbb{R}^d$ be a smooth bounded open set representing the structure in Fig.1 (with $d = 2, 3$ the space dimension). For $T > 0$, $[0, T]$ is the bounded time interval of interest. The imposed displacement is denoted by $\bar{\boldsymbol{u}} : \Omega \times [0, T] \mapsto \mathbb{R}^d$ and is assumed to be smooth, typically $\bar{\boldsymbol{u}} \in \mathcal{C}^0([0, T], H^2(\mathbb{R}^d)^d)$. Let $\partial\Omega = \Gamma_D \cup \Gamma \cup \Gamma_{\bar{\boldsymbol{u}}}$, be the union of disjoint boundaries and $\boldsymbol{n}$ denote the outward normal to $\partial\Omega$: Γ is the free boundary (no traction), Γ_D is the Dirichlet boundary and $\Gamma_{\bar{\boldsymbol{u}}}$ is the boundary where the displacement $\bar{\boldsymbol{u}}$ is imposed.

The elastic displacement vector field is $\boldsymbol{u} : \Omega \times [0,T] \mapsto \mathbb{R}^d$ and $\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right)$ denotes the second order strain tensor. The fourth order Hooke's tensor $\mathbb{C}_0$ is assumed to be coercive on the set $\mathcal{M}_s^d$ of symmetric $d \times d$ matrices, i.e., $\exists\ c_{\min} > 0,\ c_{\max} > 0$ such that, $\forall \boldsymbol{\xi} \in \mathcal{M}_s^d$,

$$c_{\min}|\boldsymbol{\xi}|^2 \leq \mathbb{C}_0\boldsymbol{\xi}\!:\!\boldsymbol{\xi} \leq c_{\max}|\boldsymbol{\xi}|^2.$$

The rate of evolution of the imposed displacement $\bar{\boldsymbol{u}}$ is assumed to be small, resulting in a quasi static evolution of the structure, and hence a negligible acceleration $\ddot{\boldsymbol{u}} = \boldsymbol{0}$. As the loading increases with time t and the elastic energy in the structure exceeds a critical elastic energy density, the structure undergoes damage, which is measured with the damage variable $\alpha : \Omega \times [0,T] \mapsto [0,1]$. The value $\alpha = 0$ corresponds to no damage and $\alpha = 1$ to a complete damage. Damage is characterized by deterioration of the stiffness and is modeled by assuming the Hooke's tensor $\mathbb{C}(\alpha)$ to be a convex function of the damage variable α, such that

$$\mathbb{C}(0) = \mathbb{C}_0, \quad \mathbb{C}(1) = \boldsymbol{0} \ \text{ and } \ \mathbb{C}'(\alpha)\boldsymbol{\xi}\!:\!\boldsymbol{\xi} < 0\ \forall \boldsymbol{\xi} \in \mathcal{M}_s^d. \tag{1}$$

The second expression in the above states that when the damage variable α attains unity, the stiffness becomes zero and structure undergoes a crack. The third expression in the above states that, when α increases, the stiffness must decrease.

The study of the damage model requires the introduction of some functional spaces. The space of admissible displacements is given by

$$V = \{\boldsymbol{u} \in H^1(\Omega)^d :\ \boldsymbol{u} = \boldsymbol{0} \ \text{ on } \Gamma_D\}. \tag{2}$$

For $v = \dot{\bar{\boldsymbol{u}}}$, the affine space of admissible velocities is defined as

$$C_v = \{\boldsymbol{v} \in V :\ \boldsymbol{v} = \boldsymbol{v},\ \text{on } \Gamma_{\bar{\boldsymbol{u}}}\}. \tag{3}$$

The subspace of functions in $H^1(\Omega)$ which vanish on Γ_D and $\Gamma_{\bar{\boldsymbol{u}}}$, is denoted by

$$H_D^1(\Omega) = \{\beta \in H^1(\Omega) :\ \beta = 0 \ \text{ on } \Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}\}. \tag{4}$$

The convex set of admissible damage is

$$\mathcal{D}_1 = \{\alpha \in H_D^1(\Omega) :\ 0 \leq \alpha(x) \leq 1 \ \ a.e.\ x \in \Omega\} \tag{5}$$

and the convex set of admissible damage evolution rate ($\gamma = \dot{\alpha}$) is

$$\mathcal{D} = \{\gamma \in H_D^1(\Omega) :\ \gamma(x) \geq 0 \ \ a.e.\ x \in \Omega\}, \tag{6}$$

To simplify notations, we define the product space

$$Z = V \times H_D^1(\Omega).$$

The initial condition of the model is

$$(\boldsymbol{u}(0), \alpha(0)) = (\boldsymbol{0}, \alpha_0) \in V \times \mathcal{D}_1. \tag{7}$$

Remark 1. *Typically, the set $\mathcal{D}_1$ of admissible damage variable and the set $\mathcal{D}$ of admissible damage evolution rate are defined for functions in $H^1(\Omega)$. In (5) and (6) we rather choose $H_D^1(\Omega)$ because we wish to forbid crack formation on Γ_D and $\Gamma_{\bar{\boldsymbol{u}}}$. Without this, the creation of cracks on these boundaries would require only half the energy needed for cracks occurring inside the structure Ω and thus would be artificial.*

Following [5, 48] we introduce the elastic energy and the damage energy respectively as

$$\mathcal{E}(\boldsymbol{u}, \alpha) = \frac{1}{2}\int_\Omega \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u})\!:\!\boldsymbol{\varepsilon}(\boldsymbol{u})\, dx, \tag{8}$$

$$\mathcal{H}(\alpha) = \frac{G_c}{4c_w}\left(\int_\Omega \ell|\nabla\alpha|^2 dx + \int_\Omega \frac{w(\alpha)}{\ell} dx\right), \quad \text{where} \quad c_w = \int_0^1 \sqrt{w(\xi)}d\xi, \tag{9}$$

where G_c is the fracture toughness of the material, $\ell > 0$ is the characteristic length, measuring the thickness of the damaged zone around the fracture, and $w(\alpha)$ is the fracture energy density [24], assumed to satisfy

$$w(0) = 0,\ w(1) = 1, \text{ and } w'(\alpha) \geq 0 \text{ for } 0 \leq \alpha \leq 1.$$

The sum of the integrands in the above two energies defines the energy density W_ℓ

$$W_\ell(\boldsymbol{u}, \alpha) = \frac{1}{2}\mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) + \frac{G_c}{4c_w}\left(\ell|\nabla\alpha|^2 + \frac{w(\alpha)}{\ell}\right). \tag{10}$$

The total energy of the structure is then given by

$$\mathcal{P}(\boldsymbol{u}, \alpha) = \int_\Omega W_\ell(\boldsymbol{u}, \alpha)\ dx. \tag{11}$$

The Francfort-Marigo regularized fracture model amounts to minimize the total energy $\mathcal{P}(\boldsymbol{u}, \alpha)$ among all fields $(\boldsymbol{u}(t), \alpha(t)) \in V \times \mathcal{D}_1$, for $t \in (0, T]$, such that $(\dot{\boldsymbol{u}}(t), \dot{\alpha}(t)) \in C_v \times \mathcal{D}$ and with the initial condition (7). In particular, the definition of the space $\mathcal{D}$ contains the irreversibility condition $\dot{\alpha} \geq 0$ and that of the space C_v implies that the boundary condition $\boldsymbol{u}(t) = \bar{\boldsymbol{u}}(t)$ on $\Gamma_{\bar{\boldsymbol{u}}}$ is satisfied at all times.

No body or surface forces are applied. Indeed, it is well-known [32] that, in the case of complete damage ($\mathbb{C}(1) = \mathbf{0}$), applying a force leads to the breakdown of the structure, which is not physical.

The solution $(\boldsymbol{u}(t), \alpha(t))$ satisfies the energy balance

$$\int_\Omega \dot{W}_\ell(\boldsymbol{u}, \alpha, \nabla\alpha)\ dx = \int_{\Gamma_{\bar{\boldsymbol{u}}}} (\mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u})\ \cdot \boldsymbol{n}) \cdot \dot{\bar{\boldsymbol{u}}}\, ds, \tag{12}$$

where overdot represents derivative with respect to time. The optimality conditions for the minimization of (11), written in a compact form, are [42, 46]: for all $t \in (0, T]$, find $(\boldsymbol{u}(t), \alpha(t)) \in V \times \mathcal{D}_1$, satisfying the initial condition (7), such that $(\dot{\boldsymbol{u}}(t), \dot{\alpha}(t)) \in C_v \times \mathcal{D}$, $\boldsymbol{u}(t) = \bar{\boldsymbol{u}}(t)$ on $\Gamma_{\bar{\boldsymbol{u}}}$ and

$$d\mathcal{P}(\boldsymbol{u}, \alpha)(\boldsymbol{v} - \dot{\boldsymbol{u}}, \beta - \dot{\alpha}) \geq 0 \qquad \forall (\boldsymbol{v}, \beta) \in C_v \times \mathcal{D}. \tag{13}$$

From (13), one can derive the strong form of the quasi-static damage model: find $(\boldsymbol{u}(t), \alpha(t))$, satisfying the initial condition (7), such that

$$\dot{\alpha} \geq 0 \qquad \text{in } \Omega \times (0, T], \tag{14a}$$

$$\frac{1}{2}\mathbb{C}'(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) - \frac{G_c}{2c_w}\ell\Delta\alpha + \frac{G_c}{4c_w}\frac{w'(\alpha)}{\ell} \geq 0 \qquad \text{in } \Omega \times (0, T], \tag{14b}$$

$$\dot{\alpha}\left(\frac{1}{2}\mathbb{C}'(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) - \frac{G_c}{2c_w}\ell\Delta\alpha + \frac{G_c}{4c_w}\frac{w'(\alpha)}{\ell}\right) = 0 \qquad \text{in } \Omega \times (0, T], \tag{14c}$$

$$\nabla\alpha \cdot \boldsymbol{n} \geq 0, \quad \dot{\alpha}\nabla\alpha \cdot \boldsymbol{n} = 0 \qquad \text{on } \partial\Omega \times (0, T], \tag{14d}$$

$$\alpha = 0 \qquad \text{on } (\Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}) \times (0, T], \tag{14e}$$

$$\operatorname{div}(\mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u})) = \mathbf{0} \qquad \text{in } \Omega \times (0, T], \tag{14f}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \qquad \text{on } \Gamma_{\bar{\boldsymbol{u}}} \times (0, T], \tag{14g}$$

$$\boldsymbol{u} = \mathbf{0} \qquad \text{on } \Gamma_D \times (0, T]. \tag{14h}$$

Inequality (14b) is known as the damage criterion. Equation (14c) is the complementary relation which essentially states that the damage criterion is an equality only if the damage evolution rate is positive. The variational formulation (or weak form) of the system of equations (14) reads: for all $t \in (0, T]$ find $(\boldsymbol{u}(t), \alpha(t)) \in V \times \mathcal{D}_1$ such that $\boldsymbol{u}(t) = \bar{\boldsymbol{u}}(t)$ on $\Gamma_{\bar{\boldsymbol{u}}}$, $\dot{\alpha} \in \mathcal{D}$

$$\int_\Omega \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\, dx = 0 \quad \forall \boldsymbol{v} \in V, \tag{15a}$$

$$\int_\Omega \frac{1}{2}\mathbb{C}'(\alpha)\beta\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u})\ dx + \int_\Omega \frac{G_c}{2c_w}\ell\nabla\alpha \cdot \nabla\beta\ dx + \int_\Omega \frac{G_c}{4c_w}\frac{w'(\alpha)}{\ell}\beta\ dx \geq 0 \qquad \forall \beta \in \mathcal{D}. \tag{15b}$$

When the characteristic length ℓ is small enough, and since $\mathbb{C}(1) = \mathbf{0}$, the above gradient damage model (15) is known to approximate brittle fracture. In Fig.1, Γ_α represents a crack which is a priori

unknown. The model (15) is able to capture a crack Γ_α which is typically a line in 2D and a surface in 3D, where α takes a unit value, which is surrounded by a band of thickness 2ℓ where α is strictly positive. Thus, the length ℓ characterizes the thickness of the region that shall undergo damage.

Two choices of dissipation function $w(\alpha)$ are usually considered and the second one is chosen for the numerical test cases in this article:

- `DQ` model (proposed in a different context by Ambrosio and Tortorelli [7])

$$w(\alpha) = \alpha^2, \qquad c_w = \frac{1}{2}. \tag{16}$$

  In this case, the damage onsets as soon as the external loading increments from zero.

- `DL` model (proposed in [48])

$$w(\alpha) = \alpha, \qquad c_w = \frac{2}{3}. \tag{17}$$

  In this case, the damage onsets only when a critical elastic energy density is exceeded.

The two choices of $w(\alpha)$, (17) and (16), make $w(\alpha)$ convex. Therefore, the functional $\alpha \mapsto \mathcal{W}_\ell(\boldsymbol{u}, \alpha)$ is convex. Similarly, for a fixed α, $\boldsymbol{u} \mapsto \mathcal{W}_\ell(\boldsymbol{u}, \alpha)$ is convex. This implies that fixing one variable at a time, one can alternately solve the minimization problem (15) [15]. However, $\mathcal{W}_\ell(\boldsymbol{u}, \alpha)$ is not convex with respect to the two variables $(\boldsymbol{u}, \alpha)$ together.

Remark 2. *The existence of at least one minimizer of* (11) *or, equivalently, of a solution to* (15) *in some weak sense was obtained in [34] for the so-called antiplanar shear case. Using Gamma-convergence, it can be proved that, as the length ℓ tends to zero, the global minimizers of* (11) *tend to a global minimizer of the original Francfort-Marigo energy (with a free discontinuity modeling fracture) [20, 34].*

2.2 Penalization

The damage criterion (15b) is a variational inequality and is not shape-differentiable in the classical sense. For a class of inequalities, called of the first kind and of the second kind, one can determine the so-called conical derivative [44, 49]. Well-known examples of inequality of the first kind include the obstacle-problem and the frictionless contact mechanics problem. The damage inequality (15b) without the irreversibility constraint (only the box constraint $\alpha \in [0, 1]$) and for a convex $\mathbb{C}(\alpha)$ classifies as an inequality of the second kind, hence easy to analyze. But with the irreversibility constraint, inequality (15b) classifies neither as the first kind nor as the second kind. The analytical treatment of (15b) is thus complex and out of the scope of this article. Instead, we prefer to convert the inequality (15b) into an equation, using penalization. Let ϵ be a penalization factor such that $0 < \epsilon \ll 1$ and let $\mathcal{M}$ be the max function, defined as

$$\mathcal{M}(\beta) = \max(\beta, 0).$$

From now on, we replace the original model (15) by the following penalized problem: for all $t \in (0, T]$, find $(\boldsymbol{u}_\epsilon(t), \alpha_\epsilon(t)) \in Z$ such that $\alpha_\epsilon(0) = \alpha_0$, $\boldsymbol{u}_\epsilon(t) = \bar{\boldsymbol{u}}(t)$ on $\Gamma_{\bar{\boldsymbol{u}}}$, and

$$\int_\Omega \frac{G_c}{2c_w}\left(\ell \nabla\alpha_\epsilon \cdot \nabla\beta + \frac{w'(\alpha_\epsilon)\beta}{2\ell}\right) dx + \int_\Omega \frac{1}{2}\mathbb{C}'(\alpha_\epsilon)\beta\boldsymbol{\varepsilon}(\boldsymbol{u}_\epsilon)\colon\boldsymbol{\varepsilon}(\boldsymbol{u}_\epsilon)\, dx + \int_\Omega \frac{G_c}{\epsilon}\left(\mathcal{M}(\alpha_\epsilon - 1) - \mathcal{M}(-\dot{\alpha}_\epsilon)\right)\beta\, dx = 0 \quad \forall \beta \in H^1_D(\Omega), \tag{18a}$$

$$\int_\Omega \mathbb{C}(\alpha_\epsilon)\boldsymbol{\varepsilon}(\boldsymbol{u}_\epsilon)\colon\boldsymbol{\varepsilon}(\boldsymbol{v})\, dx = 0 \qquad \forall \boldsymbol{v} \in V. \tag{18b}$$

We call (18), the state problem, and the solution to it $(\boldsymbol{u}_\epsilon, \alpha_\epsilon)$, the state solution. Note that we implicitly assume in writing (18a) that α_ϵ admits a time derivative. In the sequel it is assumed that there exists a unique state solution. As is clear in (18a), only the bound constraints $\dot{\alpha}_\epsilon \geq 0$ and $\alpha_\epsilon \leq 1$ are penalized. The constraint $\alpha_\epsilon \geq 0$ is not penalized explicitly, rather implicitly by penalizing $\dot{\alpha}_\epsilon \geq 0$ and defining an initial condition $\alpha_0 \geq 0$. A similar penalization approach was

studied numerically [43], where authors penalize only the irreversibility criterion. Our work defers as we penalize the upper bound $\alpha_\epsilon \leq 1$ as well.

In the sequel, to simplify notations, we shall drop all ϵ indices and simply denote by $(\boldsymbol{u}, \alpha)$ the solution $(\boldsymbol{u}_\epsilon, \alpha_\epsilon)$ of problem (18).

Remark 3. *The conversion of* (15) *to* (18) *has been made using the* max *function, but one can also consider a regularized form of the* max *function. This conversion not only simplifies the computation of the shape-derivative, but also helps in the numerical resolution using alternate minimizations (precisely, step 2 of the algorithm 1 using a Newton scheme). Without penalization, the damage model* (15) *can be solved with a sequential quadratic programming solver, capable of taking simple bound constraints into account [15].*

3 Optimization problem

3.1 Setting of the problem

We minimize an objective functional $J(\Omega)$ given by

$$J(\Omega) = \int_0^T \int_\Omega m(\boldsymbol{u}(\Omega), \alpha(\Omega))\, dx\, dt, \tag{19}$$

where $(\boldsymbol{u}(\Omega), \alpha(\Omega))$ is the solution of (18) and the function $m(\cdot, \cdot)$ is assumed to be $\mathcal{C}^1$ smooth with quadratic growth and linear growth for its derivative, so as to ensure that the objective function (19) is well-defined and the adjoint equation is well-posed. This objective functional represents a mechanical property such as total power, total elastic energy, or total fracture energy. In Section 5, we shall maximize the total elastic energy and thus choose

$$m(\boldsymbol{u}, \alpha) = -\mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}).$$

The justification for the above choice shall be given at the beginning of Section 5.

In practice, the shape Ω must be found inside a pre-fixed design space $D \subset \mathbb{R}^d$. Figure 2 shows

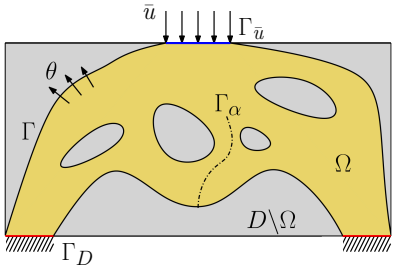


Figure 2: Design domain D and shape Ω

the shape Ω (in gold) and the design space D (in gold and grey). The crack Γ_α is an unknown of the problem (18) that shall be determined for every shape Ω. This crack Γ_α might appear anywhere in the shape Ω and there is no postulated initial crack. The space of admissible shapes $\mathcal{U}_{ad}$ is defined as

$$\mathcal{U}_{ad} = \left\{ \Omega \subset D : \int_\Omega dx = V_f \right\},$$

where Ω is an open set and V_f is the target volume. The minimization problem then reads

$$\min_{\Omega \in \mathcal{U}_{ad}} J(\Omega). \tag{20}$$

As is well known, very often there exists an optimal shape only if additional uniform smoothness conditions are imposed to the admissible shapes, that we shall not consider in the sequel. The study of existence of optimal shapes is outside the scope of this article (see [37] for more details) and we content ourselves with computing numerical minimizers, using a gradient-descent method.

3.2 Hadamard method

In the context of shape optimization, the derivative needed for a gradient descent method is based on the notion of Hadamard shape derivative [37, 49]. Given a smooth set Ω, its perturbation is expressed as

$$\Omega_{\boldsymbol{\theta}} = (I_d + \boldsymbol{\theta})(\Omega),$$

where $\boldsymbol{\theta} \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ and I_d is the identity map. It is well-known that when the norm of $\boldsymbol{\theta}$ is sufficiently small, the map $I_d + \boldsymbol{\theta}$ is a diffeomorphism in $\mathbb{R}^d$. Using this small perturbation one can define the Fréchet derivative for a function $J(\Omega)$.

Definition 1. *The shape derivative of $J(\Omega)$ at Ω is defined as the Fréchet derivative in $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ evaluated at $\mathbf{0}$ for the mapping $\boldsymbol{\theta} \mapsto J((I_d + \boldsymbol{\theta})\Omega)$ i.e.,*

$$J((I_d + \boldsymbol{\theta})\Omega) = J(\Omega) + J'(\Omega)(\boldsymbol{\theta}) + o(\boldsymbol{\theta}) \qquad \text{with } \lim_{\boldsymbol{\theta}\to 0} \frac{o(\boldsymbol{\theta})}{\|\boldsymbol{\theta}\|} = 0,$$

where $J'(\Omega)(\cdot)$ is a continuous linear form on $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$.

Given a shape Ω, one can then determine the above gradient $J'(\Omega)$, and move the shape along the gradient in an iterative manner, attempting to minimize the objective functional (19).

In a structural design process, typically, the clamped and the forced boundaries are assumed to be non-optimizable. Hence in our optimization, we allow only the free boundary Γ to move along $\boldsymbol{\theta}$ as shown in Fig.2. Furthermore, for simplicity it is assumed that $\Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}} \subset \partial D$. This constraint is incorporated by introducing the space of admissible perturbations

$$W_0^{1,\infty}(D, \mathbb{R}^d) = \{\boldsymbol{\theta} \in W^{1,\infty}(D, \mathbb{R}^d) : \ \boldsymbol{\theta} \cdot \boldsymbol{n} = \mathbf{0} \text{ on } \Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}\}.$$

Using this space, we state a lemma, useful to prove Theorem 1.

Lemma 1. *Let Ω be a smooth bounded open set and $\varphi \in W^{1,1}(\mathbb{R}^d, \mathbb{R})$. Define the function $J(\Omega)$ by*

$$J(\Omega) = \int_\Omega \varphi(x)\ dx.$$

Then $J(\Omega)$ is shape-differentiable at Ω and its derivative is

$$J'(\Omega)(\boldsymbol{\theta}) = \int_\Gamma \boldsymbol{\theta} \cdot \boldsymbol{n}\ \varphi(\boldsymbol{x})\ ds \qquad \forall \boldsymbol{\theta} \in W_0^{1,\infty}(D, \mathbb{R}^d).$$

3.3 Shape derivative computation

To define the adjoint problem for the shape derivative, we introduce the subspace V_0 of V

$$V_0 = \{\boldsymbol{u} \in H^1(\Omega)^d : \ \boldsymbol{u} = \mathbf{0} \ \text{ on } \Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}\}. \tag{21}$$

The adjoint variational formulation is defined as: find $(\boldsymbol{v}, \beta) \in H^1([0,T], Z)$, satisfying the final condition $\beta(T) = 0$, such that, for all $t \in [0,T)$,

$$\int_\Omega (\partial_{\boldsymbol{u}} m(\boldsymbol{u}, \alpha)\boldsymbol{\psi} + \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) + \mathbb{C}'(\alpha)\beta\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}))\, dx = 0 \qquad \forall \boldsymbol{\psi} \in V_0, \tag{22a}$$

$$\begin{aligned}\int_\Omega \Bigg(&\partial_\alpha m(\boldsymbol{u}, \alpha)\varphi + \mathbb{C}'(\alpha)\varphi\, \boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) + \frac{G_c \ell}{2c_w} \nabla\beta \cdot \nabla\varphi \\ &+ \left(\frac{1}{2}\mathbb{C}''(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) + \frac{G_c}{4c_w}\frac{w''(\alpha)}{\ell} + \frac{G_c}{\epsilon}\mathcal{M}'(\alpha - 1) \right) \beta\varphi \Bigg)\, dx \\ &- \int_\Omega \frac{G_c}{\epsilon} \left(\frac{d}{dt}(\mathcal{M}'(-\dot{\alpha})\beta) \right) \varphi\, dx = 0 \qquad \forall \varphi \in H_D^1(\Omega). \end{aligned} \tag{22b}$$

In the sequel, we assume that there exists a unique solution of the adjoint equation. Note that (22b) features the time derivative of the derivative $\mathcal{M}'$ of the maximum function $\mathcal{M}(\beta) = \max(\beta, 0)$.

Since $\mathcal{M}'$ is the Heaviside function, its time derivative is a Dirac mass and the precise meaning of the last integral in (22b) is unclear. However, if $\mathcal{M}(\beta)$ was a regularization of $\max(\beta, 0)$, then everything makes sense, including the following proposition. In other words, our computation of the shape derivative below is fine for a smooth function $\mathcal{M}(\beta)$ and merely formal for the maximum function. Further time discretization will make clear in which sense the time derivative of the Heaviside function is computed (see Subsection 3.4).

Proposition 1. *Let Ω be a smooth bounded open set. Assume that there exists a unique solution $(\boldsymbol{u}, \alpha)$ to (18), which belongs to $H^1([0,T], Z)$, and that there exists a unique solution $(\boldsymbol{v}, \beta) \in H^1([0,T], Z)$ of the adjoint equation (22). If the state solution $(\boldsymbol{u}, \alpha) \equiv (\boldsymbol{u}(\Omega), \alpha(\Omega))$ is shape-differentiable, then the objective function (19) admits a shape derivative, given, for any $\boldsymbol{\theta} \in W_0^{1,\infty}(D, \mathbb{R}^d)$, by*

$$J'(\Omega)(\boldsymbol{\theta}) = \int_0^T \int_\Gamma \boldsymbol{\theta} \cdot \boldsymbol{n} \left(m(\boldsymbol{u}, \alpha) + \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) + \frac{1}{2}\mathbb{C}'(\alpha)\beta\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) \right.$$
$$\left. + \frac{G_c}{2c_w}\left(\ell \nabla\alpha \cdot \nabla\beta + \frac{w'(\alpha)\beta}{2\ell} \right) + \frac{G_c}{\epsilon}\Big(\mathcal{M}(\alpha - 1) - \mathcal{M}(-\dot{\alpha}) \Big)\beta \right) ds\ dt. \tag{23}$$

Remark 4. *The uniqueness of the solution of the damage model (18) is far from being obvious since this model is the optimality condition for the minimization of a non-convex energy, which thus may have multiple minima. For the same reason, the existence of a solution for the adjoint equation (22) is not obvious either because the corresponding operator is not coercive (nevertheless, see Subsection 3.4 for a positive result in this direction). Note that we also assume that the solutions are smooth with respect to time since they belong to $H^1([0,T], Z)$ and, in particular, are continuous with respect to time. We use this assumption in the variational formulations (18a) and (22b), which involve the time derivative of the damage variable α. Unfortunately, as discussed further in Remark 10, it is likely that, in some cases, the solution $(\boldsymbol{u}(\Omega), \alpha(\Omega))$ is discontinuous in time and thus the shape derivative (23) is not rigorously justified.*

Proof. The idea of the proof is well-known, based on Céa's method [25]. Introduce a few spaces on the full space $\mathbb{R}^d$ and thus independent of Ω:

$$\tilde{V} = \{\boldsymbol{v} \in H^1(\mathbb{R}^d)^d : \boldsymbol{v} = \boldsymbol{0} \ \text{ on } \Gamma_D\}, \quad \tilde{\mathcal{C}}_t = \{\boldsymbol{v} \in \tilde{V} : \boldsymbol{v} = \bar{\boldsymbol{u}}(t) \ \text{ on } \Gamma_{\bar{\boldsymbol{u}}}\},$$
$$\tilde{H}^1_D(\mathbb{R}^d) = \{\beta \in H^1(\mathbb{R}^d) : \ \beta = 0 \ \text{ on } \Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}\}, \quad \tilde{Z}_t = \tilde{\mathcal{C}}_t \times \tilde{H}^1_D(\mathbb{R}^d),$$
$$\tilde{V}_0 = \{\boldsymbol{v} \in \tilde{V} : \boldsymbol{v} = \boldsymbol{0} \ \text{ on } \Gamma_{\bar{\boldsymbol{u}}}\}, \quad \tilde{Z}_0 = \tilde{V}_0 \times \tilde{H}^1_D(\mathbb{R}^d).$$

For independent variables $\tilde{\boldsymbol{u}}(t), \tilde{\boldsymbol{v}}(t), \tilde{\alpha}(t), \tilde{\beta}(t)$ and $\tilde{\lambda}$, belonging to the spaces

- $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) \in H^1([0,T], \tilde{Z}_t)$,
- $(\tilde{\boldsymbol{v}}, \tilde{\beta}) \in H^1([0,T], \tilde{Z}_0)$ (the Lagrange multiplier for the state equation (18)),
- $\tilde{\lambda} \in L^2(\mathbb{R}^d)$ (the Lagrange multiplier for the initial condition $\tilde{\alpha}(0) = \alpha_0$),

define a Lagrangian as

$$\mathcal{L}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}, \tilde{\alpha}, \tilde{\beta}, \tilde{\lambda}, \Omega) = \int_0^T \int_\Omega m(\tilde{\boldsymbol{u}}, \tilde{\alpha}) dx dt + \int_0^T \left(\int_\Omega \mathbb{C}(\tilde{\alpha})\boldsymbol{\varepsilon}(\tilde{\boldsymbol{u}}) : \boldsymbol{\varepsilon}(\tilde{\boldsymbol{v}}) \ dx \right) dt$$
$$+ \int_0^T \int_\Omega \left(\frac{1}{2}\mathbb{C}'(\tilde{\alpha})\tilde{\beta}\boldsymbol{\varepsilon}(\tilde{\boldsymbol{u}}) : \boldsymbol{\varepsilon}(\tilde{\boldsymbol{u}}) + \frac{G_c}{2c_w}\left(\ell \nabla\tilde{\alpha} \cdot \nabla\tilde{\beta} + \frac{w'(\tilde{\alpha})\tilde{\beta}}{2\ell} \right) + \frac{G_c}{\epsilon}\Big(\mathcal{M}(\tilde{\alpha} - 1) \right.$$
$$\left. - \mathcal{M}(-\dot{\tilde{\alpha}}) \Big)\tilde{\beta} \right) dx\ dt + \int_\Omega \tilde{\lambda}(\tilde{\alpha}(0) - \alpha_0)\ dx. \tag{24}$$

Since the boundaries $\Gamma_D \cup \Gamma_{\bar{\boldsymbol{u}}}$ are non-optimizable, the variables $\tilde{\boldsymbol{u}}(t), \tilde{\boldsymbol{v}}(t), \tilde{\alpha}(t), \tilde{\beta}(t)$, and $\tilde{\lambda}$ are independent of Ω. When the optimality condition are applied to the Lagrangian (24) (that is, its partial derivatives with respect to its independent variables are set to zero), we obtain the state equation (18), the adjoint equation (22) and the shape derivative (23).

At first, differentiating the Lagrangian (24) with respect to the adjoint variable $(\tilde{\boldsymbol{v}}, \tilde{\beta})$ in the direction $(\boldsymbol{\psi}, \varphi) \in H^1([0,T], \tilde{Z}_0)$ and equating it to zero at $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) = (\boldsymbol{u}, \alpha)$, we obtain

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \tilde{\beta}}(\varphi) = \int_0^T \left(\int_\Omega \frac{G_c}{2c_w} \left(\ell \nabla\alpha \cdot \nabla\varphi + \frac{w'(\alpha)\varphi}{2\ell} \right) dx + \int_\Omega \frac{1}{2} \mathbb{C}'(\alpha)\varphi \boldsymbol{\varepsilon}(\boldsymbol{u}) \colon \boldsymbol{\varepsilon}(\boldsymbol{u}) \; dx \right. \\ \left. + \int_\Omega \frac{G_c}{\epsilon} \left(\mathcal{M}(\alpha - 1) - \mathcal{M}(-\dot{\alpha}) \right) \varphi \, dx \right) dt = 0 \qquad \forall \varphi \in H^1([0,T], \tilde{H}^1_D(\mathbb{R}^d))) \, ,\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{\boldsymbol{v}}}(\boldsymbol{\psi}) = \int_0^T \left(\int_\Omega \mathbb{C}(\alpha) \boldsymbol{\varepsilon}(\boldsymbol{u}) \colon \boldsymbol{\varepsilon}(\boldsymbol{v}) \; dx \right) dt = 0 \qquad \forall \boldsymbol{\psi} \in H^1([0,T], \tilde{V}_0).$$

To obtain the initial condition $\alpha(0) = \alpha_0$, it suffices to differentiate (24) with respect to $\tilde{\lambda}$ at $\tilde{\alpha} = \alpha$. We thus recover the state equation (18).

Second, we differentiate the Lagrangian (24) with respect to $(\tilde{\boldsymbol{u}}, \tilde{\alpha})$ to recover the adjoint equation. By definition, $\tilde{\boldsymbol{u}} \in \tilde{\mathcal{C}}_t$, which is an affine space. The admissible perturbations $\boldsymbol{\psi}$ with respect to $\tilde{\boldsymbol{u}}$, must be such that $\tilde{\boldsymbol{u}} + \boldsymbol{\psi} \in \tilde{\mathcal{C}}_t$, hence $\boldsymbol{\psi} \in \tilde{V}_0$. Equating to zero the partial derivative in the direction $(\boldsymbol{\psi}, \varphi) \in H^1([0,T], \tilde{Z}_0)$, and denoting by $(\boldsymbol{v}, \beta)$ its solution for $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) = (\boldsymbol{u}, \alpha)$ and $\tilde{\lambda} = \lambda$, we arrive at

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \boldsymbol{u}}(\boldsymbol{\psi}) = \int_0^T \left(\int_\Omega \left(\partial_{\boldsymbol{u}} m(\boldsymbol{u}, \alpha)\boldsymbol{\psi} + \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) \right. \right. \\ \left. \left. + \mathbb{C}'(\alpha)\beta\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) \right) dx \right) dt = 0 \qquad \forall \boldsymbol{\psi} \in H^1([0,T], \tilde{V}_0),\end{aligned} \tag{26a}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \alpha}(\varphi) = \int_0^T \left(\int_\Omega \left(\partial_\alpha m(\boldsymbol{u}, \alpha)\varphi + \mathbb{C}'(\alpha)\varphi\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) + \frac{G_c \ell}{2c_w} \nabla\beta \cdot \nabla\varphi \right. \right. \\ \left. + \left(\frac{1}{2}\mathbb{C}''(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) + \frac{G_c}{4c_w}\frac{w''(\alpha)}{\ell} + \frac{G_c}{\epsilon}\mathcal{M}'(\alpha - 1) \right) \beta\varphi + \frac{G_c}{\epsilon}\mathcal{M}'(-\dot{\alpha})\dot{\varphi}\beta \left. \right) dx \right) dt \\ + \int_\Omega \lambda\varphi(0) dx = 0 \qquad \forall \varphi \in H^1([0,T], \tilde{H}^1_D(\mathbb{R}^d)).\end{aligned} \tag{26b}$$

Varying the test function $\boldsymbol{\psi}$ in (26a), we get the boundary condition $\boldsymbol{v}(t) = \boldsymbol{0}$ on $\Gamma_{\tilde{\boldsymbol{u}}}$ and for all $t \in [0,T)$

$$\int_\Omega \left(\partial_{\boldsymbol{u}} m(\boldsymbol{u}, \alpha)\boldsymbol{\psi} + \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) + \mathbb{C}'(\alpha)\beta\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) \right) dx = 0 \qquad \forall \boldsymbol{\psi} \in V_0,$$

where we used definition (21) of V_0. We have thus derived the adjoint equation (22a). Now, to get rid of the time derivative $\dot{\varphi}$ in equation (26b), we integrate the term $\mathcal{M}'(-\dot{\alpha})\dot{\varphi}\beta$ by parts with respect to t and obtain

$$\begin{aligned}&\int_0^T \left(\int_\Omega \left(\partial_\alpha m(\boldsymbol{u}, \alpha)\varphi + \mathbb{C}'(\alpha)\varphi\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) + \frac{G_c \ell}{2c_w} \nabla\beta \cdot \nabla\varphi \right. \right. \\ &\left. + \left(\frac{1}{2}\mathbb{C}''(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) + \frac{G_c}{4c_w}\frac{w''(\alpha)}{\ell} + \frac{G_c}{\epsilon}\mathcal{M}'(\alpha - 1) \right) \beta\varphi \right) dx \\ &- \int_\Omega \frac{G_c}{\epsilon} \left(\frac{d}{dt}(\mathcal{M}'(-\dot{\alpha})\beta) \right) \varphi \; dx \left. \right) dt + \int_\Omega \lambda \, \varphi|_{t=0} \, dx \\ &+ \int_\Omega \frac{G_c}{\epsilon} \left(\mathcal{M}'(-\dot{\alpha})\beta\varphi|_{t=T} - \mathcal{M}'(-\dot{\alpha})\beta\varphi|_{t=0} \right) dx = 0 \qquad \forall \varphi \in H^1([0,T], \tilde{H}^1_D(\mathbb{R}^d)).\end{aligned} \tag{27}$$

This integration by part is legitimate if $\mathcal{M}$ is a smooth function but is purely formal if $\mathcal{M}$ is the maximum function since in such a case $\mathcal{M}'$ is a Heaviside function and its time derivative involves a Dirac function. Varying φ in (27), we find that the Lagrange multiplier λ is given by

$$\lambda = \frac{G_c}{\epsilon} \, \mathcal{M}'(-\dot{\alpha})\beta|_{t=0} \, ,$$

and that the adjoint problem (22b) for β holds true. Finally, since $J(\Omega) = \mathcal{L}(\boldsymbol{u}, \alpha, \tilde{\boldsymbol{v}}, \tilde{\beta}, \tilde{\lambda}, \tilde{\boldsymbol{\mu}}, \Omega)$, the shape derivative $J'(\Omega)(\boldsymbol{\theta})$ satisfies, for any $\boldsymbol{\theta} \in W_0^{1,\infty}(D, \mathbb{R}^d)$,

$$J'(\Omega)(\boldsymbol{\theta}) = \frac{\partial \mathcal{L}}{\partial \Omega}(\boldsymbol{\theta}) + \frac{\partial \mathcal{L}}{\partial \boldsymbol{u}}\left(\frac{\partial \boldsymbol{u}}{\partial \Omega}(\boldsymbol{\theta})\right) + \frac{\partial \mathcal{L}}{\partial \alpha}\left(\frac{\partial \alpha}{\partial \Omega}(\boldsymbol{\theta})\right)$$

as the variables $\tilde{\boldsymbol{v}}, \tilde{\beta}, \tilde{\lambda}$ and $\tilde{\boldsymbol{\mu}}$ are independent of Ω. Substituting these variables by the optimal ones $\boldsymbol{v}, \beta, \lambda$, $\boldsymbol{\mu}$ and using the adjoint equation (22), the two last terms in the above formula vanish, resulting in

$$J'(\Omega)(\boldsymbol{\theta}) = \frac{\partial \mathcal{L}}{\partial \Omega}(\boldsymbol{\theta}).$$

Consequently, formula (23) is deduced by a straightforward application of Lemma 1. □

3.4 Time-discretized state and adjoint equations

The adjoint equation (22) is a linear backward parabolic equation with a final condition at $t = T$. This equation was assumed to be well-posed in the statement of Theorem 1. There are two difficulties in proving that (22) admits a unique solution. First, the bilinear form, involved in (22), is the Hessian of the non-convex energy functional (10) and thus is not coercive. Second, if $\mathcal{M}$ is the maximum function, then (22) features a time derivative which is a Dirac function (the precise meaning of which is unclear). This second issue can be settled upon time discretization, as can be expected. It turns out that, upon a technical assumption (see (34) in Theorem 1), the first issue can also be circumvented by a trick similar to what is used for solving Helmholtz equation (this equation is not coercive but its kernel is at most finite dimensional and often reduced to zero).

To construct a time-discretized version of the adjoint problem (22), we first time-discretize the state equation (18) along with the objective function (19). The time interval $[0, T]$ is split in N sub-intervals of length $\delta t = T/N$. Let $(\boldsymbol{u}_n, \alpha_n)$ and $(\boldsymbol{v}_n, \beta_n)$ denote the discrete state and adjoint solutions, respectively, at the end of every n-th time interval. The discrete state is determined using an implicit scheme: initialize $(\boldsymbol{u}_0, \alpha_0) = (\boldsymbol{0}, \alpha_0)$ and, for $0 < n \leq N$, find $(\boldsymbol{u}_n, \alpha_n) \in Z$ such that $\boldsymbol{u}_n = \bar{\boldsymbol{u}}(t_n)$ on $\Gamma_{\bar{\boldsymbol{u}}}$ and

$$\int_\Omega \mathbb{C}(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{v})\ dx = 0\ ds \qquad \forall \boldsymbol{v} \in V, \tag{28a}$$

$$\begin{aligned}\int_\Omega \frac{G_c}{2c_w}\left(\ell \nabla \alpha_n \cdot \nabla \beta + \frac{w'(\alpha_n)}{2\ell}\right)\ dx + \int_\Omega \frac{1}{2}\mathbb{C}'(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n)\ dx \\ + \int_\Omega \frac{G_c}{\epsilon}\left(\mathcal{M}(\alpha_n - 1)\beta\ dx - \int_\Omega \mathcal{M}\left(\frac{\alpha_{n-1} - \alpha_n}{\delta t}\right)\beta\right) dx = 0 \qquad \forall \beta \in H_D^1(\Omega).\end{aligned} \tag{28b}$$

The above problem can be shown to admit at least one solution [29], while uniqueness of the solution is not guaranteed (like its continuous counterpart (18)). Nevertheless, we assume that the solution $(\boldsymbol{u}_n, \alpha_n) \in Z$ is unique. The discretization of the objective function (19) reads:

$$J_N(\Omega) = \sum_{n=0}^{N} \delta t \int_\Omega m(\boldsymbol{u}_n, \alpha_n)\ dx. \tag{29}$$

Introducing a Lagrangian, as in the proof of Proposition 1, adapted to the above discretization, we obtain the following discrete adjoint problem: initialize $(\boldsymbol{v}_N, \beta_N) = (\boldsymbol{0}, 0)$ and, for $N - 1 \geq n \geq 0$, find $(\boldsymbol{v}_n, \beta_n) \in Z$ such that

$$\int_\Omega (\partial_{\boldsymbol{u}_n} m(\boldsymbol{u}_n, \alpha_n)\boldsymbol{\psi} + \mathbb{C}(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{v}_n) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}) + \mathbb{C}'(\alpha_n)\beta_n \boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{\psi}))\, dx = 0 \qquad \forall \boldsymbol{\psi} \in V_0, \tag{30a}$$

$$\begin{aligned}&\int_\Omega \frac{G_c \ell}{2c_w}\nabla\varphi \cdot \nabla\beta_n\, dx + \int_\Omega \left(\mathbb{C}'(\alpha_n)\varphi\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{v}_n) + \frac{1}{2}\mathbb{C}''(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n) + \frac{G_c}{4c_w}\frac{w''(\alpha_n)}{\ell}\right)\varphi\beta_n\, dx \\ &+ \int_\Omega \frac{G_c}{\epsilon}\left(\mathcal{M}'(\alpha_n - 1) + \frac{1}{\delta t}\mathcal{M}'(\alpha_{n-1} - \alpha_n)\right)\varphi\beta_n\ dx \\ &+ \int_\Omega \partial_{\alpha_n} m(\boldsymbol{u}_n, \alpha_n)\varphi\, dx = \frac{G_c}{\epsilon\delta t}\int_\Omega \mathcal{M}'(\alpha_n - \alpha_{n+1})\beta_{n+1}\varphi\ dx \quad \forall \varphi \in H_D^1(\Omega).\end{aligned} \tag{30b}$$

Remark 5. *When $\mathcal{M}$ is the maximum function, its derivative is the Heaviside function, $\mathcal{M}' = \mathcal{H}$. The value $\mathcal{H}(0)$ is not precisely defined since the Heaviside function is discontinuous at zero. Numerically, we tested the adjoint equation* (31) *(along with the corresponding shape derivative) for values $\mathcal{H}(0) = 0$ and $\mathcal{H}(0) = 1$, and both choices yield the same optimized shape (at least, for the 2D cantilever in Section 5).*

Since the variational formulation (30) is linear, it can be written in a compact form: $(\boldsymbol{v}_N, \beta_N) = (\mathbf{0}, 0)$ and, for $N - 1 \geq n \geq 0$, find $(\boldsymbol{v}_n, \beta_n) \in Z$ such that

$$a_n(\boldsymbol{v}_n, \beta_n, \boldsymbol{\psi}, \varphi) = f_n(\boldsymbol{\psi}, \varphi) \qquad \forall (\boldsymbol{\psi}, \varphi) \in Z, \tag{31}$$

where the symmetric bilinear form $a_n : Z \times Z \mapsto \mathbb{R}$ is defined as

$$\begin{aligned} a_n(\boldsymbol{v}, \beta, \boldsymbol{\psi}, \varphi) = &\int_\Omega \mathbb{C}(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{\psi}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\ dx + \int_\Omega \mathbb{C}'(\alpha_n)\beta\boldsymbol{\varepsilon}(\boldsymbol{\psi}) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n)\ dx + \int_\Omega \frac{G_c \ell}{2c_w} \nabla\varphi \cdot \nabla\beta\ dx \\ &+ \int_\Omega \mathbb{C}'(\alpha_n)\varphi\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n)\, dx + \int_\Omega \left(\frac{1}{2}\mathbb{C}''(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n) + \frac{G_c}{4c_w}\frac{w''(\alpha_n)}{\ell} \right) \varphi\beta\, dx \\ &+ \frac{G_c}{\epsilon} \int_\Omega \left(\mathcal{M}'(\alpha_n - 1) + \frac{1}{\delta t}\mathcal{M}'(\alpha_{n-1} - \alpha_n) \right) \varphi\beta\, dx, \end{aligned} \tag{32}$$

and the linear form $f_n : Z \mapsto \mathbb{R}$ is

$$f_n(\boldsymbol{\psi}, \varphi) = -\int_\Omega \partial_{\boldsymbol{u}} m(\boldsymbol{u}_n, \alpha_n)\boldsymbol{\psi}\, dx - \int_\Omega \partial_\alpha m(\boldsymbol{u}_n, \alpha_n)\varphi\, dx + \frac{G_c}{\epsilon\delta t}\int_\Omega \mathcal{M}'(\alpha_n - \alpha_{n+1})\beta_{n+1}\varphi\, dx. \tag{33}$$

To the bilinear form a_n is associated an operator $A_n : Z \mapsto Z$, defined by

$$a_n(\boldsymbol{v}, \beta, \boldsymbol{\psi}, \varphi) = \langle A_n(\boldsymbol{v}, \beta), (\boldsymbol{\psi}, \varphi) \rangle. \tag{34}$$

To prove the existence of a solution to the time-discretized adjoint equation (31), we have to change the assumption (1) on $\mathbb{C}(\alpha)$, which cannot be anymore degenerate when damage is complete.

Lemma 2. *Assume $\mathbb{C}(\alpha)$ is convex, decreasing, $\mathbb{C}(0) = \mathbb{C}_0$ and there exists $\kappa > 0$ (a residual stiffness) such that*

$$\mathbb{C}(\alpha)\boldsymbol{\xi} : \boldsymbol{\xi} \geq \kappa|\boldsymbol{\xi}|^2 \qquad \forall \boldsymbol{\xi} \in \mathcal{M}_s^d,\ 0 \leq \alpha \leq 1. \tag{35}$$

Assume that $\boldsymbol{u}_n \in W^{1,\infty}(\Omega)^d$. There exists a real number $c > 0$ such that $(A_n + c\mathrm{I})$ is invertible from Z to Z (where I is the identity operator) and its inverse is a compact linear continuous operator.

Proof. It is enough to check the coercivity of the bilinear form

$$a_n(\boldsymbol{v}, \beta, \boldsymbol{\psi}, \varphi) + c\langle (\boldsymbol{v}, \beta), (\boldsymbol{\psi}, \varphi) \rangle,$$

for some constant c, large enough. Since $\mathbb{C}(\alpha)$ is assumed to be convex and, by its definition (17) or (16), $w(\alpha)$ is convex too, we have

$$\mathbb{C}''(\alpha) \geq 0, \quad w''(\alpha) \geq 0. \tag{36}$$

Compute

$$
\begin{aligned}
a_n(\boldsymbol{\psi},\varphi,\boldsymbol{\psi},\varphi) &= \int_\Omega \mathbb{C}(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{\psi}):\boldsymbol{\varepsilon}(\boldsymbol{\psi})\ dx + \int_\Omega 2\mathbb{C}'(\alpha_n)\varphi\boldsymbol{\varepsilon}(\boldsymbol{\psi}):\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\ dx \\
&+ \int_\Omega \frac{G_c\ell}{2c_w}\nabla\varphi\cdot\nabla\varphi\ dx + \int_\Omega \left(\frac{1}{2}\mathbb{C}''(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n):\boldsymbol{\varepsilon}(\boldsymbol{u}_n) + \frac{G_c}{4c_w}\frac{w''(\alpha_n)}{\ell}\right)\varphi^2 dx \\
&+ \frac{G_c}{\epsilon}\int_\Omega \left(\mathcal{M}'(\alpha_n-1) + \frac{1}{\delta t}\mathcal{M}'(\alpha_{n-1}-\alpha_n)\right)\varphi^2\ dx \\
&\geq \kappa\int_\Omega |\boldsymbol{\varepsilon}(\boldsymbol{\psi})|^2 dx + \int_\Omega 2\mathbb{C}'(\alpha_n)\varphi\boldsymbol{\varepsilon}(\boldsymbol{\psi}):\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\ dx \\
&\quad + \int_\Omega \frac{G_c\ell}{2c_w}|\nabla\varphi|^2 dx + \int_\Omega \left(\frac{1}{2}\mathbb{C}''(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n):\boldsymbol{\varepsilon}(\boldsymbol{u}_n) + \frac{G_c}{4c_w}\frac{w''(\alpha_n)}{\ell}\right)\varphi^2 dx \qquad (\text{since } \mathcal{M}' = \mathcal{H} \geq 0\ ) \\
&\geq \kappa\int_\Omega |\boldsymbol{\varepsilon}(\boldsymbol{\psi})|^2 dx + \int_\Omega 2\mathbb{C}'(\alpha_n)\varphi\boldsymbol{\varepsilon}(\boldsymbol{\psi}):\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\ dx + \int_\Omega \frac{G_c\ell}{2c_w}|\nabla\varphi|^2 dx \qquad (\text{using (36) }) \\
&\geq \kappa\int_\Omega |\boldsymbol{\varepsilon}(\boldsymbol{\psi})|^2\ dx - 2\left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}\int_\Omega |\boldsymbol{\varepsilon}(\boldsymbol{\psi})|\ |\varphi\boldsymbol{\varepsilon}(\boldsymbol{u}_n)| dx + \int_\Omega \frac{G_c\ell}{2c_w}|\nabla\varphi|^2 dx \\
&\geq \kappa\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} - \left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}\left(s\left\|\varphi|\boldsymbol{\varepsilon}(\boldsymbol{u}_n)|\right\|^2_{L^2(\Omega)} + \frac{1}{s}\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)}\right) + \frac{G_c\ell}{2c_w}\left\|\nabla\varphi\right\|^2_{L^2(\Omega)} \\
&\qquad (\text{using Young's inequality with } s>0) \\
&\geq \kappa\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} - \left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}\left(s\left\|\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\right\|_{L^\infty(\Omega)}\left\|\varphi\right\|^2_{L^2(\Omega)} + \frac{1}{s}\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)}\right) + \frac{G_c\ell}{2c_w}\left\|\nabla\varphi\right\|^2_{L^2(\Omega)} \\
&\qquad (\text{because of our assumption } \boldsymbol{u}_n \in W^{1,\infty}(\Omega)^d\ \ ) \\
&= \left(\kappa - \frac{1}{s}\left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}\right)\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} + \frac{G_c\ell}{2c_w}\left\|\nabla\varphi\right\|^2_{L^2(\Omega)} - s\left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}\left\|\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\right\|_{L^\infty(\Omega)}\left\|\varphi\right\|^2_{L^2(\Omega)}.
\end{aligned}
$$

We choose $s = 2\left\|\mathbb{C}'(\alpha_n)\right\|_{L^\infty(\Omega)}/\kappa$ resulting in the bound

$$
\begin{aligned}
a_n(\boldsymbol{\psi},\varphi,\boldsymbol{\psi},\varphi) &\geq \frac{\kappa}{2}\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} + \frac{G_c\ell}{2c_w}\left\|\nabla\varphi\right\|^2_{L^2(\Omega)} - \frac{2}{\kappa}\left\|\mathbb{C}'(\alpha_n)\right\|^2_{L^\infty(\Omega)}\left\|\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\right\|_{L^\infty(\Omega)}\left\|\varphi\right\|^2_{L^2(\Omega)} \\
&\geq C_1\left(\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} + \left\|\nabla\varphi\right\|^2_{L^2(\Omega)}\right) - C_2\left\|\varphi\right\|^2_{L^2(\Omega)} \\
&\geq C_1\left(\left\|\boldsymbol{\varepsilon}(\boldsymbol{\psi})\right\|^2_{L^2(\Omega)} + \left\|\nabla\varphi\right\|^2_{L^2(\Omega)}\right) - C_2\left(\left\|\boldsymbol{\psi}\right\|^2_{L^2(\Omega)^d} + \left\|\varphi\right\|^2_{L^2(\Omega)}\right),
\end{aligned}
$$

where $C_1 = \min\left(\frac{\kappa}{2}, \frac{G_c\ell}{2c_w}\right)$ and $C_2 = \frac{2}{\kappa}\left\|\mathbb{C}'(\alpha_n)\right\|^2_{L^\infty(\Omega)}\left\|\boldsymbol{\varepsilon}(\boldsymbol{u}_n)\right\|_{L^\infty(\Omega)}$. Therefore, choosing $c > C_2$ yields the result. □

Lemma 2 implies that the operator A_n has a discrete countably infinite spectrum, like any elliptic operator, although it is not coercive (like the Helmholtz equation). It allows us to state a result about the well-posedness of the time-discretized adjoint equation (31).

Theorem 1. *Under the hypotheses of Lemma 2 and assuming that* 0 *does not belong to the spectrum of* A_n*, the time-discretized adjoint equation* (31) *admits a unique solution* $(\boldsymbol{v}_n, \beta_n) \in Z,\ N-1 \geq n \geq 1$.

Remark 6. *In Theorem 1, we made a strong assumption that zero is not in the spectrum of* A_n *or equivalently that the kernel of* A_n *is reduced to zero. Since the spectrum of* A_n *is discrete, it is unlikely that it generically contains the value zero. Even if zero is an eigenvalue of* A_n*, we can always perturb the coefficients in* A_n *(*$\boldsymbol{u}_n$ *and* α_n*) in such a way that the spectrum is perturbed so that zero is not anymore an eigenvalue.*

Proof. The linear form (33) is clearly continuous on Z. Therefore, solving the variational formulation (31) amounts to solve the linear equation in Z, $A_n(\boldsymbol{v},\beta) = f_n$. By virtue of Lemma 2, A_n admits a family of eigenvectors, which form a Hilbert basis of Z. By the spectral decomposition of A_n this equation has a unique solution if 0 does not belong to the spectrum of A_n. □

4 Numerical Implementation

In this section, we expound the numerical aspects of the resolution of the state equation (18) and adjoint equation (22) and the proposed shape optimization algorithm. For all the numerical test cases presented in Section 5, the material is chosen to be concrete having the following properties [12]: Young modulus $E = 29\text{GPa}$, Poisson ratio $\nu = 0.3$, ultimate tensile strength $\sigma_M = 4.5\text{MPa}$ and fracture toughness $G_c = 70\text{MPa}$. For the damage model, the characteristic length ℓ is calculated using the formula [50]

$$\ell = \frac{3G_cE}{8\sigma_M^2}. \tag{37}$$

The domain Ω is discretized by a simplicial unstructured mesh Ω^h. The mesh is produced by the `MMG` software [27] which features two important input parameters: the minimal and maximal mesh size, denoted by $h_{\min}$ and $h_{\max}$, respectively. The mesh Ω^h is assumed to be uniform in the sense that $h_{\max}$ and $h_{\min}$ are of the same order of magnitude. Following the numerical experiments in [43], for all our numerical test cases the mesh is chosen such that

$$2h_{\min} < \ell. \tag{38}$$

Although this choice was proposed in [43] for quadrilateral mesh elements, we follow it for our simplicial meshes. Nevertheless, we have to make one exception with the rule (38) in the test case of Subsection 5.8 (a realistic column of height $4m$) where we just enforce $h_{\min} < \ell$ in order to have a not too fine mesh which can be treated without resorting to high performance computing. The penalization parameter is chosen to be small, $\epsilon = \mathcal{O}(h_{\max}^2)$ (its precise value is given in the beginning of Subsection 5.1).

Remark 7. *Despite the fact that the damage model* (18) *is non-local, the crack initiation is mesh-dependent. For instance, mesh-refinement at corners of the shape* Ω *makes the crack initiation easier at these corners. For this reason we rely on uniform meshes (*$h_{\max}$ *and* $h_{\min}$ *of the same order), so that the crack initiation is unbiased.*

The spaces $V, C_v, H_D^1(\Omega), Z$, defined by (2), (3), (4) are discretized by piecewise affine continuous (linear) finite elements and their discrete counterparts are denoted by $V^h, C_v^h, H_D^1(\Omega^h), Z^h$, respectively.

The time interval $[0, T]$ is discretized in N intervals of length $\delta t = T/N$. The time at the end of the n-th time interval is denoted $t_n,\ n = 1, 2, \cdots, N$.

4.1 Solving the fracture model

We still denote by $(\boldsymbol{u}_n, \alpha_n) \in Z^h$ the time-space discretized solution at time interval t_n (we do not write its dependence to h). The space discretized version of (28) is simply the same variational formulation with the finite dimensional spaces $V^h, H_D^1(\Omega^h), Z^h$ replacing their continuous counterparts $V, H_D^1(\Omega), Z$. Following [16], the nonlinear variational formulation (28) is solved by a sequentially alternate algorithm: fixing α_n, solve (28a) for $\boldsymbol{u}_n$; fixing $\boldsymbol{u}_n$, solve (28b) for α_n. This algorithm exploits the fact that the total energy (11) is separately convex in $\boldsymbol{u}$ and α, but not with respect to the couple $(\boldsymbol{u}, \alpha)$, which may hinder the convergence of a standard Newton algorithm. This algorithm of [16] is precisely recalled in Algorithm 1.

The tolerance *tol* in Algorithm 1 is chosen to be 10^{-6}. The resolution for α_n (step 2 in the algorithm 1) using Newton algorithm is easy given that there are no constraints on α_n, because of penalization. Without penalization, one needs to apply a constrained Newton algorithm, the numerical implementation of which is not straight-forward. Algorithm 1 builds a sequence $(\boldsymbol{u}_n^i, \alpha_n^i) \longrightarrow (\boldsymbol{u}_n, \alpha_n)$ as $i \longrightarrow \infty$. Typically, its convergence is very fast when α_n is close to zero everywhere in Ω^h. Whereas the convergence is very slow when α_n approaches unity and there is crack formation in Ω^h.

Of course, the solution $(\boldsymbol{u}_n, \alpha_n)$ depends on the mesh size $h_{\max}$ and on the time step δt. Furthermore, each solution at time t_n depends on the whole time history before t_n. In practice, the solution depends on the initialization $(\boldsymbol{u}_n^0, \alpha_n^0)$ which is usually taken as the solution at the previous time step $(\boldsymbol{u}_{n-1}, \alpha_{n-1})$. Therefore, it is not clear that Algorithm 1 delivers an approximation of the global minimizer of the total energy (11) [20]. Rather, we may end up in local minima. Following again [16], we rather use the backtracking Algorithm 2 which escapes from local minima in practice.

Algorithm 1 Numerical resolution of (28)

Initialization: $(\boldsymbol{u}_n^0, \alpha_n^0) = (\boldsymbol{0}, 0)$ for $n = 0$ and $(\boldsymbol{u}_n^0, \alpha_n^0) = (\boldsymbol{u}_{n-1}^0, \alpha_{n-1}^0)$ for $n > 0$.
For $i = 0, 1, \cdots$, do

1. Substitute $\alpha_n = \alpha_n^i$ in (28a), and solve it for $\boldsymbol{u}_n = \boldsymbol{u}_n^{i+1}$
2. Substitute $\boldsymbol{u}_n = \boldsymbol{u}_n^{i+1}$ in (28b), and solve it for $\alpha_n = \alpha_n^{i+1}$ using a Newton-Raphson or a fixed point algorithm
3. If $\left\|\alpha_n^{i+1} - \alpha_n^i\right\|_{L^2(\Omega^h)} < tol \left\|1\right\|_{L^2(\Omega^h)}$, then exit the loop,
   else repeat

Algorithm 2 Numerical resolution of (28) with backtracking

Set $n = 1$ and $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) = (\boldsymbol{0}, 0)$
While $n \leq N$ (time steps), do

1. Solve (28) for $(\boldsymbol{u}_n, \alpha_n)$ using algorithm 1 using an initial guess $(\boldsymbol{u}_n^0, \alpha_n^0) = (\tilde{\boldsymbol{u}}, \tilde{\alpha})$
2. If $\|\alpha_n - \tilde{\alpha}\|_{L^\infty(\Omega^h)} > 0.5$ and $n > 1$

   then $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) = (\boldsymbol{u}_n, \alpha_n)$, and set $n = n - 1$,

   else $(\tilde{\boldsymbol{u}}, \tilde{\alpha}) = (\boldsymbol{u}_n, \alpha_n)$, and set $n = n + 1$.

The idea of Algorithm 2 is that, if the solution found at time-step n using Algorithm 1 features a strong increase of damage, the solution at the previous time step $n-1$ is re-computed using the solution at time step n as initial guess. In doing so, we expect to find a new solution $(\boldsymbol{u}_{n-1}, \alpha_{n-1})$ which is a better minimizer of the total energy (11).

4.2 Solving the adjoint problem

We still denote by $(\boldsymbol{v}_n, \beta_n) \in Z^h$ the time-space discretized adjoint solution at time interval t_n. As for the state solution in the previous subsection, the space discretized version of (30) is simply the same variational formulation with the finite dimensional spaces $V^h, H_D^1(\Omega^h), Z^h$ replacing their continuous counterparts $V, H_D^1(\Omega), Z$. As usual the adjoint problem is solved backward in time, i.e., for decreasing indices $n = N-1, \cdots, 1, 0$. One ought to solve the state equation (28) (using Algorithm 2) until the last time step, store the solutions $(\boldsymbol{u}_n, \alpha_n)$ for every time-step and retrieve the solutions starting from the last time step. As explained in Section 3.4, the bilinear form in the variational formulation (30) is not coercive. Hence, for numerical implementations, one ought to use a direct solver or an iterative technique like `GMRES`, that is capable of resolving indefinite matrices.

Remark 8. *The numerical resolution of* (30) *using a direct solver or* `GMRES` *is slow. If the damage variable is small* $\|\alpha_n\|_{L^\infty(\Omega^h)} \ll 1$*, the adjoint problem is close to a simple linear elasticity problem and one can rather use an iterative solver meant for positive definite matrices, for instance* `CG`*, to save computational effort.*

Finally, the space-time discretized version of the shape derivative of (29) is

$$J'_N(\Omega^h)(\boldsymbol{\theta}) = \int_\Gamma \boldsymbol{\theta}\cdot\boldsymbol{n} \sum_{n=0}^{N} \delta t \left(m(\boldsymbol{u}_n, \alpha_n) + \mathbb{C}'(\alpha_n)\beta_n \boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{u}_n) + \frac{G_c \ell}{2c_w} \nabla\alpha_n \cdot \nabla\beta_n + \frac{G_c}{4c_w}\frac{w'(\alpha_n)}{\ell}\beta_n \right.$$
$$\left. + \mathbb{C}(\alpha_n)\boldsymbol{\varepsilon}(\boldsymbol{u}_n) : \boldsymbol{\varepsilon}(\boldsymbol{v}_n) + \frac{G_c}{\epsilon}\left(\mathcal{M}(\alpha_n - 1) - \mathcal{M}\left(\frac{\alpha_{n-1} - \alpha_n}{\delta t}\right)\right)\beta_n \right) ds. \tag{39}$$

4.3 Level-set method and remeshing

The level-set method was introduced by Osher and Sethian [47] and adapted to the shape optimization framework [3, 51]. In this method, a shape $\Omega \subset D$ is represented by a level-set function

$\phi : D \mapsto \mathbb{R}$, which is defined as

$$\begin{cases} \phi(x) < 0 & \text{if } x \in \Omega, \\ \phi(x) = 0 & \text{if } x \in \Gamma, \\ \phi(x) > 0 & \text{if } x \in D \setminus \overline{\Omega}. \end{cases}$$

where we recall that Γ is the movable part of the boundary $\partial\Omega$. The crux of the method lies in letting the shape deform along a velocity field $\boldsymbol{\theta} : D \mapsto \mathbb{R}^d$. The evolution of the shape is governed by the transport equation

$$\frac{\partial \phi}{\partial t} + \boldsymbol{\theta} \cdot \nabla \phi = 0. \tag{40}$$

Very often, the velocity field is oriented along the normal, namely $\boldsymbol{\theta} = \theta \boldsymbol{n}$ where $\boldsymbol{n} = \nabla\varphi/|\nabla\varphi|$ and the scalar function θ is the normal velocity. In such a case, (40) can be re-written as a Hamilton-Jacobi equation

$$\frac{\partial \phi}{\partial t} + \theta |\nabla \phi| = 0. \tag{41}$$

In (41) the scalar normal velocity θ is chosen to be the shape derivative (39), or a proper extension-regularization of it (see the next subsection and (47) for details).

In our numerical setting, we solve the linear transport equation (40) instead of (41) because we use non-cartesian meshes and rely on the advect library [18] which solves (40) by the method of characteristics, known to be unconditionally stable. The level-set function is a $\mathbb{P}^1$ function on a simplicial mesh of the computational domain D. After every advection, the new shape is captured using a body-fitted mesh obtained using the `MMG` software [27]. On this new mesh, the level-set function (precisely, the signed distance function) is reconstructed using another open-source library `mshdist` [28]. Remeshing based shape optimization, relying on `MMG`, has already been carried out for several physical problems [1, 30, 31] and we refer to these papers for more details. The `MMG` software requires two important input parameters: $h_{\min}$, the lower bound on the smallest mesh size, and $h_{\max}$, the upper bound on the largest mesh size. Then, it builds a good quality mesh, satisfying these constraints, as much as possible.

4.4 Regularization and extension of the shape derivative

The shape derivative (39) is defined only on the boundary Γ, while it is needed in the full computational domain D for solving the transport equation (40). Furthermore, both the derivative and the shape may be not smooth enough, which may result in poor numerical efficiency. Therefore, it is convenient to regularize the shape derivative (39) [2, 19], still ensuring that it is a descent direction. A classical possibility is to consider the $H^1(D)$ scalar product (instead of the $L^2(\Gamma)$ scalar product) for identifying the gradient. In other words, introducing a mesh D^h of D and denoting by $H^1(D^h)$ the linear finite element subspace of $H^1(D)$, we seek a function $dj(\Omega^h) \in H^1(D^h)$ such that

$$\int_D \left(4h_{\min}^2 \nabla dj(\Omega^h) \cdot \nabla\varphi + dj(\Omega^h)\varphi\right) dx = \int_\Gamma j'(\Omega^h)\, \varphi \, dx \qquad \forall \varphi \in H^1(D^h), \tag{42}$$

where $h_{\min}$ is the fixed minimal mesh size and $j'(\Omega^h)$ is the function defined by formula (39) with

$$J'_N(\Omega^h)(\boldsymbol{\theta}) = \int_\Gamma \boldsymbol{\theta} \cdot \boldsymbol{n}\, j'(\Omega^h)\, ds.$$

For the implementation details of the above, an interested reader can refer to [29].

4.5 Shape optimization algorithm

We consider the shape optimization problem (20) where the admissible shapes must satisfy a constraint on the target volume V_f. To do so, the following Lagrangian is introduced

$$\mathcal{L}(\boldsymbol{u}, \alpha, \boldsymbol{v}, \beta, \Omega^h, \lambda) = J(\Omega^h) + \frac{\lambda}{C_V}\left(\int_{\Omega^h} dx - V_f\right), \tag{43}$$

where λ is the Lagrange multiplier for the volume constraint and C_V is a normalization constant. Denoting by Ω_0^h the initial shape, the constant C_V is defined by

$$C_V = \left| \int_{\Omega_0^h} dx - V_f \right|. \tag{44}$$

We apply a standard gradient-based Uzawa-type algorithm to the Lagrangian (43). Let $\mathcal{N}$ be the maximal number of shape optimization iterations (typically $\mathcal{N} = 200$ for most of the test cases in the next section). The iteration number is denoted by i with $1 \leq i \leq \mathcal{N}$. At each iteration i, once the shape derivative $dj(\Omega_i^h)$ is evaluated by (42), a pseudo-time step (or descent step) τ is defined by

$$\tau = \frac{h_{\min}}{2C_i}, \tag{45}$$

where $h_{\min}$ is the minimal mesh size of the first iteration and C_i is a normalization constant, given by

$$C_i = \int_{\partial\Omega_i^h} |dj(\Omega_i^h)|\, dx.$$

Updating the constant C_i at every iteration of the optimization process ensures a control of the descent step τ. The multiplier λ is also updated at each iteration by

$$\lambda_{i+1} = \lambda_i + \frac{C_i \tau}{C_V} \left(\int_{\Omega_i^h} dx - V_f \right), \tag{46}$$

ensuring that the volume will converge (slowly) to the target volume. Then, for the descent step τ, the transport equation (40) is solved with a velocity θ_i, given by

$$\boldsymbol{\theta}_i = \left(dj(\Omega_i^h) + \frac{\lambda_{i+1}}{C_V} \right) \boldsymbol{n}, \tag{47}$$

where $\boldsymbol{n} = \nabla \phi_i$ is the normal to the level-set function associated to the shape Ω_i^h.

To improve the satisfaction of the volume constraint, we apply the following trick. As soon as the volume is close to the volume target, namely $|V_{i+1} - V_f| \leq 10^{-1} V_f$, we apply a projection algorithm to satisfy the target volume exactly. More precisely, the level-set ϕ_{i+1} is iteratively updated by

$$\phi_{i+1} = \phi_{i+1} + \frac{V_{i+1} - V_f}{\int_{\partial\Omega_{i+1}^h} ds}, \tag{48}$$

until $|V_{i+1} - V_f| \leq 10^{-4} V_f$. The newly obtained shape Ω_{i+1}^h is remeshed with `MMG` [27]. Eventually, the objective function $J(\Omega_{i+1}^h)$ is evaluated but is not compared to the previous value $J(\Omega_i^h)$. Summing up this sub-section, we basically implement Algorithm 3.

Remark 9. *If at iteration i the objective function does not decrease, compared to its value at the previous iteration $i-1$, we do not step back to the previous iteration with a reduced descent or pseudo-time step (in order to ensure a decrease in objective function). This is due to the non-smooth nature of the damage problem* (28)*, where the onset of fracture is very sensitive to the loading and the geometry and yields large variations of the objective function. We shall explain this issue in greater details in Remark 10 in Sub-section 5.2.*

4.6 Parallel implementation

All our numerical experiments are performed with the open-source software `FreeFEM 4.8` [36], installed on a workstation featuring a Intel(R) Xeon(R) Gold 6230 CPU and 40 processors. In the next section we shall present a large scale topology optimization test case, consisting of approximately 1.3 million tetrahedra. It would be impossible to perform this test case on a single processor, because its CPU time would be at least a few weeks. Indeed, 90% of the total computation time is spent on the first three steps of Algorithm 3, namely on the finite element analyses.

Algorithm 3 Shape optimization for the damage model

Initialize with a shape Ω_0^h and repeat over $i = 1, \cdots, \mathcal{N}$

1. Solve for the state $(\boldsymbol{u}, \alpha)$ in Ω_i^h marching in time from t_1 until t_N using Algorithm 2
2. Solve for the adjoint $(\boldsymbol{v}, \beta)$ in Ω_i^h backward in time from t_N up to t_1
3. Compute the shape derivative using (39) and regularize it with (42) to deduce $dj(\Omega_i^h)$
4. Update the Lagrange multiplier λ_{i+1} with (46)
5. Solve the transport equation (40) with the velocity given by (47) for the pseudo-time step τ given by (45) to obtain the new level-set function $\tilde{\phi}_{i+1}$
6. Re-initialize $\tilde{\phi}_{i+1}$ to the signed distance function ϕ_{i+1} (defining a new shape Ω_{i+1}^h)
7. Compute the volume V_{i+1}. If it is close to the volume target, apply the projection algorithm (48) to satisfy exactly the volume constraint.
8. Remesh the box D using `MMG` [27] to obtain the body fitted mesh of the new shape Ω_{i+1}^h

Therefore, in order to minimize the total computation time, we must perform parallel computation for the finite element analysis. Fortunately, `FreeFEM` comes with built-in OpenMPI and the open-source package `PETSc` [10]. The resolution of the state equation (28) (using Algorithm 2), the adjoint equation (30) and the regularization of the shape derivative (42) are performed using the parallel-solver of `PETSc`. The mesh is partitioned in sub-domains using the open-source package `METIS` [41]. The finite element rigidity matrix is thus partitioned accordingly and the linear systems are solved in parallel with the `GAMG` (geometric algebraic multigrid) preconditioner. The state equation and the regularization problem are solved by the conjugated gradient algorithm, while the adjoint equation is solved by `GMRES`. For details of implementation we refer to the tutorial on the parallel version of `FreeFEM` [39].

5 Results

We now present 2D and 3D shape optimization results for the damage model (18), which prove the effectiveness of our algorithm to produce crack-free structures.

5.1 Setting and parameters

We choose to maximize rigidity, namely to minimize an objective function which is the total compliance. As already explained in Subsection 2.1, no body or surface forces can be considered in the Francfort-Marigo damage model and, rather, one has to impose given displacements to the structure. In such a case, there is a subtle definition of compliance, see e.g. [11], which takes into account the reaction force on the surface where the displacement is imposed. Since the work done by this reaction force is equal to the elastic energy, the total compliance is defined as

$$J(\Omega) = -\int_0^T \int_\Omega \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) dx\, dt. \tag{49}$$

The objective function (49) is minimized in all test cases, except otherwise mentioned.

The final time is $T = 1s$ and the time step is $\delta t = 0.15s$. The solution $(\boldsymbol{u}, \alpha)$ is computed by solving the penalized formulation (28) using Algorithm 2. The penalization parameter is chosen equal to $\epsilon = 10^{-5}$.

An isotropic degradation function, with a small residual stiffness $\mathbb{C}_{res}$, is considered

$$\mathbb{C}(\alpha) = (1-\alpha)^2 \mathbb{C}_0 + \mathbb{C}_{res}. \tag{50}$$

In practice the residual stiffness is $\mathbb{C}_{res} = 10^{-6}\mathbb{C}_0$ but it could even be taken to be zero if the finite element solver is robust enough (which is the case with `FreeFEM`). For all test cases, only the dissipation function (17) of the `DL` model is considered.

Since we are in a quasi-static evolution framework, the rate of increment of the imposed displacement has no effect on the solution $(\boldsymbol{u}, \alpha)$ at the final time $T = 1s$. But the rate does have a strong influence on the objective function (49) (since it contains a time-integral). In order to see a greater influence of the damage variable α on the optimized shape, we consider an imposed displacement $\bar{\boldsymbol{u}}(t)$ that grows from zero to a certain value and then remains constant for some period.

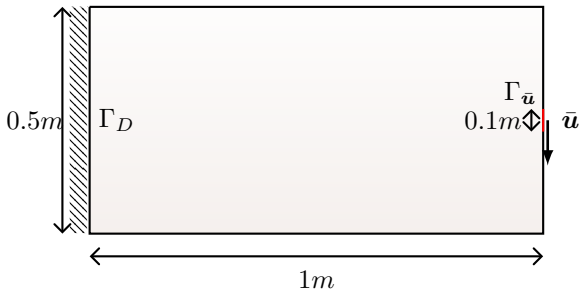


Figure 3: 2D Cantilever boundary conditions

5.2 2D Cantilever

We study a 2D cantilever beam represented by a rectangle of dimensions $1m \times 0.5m$ as shown in Fig. 3. The cantilever beam is clamped all along its leftmost edge and subjected to an applied vertical displacement,

$$\bar{\boldsymbol{u}}(t) = (0, 4\min(1.2t, 1)) \times 10^{-4} m, \quad t \in [0, 1], \tag{51}$$

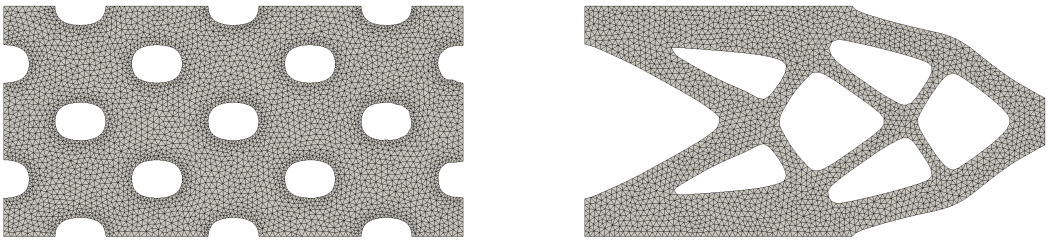
on a centered part of its rightmost edge $\Gamma_{\bar{\boldsymbol{u}}}$ of length $0.1m$. The above displacement (51) is chosen in such a way that the initial shape of Fig. 4a suffers from a crack as seen in Fig. 6a. A target volume $V_f = 0.25m^2$ is imposed for all the test cases in this subsection. The parameters of the remeshing tool `MMG` are: $h_{\min} = 0.0064m, h_{\max} = 0.0128m$.

The initial shape is displayed in Fig. 4a. The shape obtained by minimizing (49) for linear elasticity (without any damage) is plotted in Fig. 4b. The shape obtained by minimizing (49) for the damage model is plotted in Fig. 4c. The convergence history is plotted in Fig. 5.

As can be seen in Fig. 4, the optimized shapes, with or without damage in the mechanical model, are slightly different, but share the same topology. The shape in Fig. 4c do not undergo a crack, unlike the shape in Fig. 4b that does undergo a crack (see Fig. 7a). For the shape in Fig. 4c, the damaged region in the intermediate shapes is plotted in Fig. 6. The optimization algorithm indeed tries hard to remove every damaged or cracked region that appears. We observe that the crack appears in the cantilever at several locations, taking different configurations at each iteration and disappears finally after the 45-th iteration.

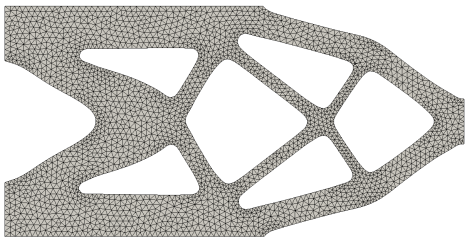
Remark 10. *One can see in Fig.5 that the objective function* (49) *features very strong oscillations during the first 50 iterations although the volume constraint is nicely satisfied after 20 iterations. These peaks in the convergence history occur whenever there is a transition from a shape without any crack to a shape with a crack, or when the crack changes from one position to a completely different one (see Fig. 6). In other words, small perturbations in the shape Ω can result in the appearance or disappearance of cracks, leading to abrupt changes in the objective function* (49)*. Reducing the descent step τ would not help here because the onset of fracture is a discontinuous process with respect to load or geometry variations. Typically, the growth in time of a fracture can be discontinuous. Therefore, it is plausible that a small change in the geometry of the shape can induce a large change in the crack profile and thus in the value of the objective function* (49)*. This non-smooth character of fracture or damage is well documented in the theoretical literature [13, 21, 22, 23, 48] but also in the numerical literature [16]. Note that our derivation of the shape derivative in Subsection 3.3 was performed under the assumption of a smooth solution of the damage model* (18)*.*

The regularization of the damage field α with the characteristic length ℓ or the penalization process of the damage irreversibility do not help at all on this matter. Our numerical experiments confirm this non-smoothness of the damage problem and the discontinuity of the objective function with respect to shape variations. Therefore, it is questionable to use a gradient-descent method to



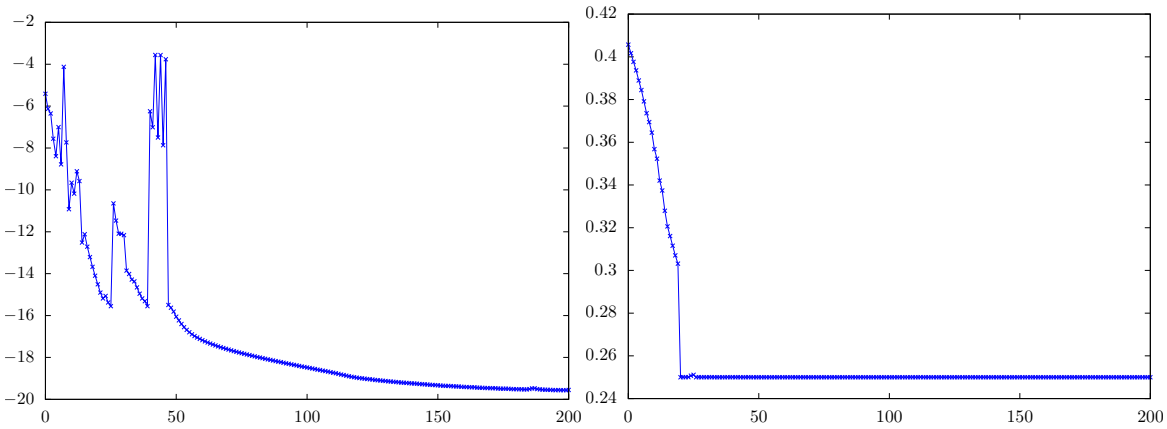
(a) Initialization

(b) Final shape obtained for linear elasticity



(c) Final shape for the damage model

Figure 4: Initial and final meshes for the cantilever shapes of Subsection 5.2



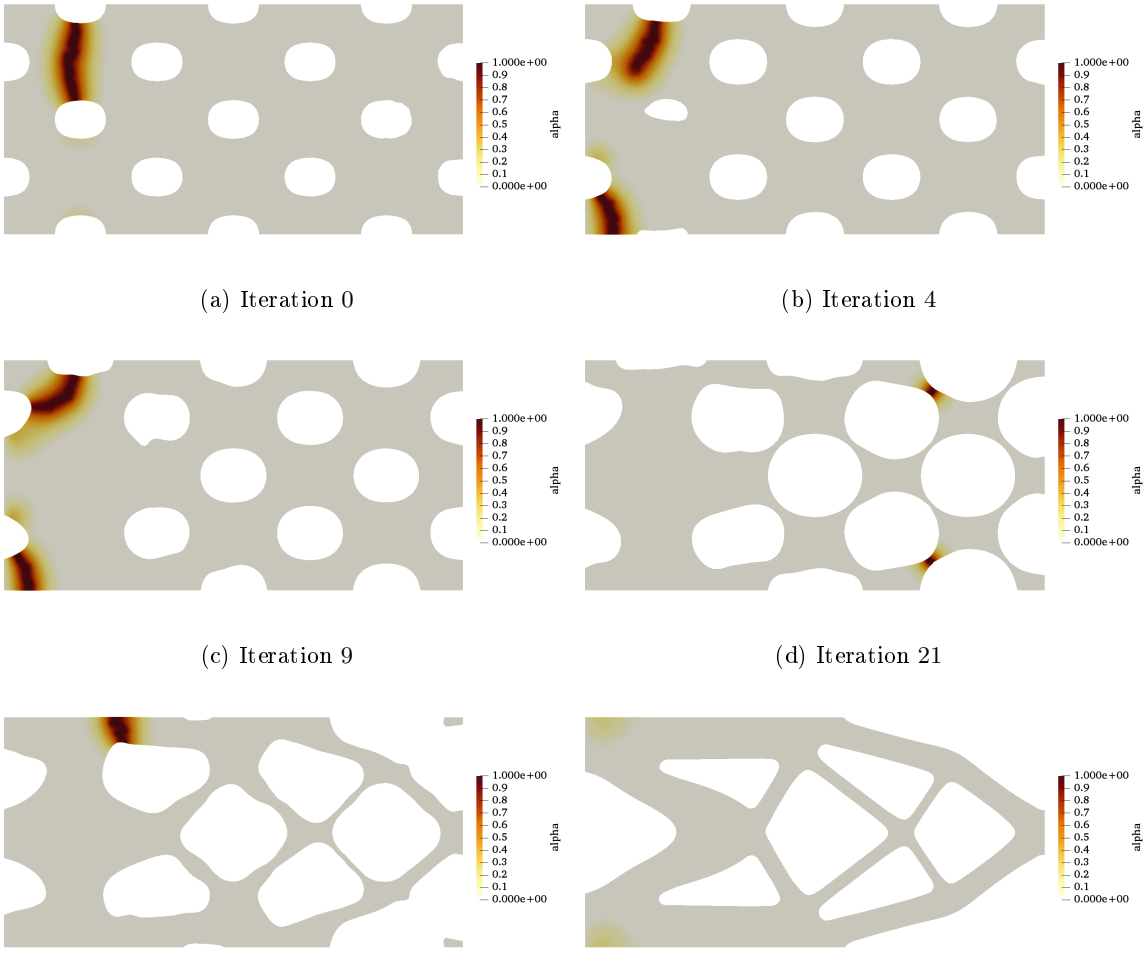
(a) Total elastic energy (49) v/s iterations

(b) Volume v/s iterations

Figure 5: Convergence history for the optimized shape of Fig. 4c

minimize the objective function (49). *Nevertheless, the presented test cases show that, after some early oscillations, our gradient algorithm does converge to a crack-free optimal shape. There are two key ingredients for this relative success. First, although the descent step* τ *is adapted at each iteration by formula* (45)*, we do not test if the objective function decreases at each iteration and we never step back with a smaller descent step. Second, we rely on the backtracking Algorithm 2 (following [16]) which plays a pivotal role in ensuring a stable damage evolution. As a consequence, the shapes obtained in the final iterations are more stable in the sense that small geometric perturbations do not cause the appearance of a crack. Hence oscillations in the objective function are avoided and a smooth convergence of the objective function is attained. This is confirmed by our attempt to replace the backtracking Algorithm 2 by the simpler Algorithm 1 (without backtracking) in our shape optimization Algorithm 3. We noticed that the fluctuations in the objective function were more violent than the ones obtained with backtracking, which was hindering convergence.*

One could think that changing the initialization could improve the convergence of the shape



(a) Iteration 0

(b) Iteration 4

(c) Iteration 9

(d) Iteration 21

(e) Iteration 40

(f) Iteration 200

Figure 6: Damage variable α at the final time T plotted for several optimization iterations for the shape in Fig. 4c

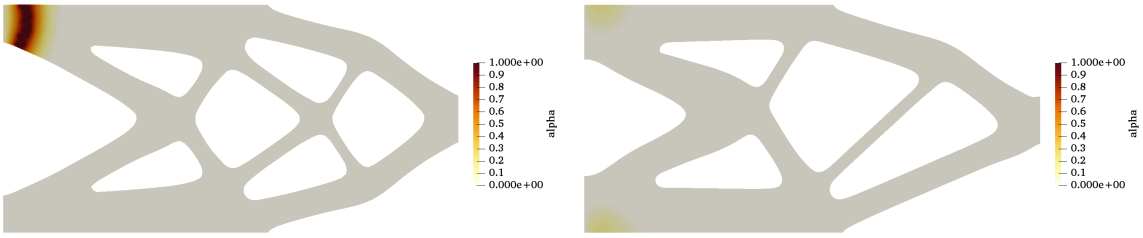
optimization for the damage model. For example, instead of starting from the periodically perforated initial shape in Fig. 4a, it is possible to initialize the damage model optimization with the optimal shape for linear elasticity in Fig. 4b. We perform this new test case and the result in Fig. 7b is quite deceiving. Indeed, it takes 600 iterations (3 times more, see Fig. 8) to converge to the shape of Fig. 7b, which is different from that previously obtained in Fig. 4c, slightly less optimal since the objective function for Fig. 4c is 0.7% better than for the shape of Fig. 7b. Note that both shapes of Fig. 7b and 4c feature no crack.

5.3 Traction-only degradation

In this subsection only, we replace the isotropic degradation function (50) by the following traction-only degradation function [43]

$$\mathbb{C}(\boldsymbol{u},\alpha) = \Big(\mathcal{H}(\mathrm{div}\boldsymbol{u})(-1+(1-\alpha)^2)+1\Big)\mathbb{C}_0 + \mathbb{C}_{res}, \tag{52}$$

where $\mathcal{H}$ denotes the Heaviside function. If $\mathrm{div}\boldsymbol{u} \geq 0$ the material is said to be in traction, otherwise it is in compression. The degradation function (52) is constructed in such a way that damage occurs only under tension. In other words, when $\mathrm{div}\boldsymbol{u} < 0$, whatever the value of α, one has $\mathbb{C}(\boldsymbol{u},\alpha) = \mathbb{C}_0$. Such a traction-only degradation function is more realistic since it can make a difference between



(a) Initial shape as in Fig. 4b (b) Final shape

Figure 7: Damage variable α at the final time T for the initial shape, as in Fig. 4b, and for the resulting optimal shape

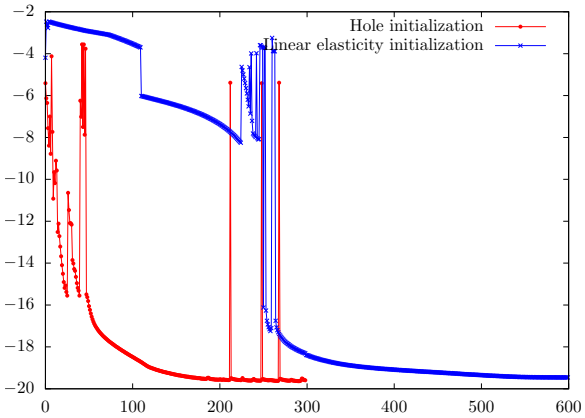


Figure 8: Objective function (49) v/s iterations for the shapes (4c) and (7b)

an opening and a closing cracks (this idea was introduced in [9] with a slightly different degradation function). The Hooke's tensor $\mathbb{C}(\boldsymbol{u}, \alpha)$ obviously depends on $\boldsymbol{u}$, and furthermore is not even differentiable with respect to $\boldsymbol{u}$. Nevertheless, for the numerical test here, we ignore this dependence and do not take it into account in the adjoint equation.

The same 2D cantilever beam, as in Subsection 5.2, is considered for the new degradation function (52). Starting from the initialization in Fig. 4a the algorithm converges to the shape of Fig. 9. The convergence history, in Fig. 10, is slightly smoother than in Fig. 5. The optimized shape in Fig. 9 is somehow intermediate between those in Fig. 4 for linear elasticity and the original isotropic degradation function (50).

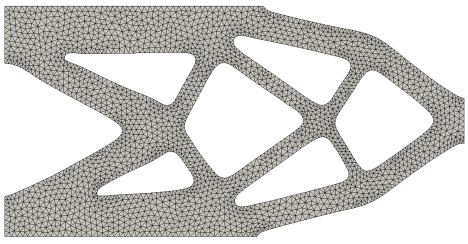
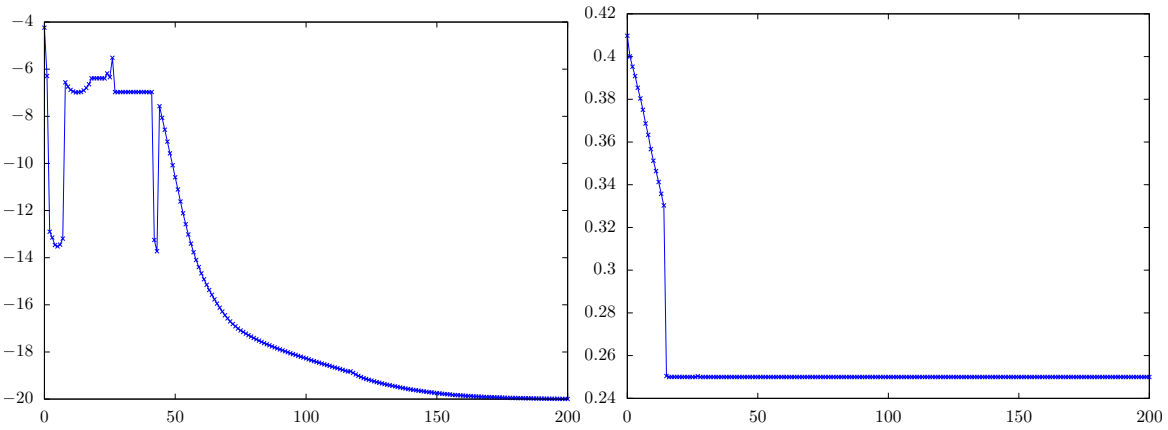


Figure 9: Final mesh for the anisotropic degradation function (52)



(a) Objective function (49) v/s iterations

(b) Volume v/s iterations

Figure 10: Convergence history for the shape of Fig. 9

5.4 2D Cantilever: some variants

From the test cases of the previous subsections, we make two important observations:

1. the objective function converges in a highly non-smooth manner (see Fig. 5),
2. the final optimized shapes have zones that undergo slight damage (see Fig. 4 and 7b).

The first point is a consequence of two facts: (i), the damage model is very sensitive to all parameters (loading, geometry, material parameters), (ii), the initialization with a periodically perforated shape, which is very far from any possible optimal shape, implies that the optimization process will explore "wild" intermediate shapes. In order to ensure that the convergence of the objective function is smoother and that, if possible, the final shape features no damaged zones, we implemented the following three variants of our approach to improve the smoothness of the optimization process.

1. Incremental approach: the residual stiffness in the degradation function (50) is changed, taking a decreasing residual stiffness, as the iteration number increases

$$\mathbb{C}_{res} = \kappa_i \mathbb{C}_0 \tag{53}$$

where κ_i is (heuristically) given by

$$\kappa_i = \frac{1}{10}\left(1 - \frac{i}{\mathcal{N}}\right)^p, \tag{54}$$

where i it the iteration number, p is an integer exponent and $\mathcal{N}$ is the total number of optimization iterations.

2. Symmetrization: the level-set function is symmetrized about the horizontal x-axis (given the observation that the optimal shapes of Fig. 4 have this symmetry which can be broken by the non-linear character of the damage model).

3. Weighted objective function: to give more importance to the final time, when the structure is more likely to endure fracture, replace the objective function (49) by its weighted or truncated version

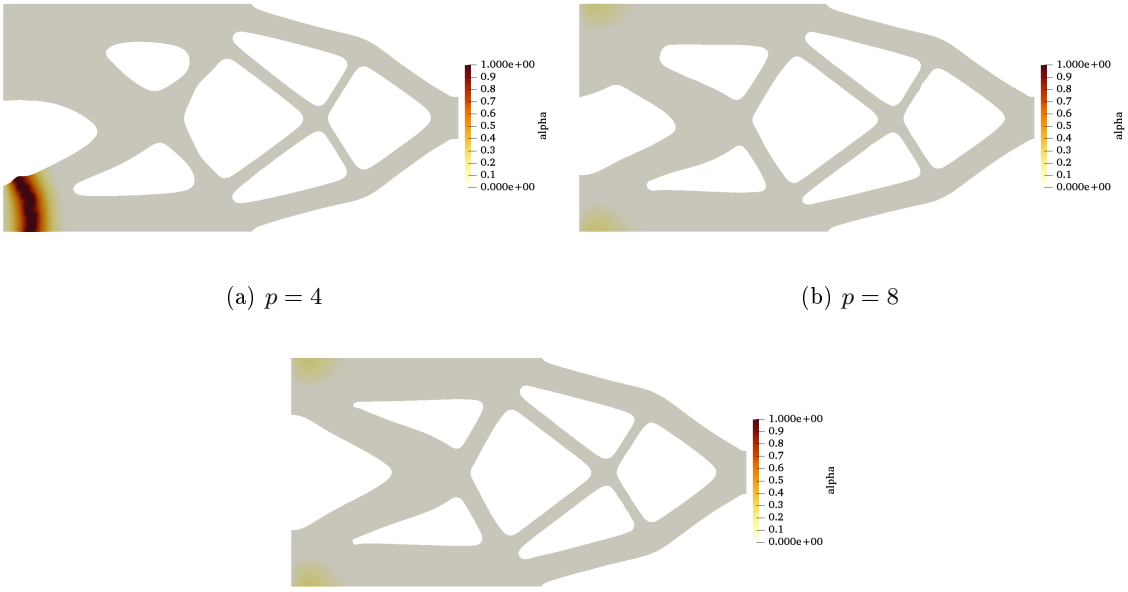
$$J(\Omega) = -\int_{T_0}^{T}\int_{\Omega} \mathbb{C}(\alpha)\boldsymbol{\varepsilon}(\boldsymbol{u}):\boldsymbol{\varepsilon}(\boldsymbol{u}) dx\, dt, \tag{55}$$

where $T = 1$ and $T_0 = 0.85 > 1/1.2$ (thus the imposed displacement (51) remains constant on this time interval).

These three variants are tested with Algorithm 3 and exactly 200 iterations.

In the first variant, the parameter κ_i of the residual stiffness in (53) is essentially the stiffness of the damaged phase. If κ_i is not small, the objective function does not increase significantly when

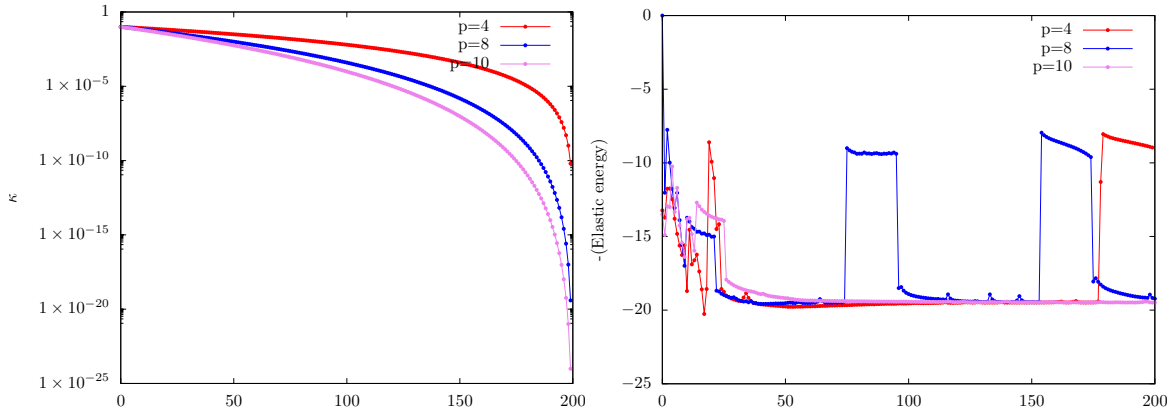
the structure undergoes a crack. However, when κ_i is small, meaning that the residual stiffness approaches zero, the objective function increases dramatically when a crack appears. Given that the objective function oscillates in the first iterations (see Fig. 5), it is legitimate to try to reduce these oscillations by having a larger residual stiffness at the start and then gradually decrease it, as is the case with (54). Three different exponent are tested: $p = 4, 8$ and 10. The graphs of κ_i are plotted in Fig. 12a. The shapes obtained for the three exponents are displayed in Fig. 11 and the corresponding convergence histories are given in Fig. 12b. As can be seen in Fig. 11, the shapes obtained for $p = 4$ and $p = 8$ are not optimal, unlike the shape obtained for $p = 10$ (which resembles the one of Fig. 4c). In the convergence history of Fig. 12b, we observe that the fluctuations in the objective functions are high for $p = 4$, lesser for $p = 8$ and least for $p = 10$. Hence it seems preferable to decrease the residual stiffness in (54) with a large exponent p.



(a) $p = 4$ (b) $p = 8$

(c) $p = 10$

Figure 11: Shapes obtained after 200 iterations and damage field α at the final time for the decreasing residual stiffness κ_i as in Fig. 12a



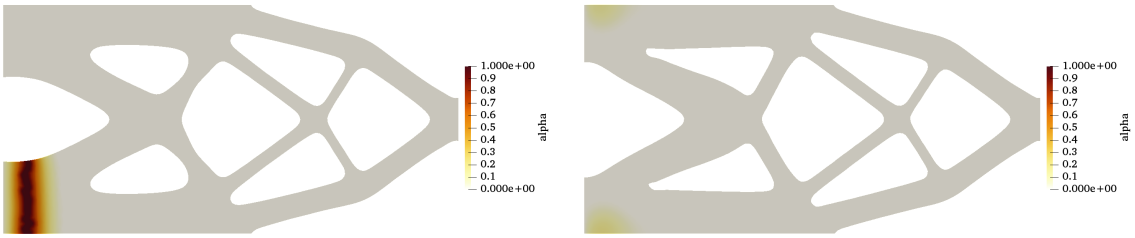
(a) Degradation parameter κ v/s iterations (b) Objective (49) v/s iterations

Figure 12: Convergence history for the shapes in Fig. 11

In the second variant, Algorithm 3 is slightly modified. After the re-initialization step 6, the

level-set function ϕ_{i+1} is symmetrized around the horizontal central axis, which ensures that the shape remains symmetric. After 200 iterations of shape optimization, the shape in Fig. 13a is obtained, which is obviously not optimal as it undergoes a crack, unlike the shape in Fig. 9.

For the third variant, we obtain the shape of Fig. 13b, which is similar to the one obtained in Fig. 4c and undergoes only a very minor damage. This is quite deceiving as we weighted or truncated the objective function (49) in the hope of totally eliminating any damage zone. Thus, considering a weighted version of the objective function (49) is not effective in avoiding damaged zones.



(a) Symmetrization variant

(b) Weighted objective variant (55)

Figure 13: Damage field α plotted at the final time T for the optimal shapes obtained after 200 iterations

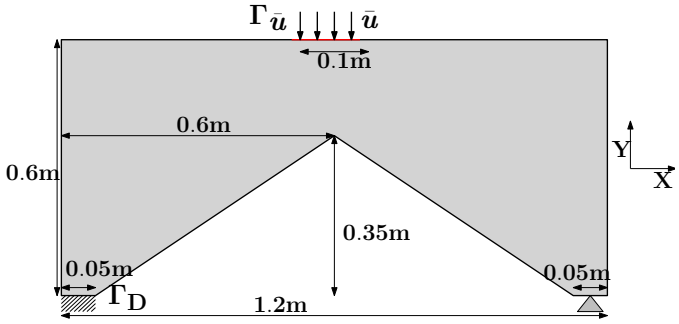


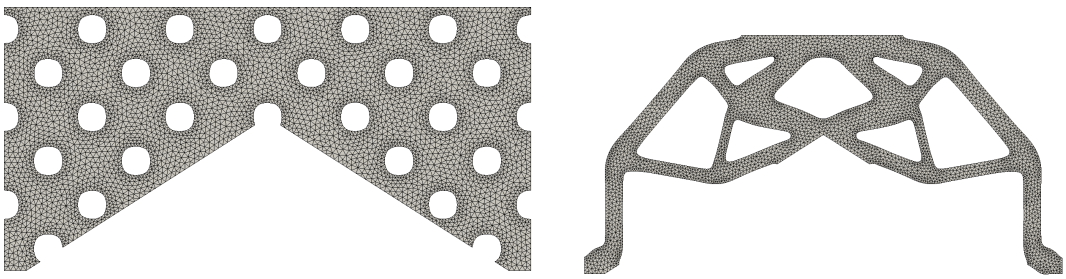
Figure 14: 2D wedge boundary conditions

5.5 2D wedge

We now study a 2D wedge as shown in Fig.14. The wedge is fixed on its leftmost support, with a zero vertical displacement on its rightmost support, and subjected to an imposed displacement,

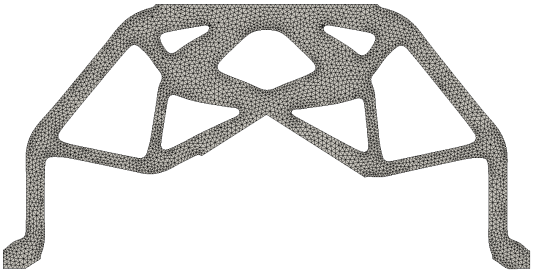
$$\bar{\boldsymbol{u}}(t) = (0, 2.33\min(1.2t, 1)) \times 10^{-4}m, \quad t \in [0, 1], \tag{56}$$

on the center of its top edge as shown in Fig. 14. These prescribed boundary conditions and the sharp corner at the center ensure that there is a crack initiation at the notch in the center. The input parameters of the remesher `MMG` are $h_{\min} = 0.0064m$ and $h_{\max} = 0.0128m$. The target volume is taken to be $0.2m^2$. The initial shape, as shown in Fig. 15a, converges to the shape in Fig. 15b for linear elasticity and to the shape in Fig. 15c for the damage model. The convergence history is plotted in Fig. 16. Like in the previous 2D-cantilever case, the shapes obtained for linear elasticity or for the damage model have the same topology. In Fig. 17 are plotted the damaged regions for several intermediate shapes corresponding to the final shape of Fig.15c. A crack is clearly present during the first iterations although the notch is rounded by a hole. Nevertheless, after 21 iterations, the algorithm finds a crack-free shape, finally converging to the final shape at iteration 200, which features a very little amount of damage close to the notch.



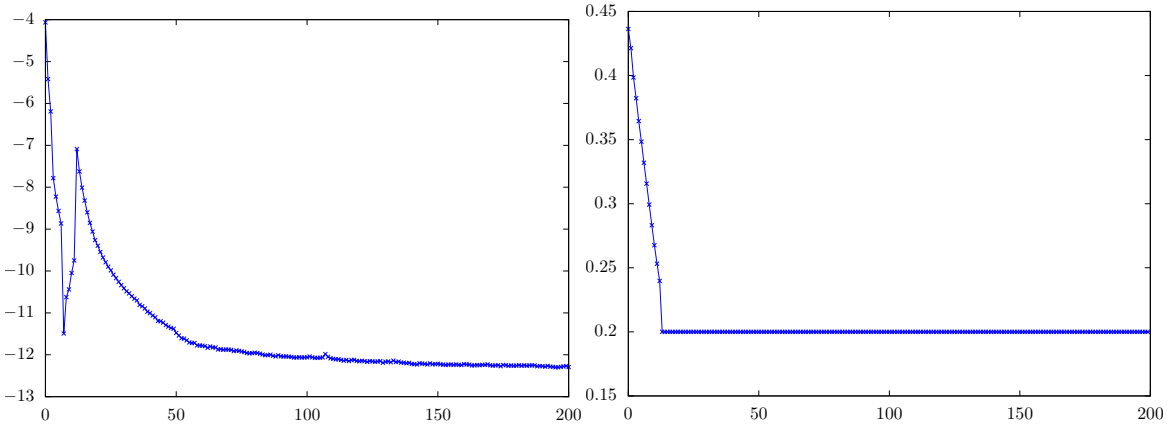
(a) Initialization

(b) Final shape obtained for linear elasticity



(c) Final shape for the damage model

Figure 15: Initial and final meshes for the 2D wedge



(a) Objective function (49) v/s iterations

(b) Volume v/s iterations

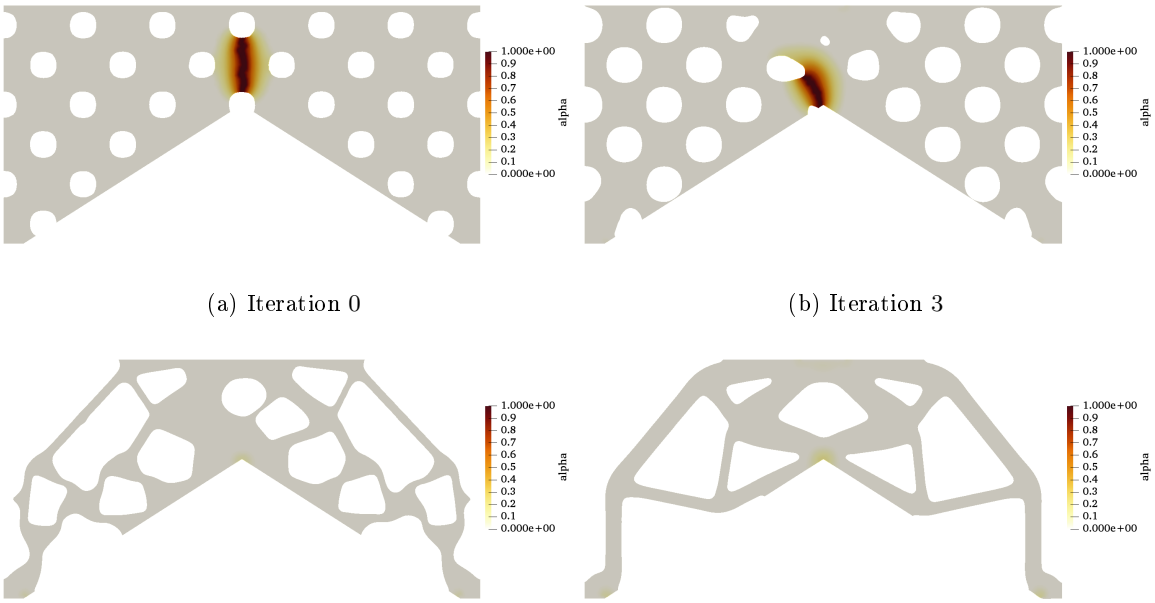
Figure 16: Convergence history for the wedge shape in Fig. 15c

5.6 2D L-beam

The last 2D example is an L-beam which is a meaningful test case because of its re-entrant right-angled corner, which is prone to crack initiation. As shown in Fig. 18, the L-beam is fixed on its topmost edge, subjected to an imposed displacement,

$$\bar{\boldsymbol{u}}(t) = (0, 1 \min(1.2t, 1)) \times 10^{-3} m, \quad t \in [0, 1], \tag{57}$$

on a small part of its rightmost edge with a vertical force as shown in Fig. 18. A target volume $V_f = 2m^2$ is considered. The input parameters for `MMG` are $h_{\min} = 0.01m$ and $h_{\max} = 0.02m$. From the initial shape in Fig. 19a, the minimization of the objective function (49) for linear elasticity leads to the final shape in Fig. 19b. Contrary to the previous test cases, we now initialize the optimization for the damage model with this shape of Fig. 19b, instead of the periodically perforated initialization



(a) Iteration 0

(b) Iteration 3

(c) Iteration 21

(d) Iteration 200

Figure 17: Damage variable α at the final time T plotted for several optimization iterations for the shape in Fig.15c

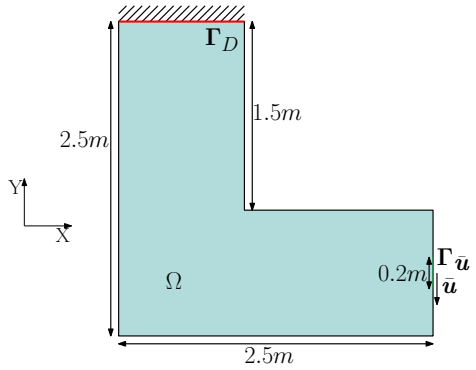
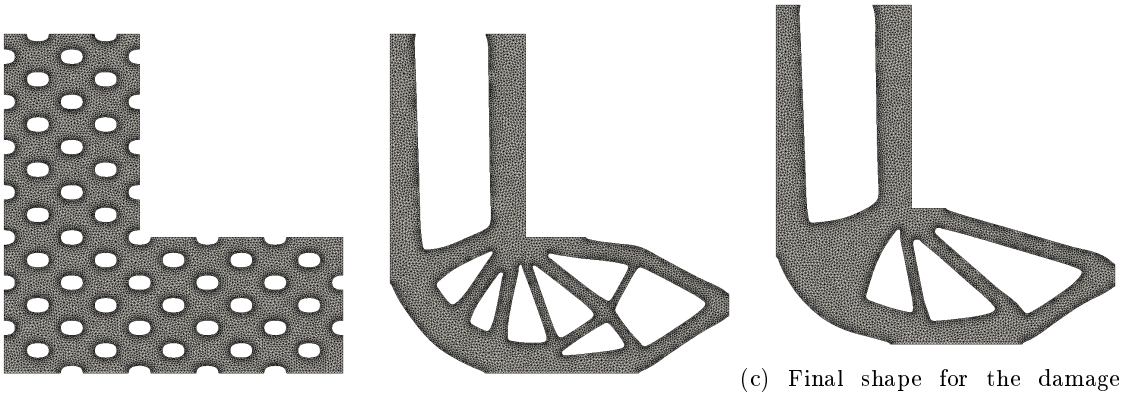


Figure 18: 2D L-beam boundary conditions

of Fig. 19a. It yields the final shape of Fig. 19c. In order to compare the shapes in Fig. 19b and in Fig. 19c, we perform a damage computation for both shapes with $\bar{\boldsymbol{u}}(t)$, given by (57), and plot the damage variable α at the final time in Fig. 20: obviously, the shape in Fig. 19c does not undergo a crack, unlike the shape in Fig. 19b.

5.7 3D wedge

Continuing the study of test cases with sharp corners, we consider a 3D wedge as shown in Fig. 21. The wedge is supported on four square surfaces, each being $0.05m \times 0.05m$, at the bottom of the working domain, see Fig. 21. The wedge is clamped in all three directions on one surface and only along the vertical y-direction on the remaining three surfaces. The wedge is subjected to an imposed displacement $\bar{\boldsymbol{u}}(t)$ on a $0.1m \times 0.1m$ square surface at the top (in yellow in Fig. 21). The input parameters for `MMG` are $h_{\min} = 0.013m$ and $h_{\max} = 0.026m$. Here, we investigate the impact of increasing the magnitude of the imposed displacement on the optimized shape and hence consider



(a) Initialization mesh (b) Final shape for linear elasticity model (c) Final shape for the damage

Figure 19: Initial and final meshes for the 2D L-beam

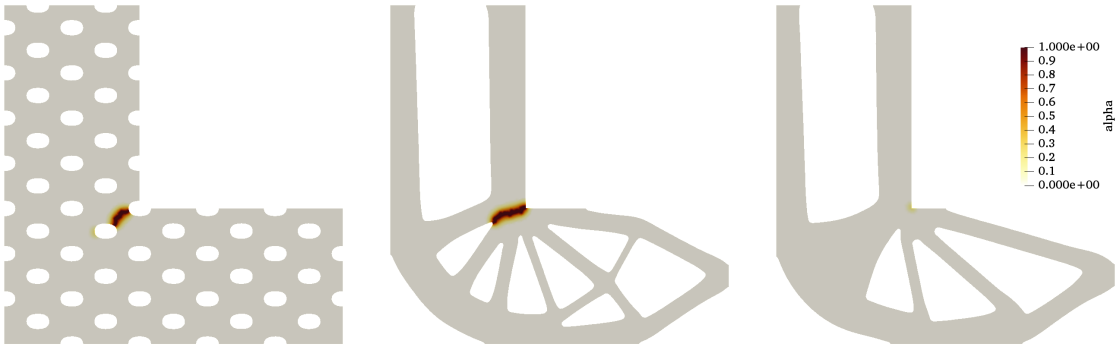


Figure 20: Damage variable α at the final time T plotted for several optimization iterations for the shapes in Fig.19

two functions

$$\bar{\boldsymbol{u}}(t) = (0, 7\min(1.2t, 1)) \times 10^{-5} m, \quad t \in [0, 1], \quad \text{and} \tag{58}$$

$$\bar{\boldsymbol{u}}(t) = (0, 8.5\min(1.2t, 1)) \times 10^{-5} m, \quad t \in [0, 1]. \tag{59}$$

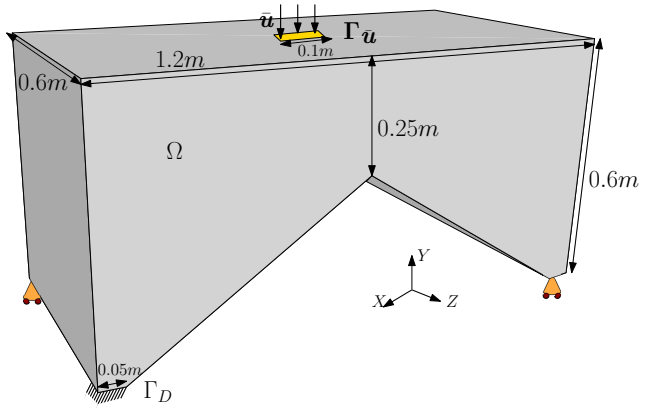
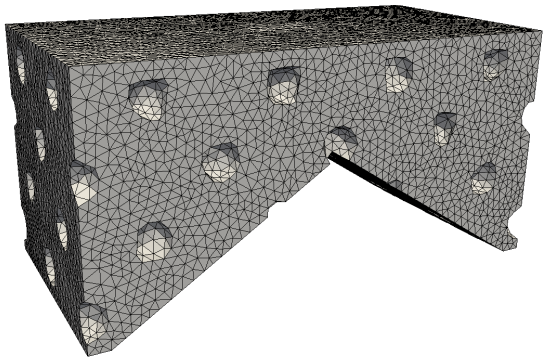


Figure 21: 3D wedge boundary conditions

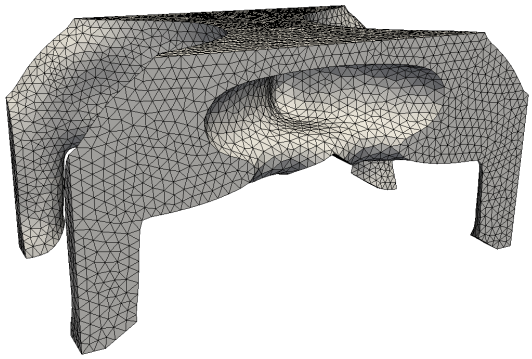
The target volume is chosen to be $0.7m^3$. The initial shape of Fig. 22a converges to the shape of Fig. 22b for linear elasticity. For the damage model, the same initial shape converges to the shape of Fig. 22c for the displacement (58) and to the shape of Fig. 22d for the displacement (59). The corresponding convergence histories are plotted in Fig. 23. It is remarkable that the shapes

in Fig.22c and in Fig.22d are very different, although the imposed displacement (58) and (59) (for which they were optimized) are quite close. This illuminates the highly non-linear nature of the damage model (18).

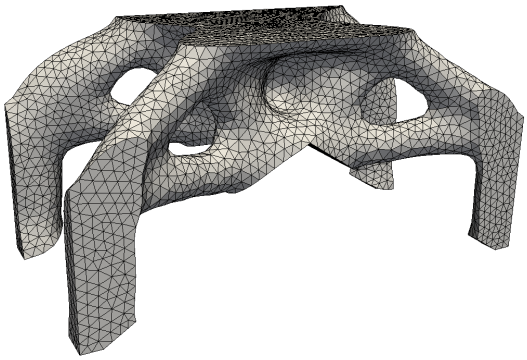
The damage variable α for the intermediate shapes corresponding to the optimal shape in Fig. 22d are plotted in Fig.24. There, one can check that the intermediate shapes undergo crack, not only at the notch at the center, but also at other places. As expected, the resulting final shape in Fig. 59 is crack-free.



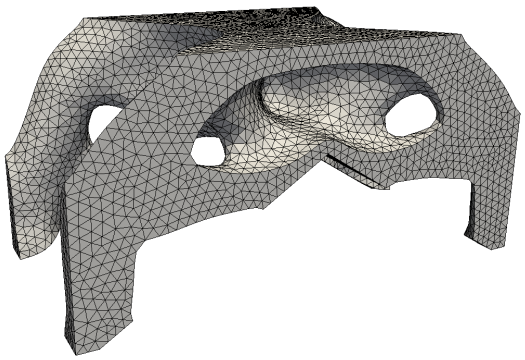
(a) Linear elasticity



(b) Final shape obtained for linear elasticity



(c) Final shape for imposed displacement (58)



(d) Final shape for imposed displacement (59)

Figure 22: Initial and final meshes for wedge shapes

5.8 Coarse 3D Column

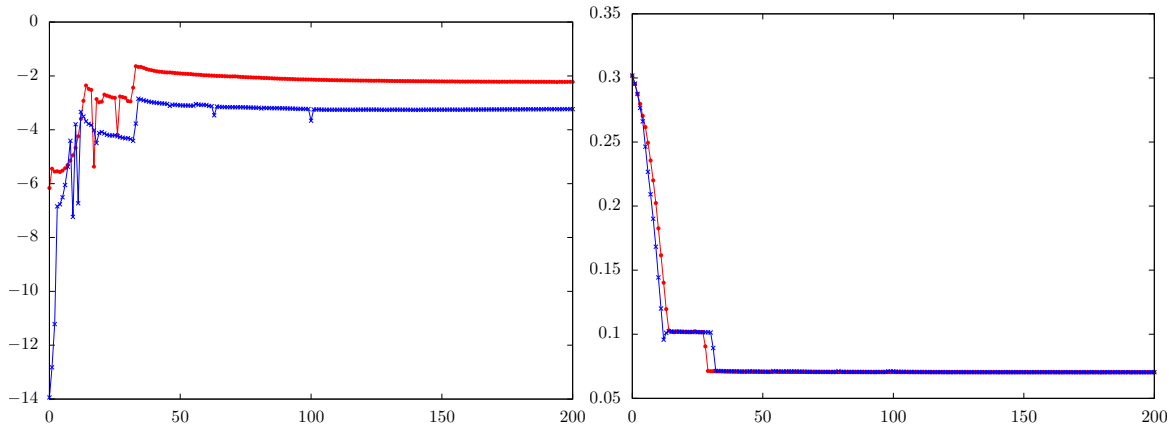
This new test case is a 3D column (see Fig. 25a) which is $4m$ high, fixed at the bottom (in red) and subjected to an imposed displacement $\bar{\boldsymbol{u}}(t)$ on the top (in yellow). The precise geometrical definition of this column can be found in [33].

Here again, we investigate the impact of increasing the magnitude of the imposed displacement on the optimized shape and hence consider two functions

$$\bar{\boldsymbol{u}}(t) = (0, 2.88 \min(1.2t, 1)) \times 10^{-4} m, \quad t \in [0,1]s, \quad \text{and} \tag{60}$$

$$\bar{\boldsymbol{u}}(t) = (0, 2.97 \min(1.2t, 1)) \times 10^{-4} m, \quad t \in [0,1]s. \tag{61}$$

The input parameters of the remesher `MMG` are $h_{\min} = 0.02m$ and $h_{\max} = 0.04m$. Since the characteristic length (using the formula (37)) is $\ell = 0.0375m$, the condition (38) is violated. Instead of choosing a smaller $h_{\max}$, we increase the characteristic length to $\ell = 0.075m$ to ensure that the condition (38) is satisfied. The target volume is $2.5m^3$. The shape is initialized as shown in Fig. 25b. The final shape obtained for linear elasticity is plotted in Fig. 26a. For the damage model, it is plotted in Fig. 26b and 26c for the imposed displacement (60) and (61), respectively. Clearly



(a) Total elastic energy (49) v/s iterations

(b) Volume v/s iterations

Figure 23: Convergence history for the shapes (22c) (in red) and (22d)(in blue)

again, we see that a slight increase in the imposed displacement results in a very different optimized shape for the damage model.

We compare the performance of the three shapes in Fig. 26 for the same damage model and for the same linear elasticity system with the imposed displacement (61) and plot the objective function (49) values in Table 1. As can be expected, the shapes optimized for damage have much better performances with the damage model. But, surprisingly, the performance of the three shapes for linearized elasticity are very similar (the optimal shape for linear elasticity is less than a fraction of percent better than the two other ones). The damage variable α for those three shapes is plotted in Fig. 29. As one can check in Fig. 29, the shapes in Fig. 26b and 26c do not undergo a crack whereas the shape in Fig.26a does.

	Fig. 26a	Fig. 26b	Fig. 26c
Objective (49) for linear elasticity	-183.47	-183.65	-186.83
Objective (49) for damage model	-179.75	-186.49	-186.13

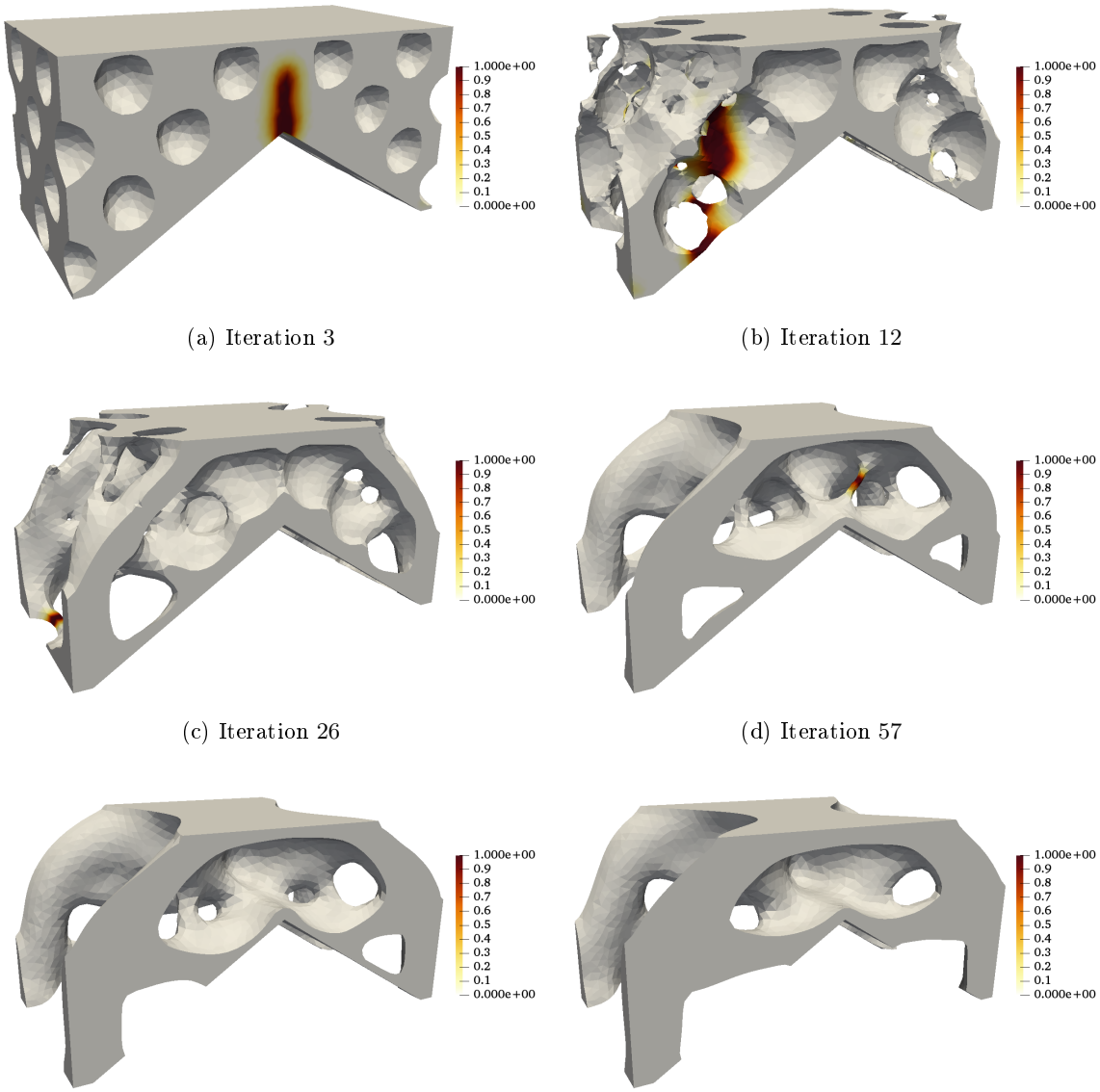
Table 1: Comparison between shapes in Fig. 26a and 26c for the imposed displacement (61)

5.9 Fine 3D Column

Eventually, we revisit the same column, as in the previous subsection, but with a highly refined mesh. The goal of this example is to show that our optimization approach is amenable to high performance computing (HPC). The details on the parallel implementation are given in Subsection 4.6. The column is subjected to the same imposed displacement (61). The input parameters of the remesher `MMG` are $h_{\min} = 0.015m$ and $h_{\max} = 0.03m$. It implies that the initial mesh has 243,641 vertices (1,359,805 tetrahedra) and the final mesh has 241,852 vertices (1,365,125 tetrahedra). The characteristic length, given by (37), is $\ell = 0.0375m$, satisfying condition (38). The target volume is again chosen to be $2.5m^3$. An incremental residual strategy is used, where the residual stiffness is defined with (53), and the residual stiffness parameter is defined as

$$\kappa = \frac{1}{2}\left(1 - \frac{i}{\mathcal{N}}\right)^8,$$

where i is the iteration number and $\mathcal{N}$ is the total number of iterations. This formula differs from (54) by a factor of 5, heuristically found to yield a smoother convergence. This incremental approach is preferred in order to ensure that the intermediate shapes do not undergo brittle fracture too often. The mesh is so fine that the time or descent step τ, given by (45), is very small and the convergence is too slow. Therefore, for this test case, we multiply it by a factor of 2, namely $\tau = h_{\min}/C_i$. The initial shape (with a rich topology) can be seen in Fig. 30a. The final shape obtained for linear



(a) Iteration 3

(b) Iteration 12

(c) Iteration 26

(d) Iteration 57

(e) Iteration 90

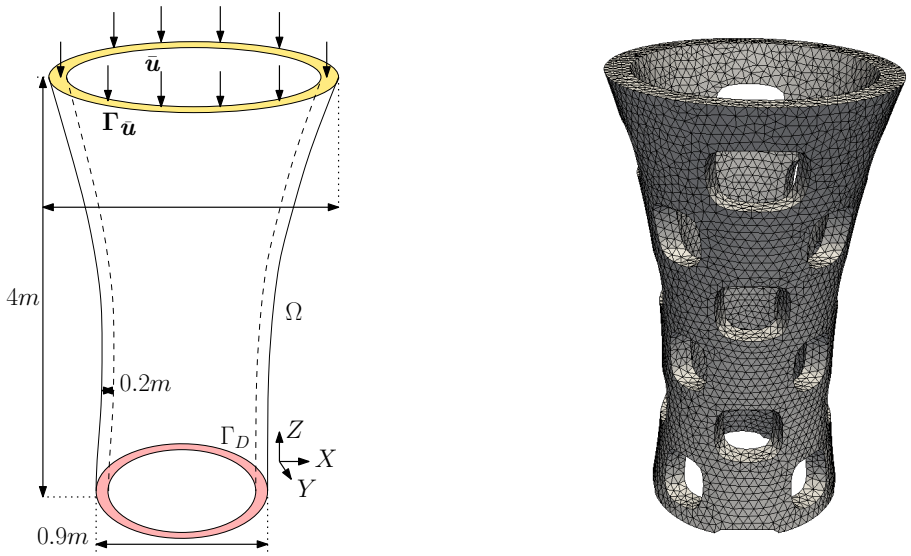
(f) Iteration 300

Figure 24: Damage variable α at the final time T plotted for several optimization iterations for the shape in Fig. 26b

elasticity is plotted in Fig. 30b and, for the damage model, is plotted in Fig. 30c. The damage variable α is plotted for some intermediate shapes in Fig.31. Actually, there were only 3 intermediate shapes, which were cracked. Our incremental approach of the residual stiffness was thus not able to eliminate completely the appearance of cracks during the optimization process. The final shape obtained is very similar to the one obtained for linear elasticity in Fig.30b. The total computational time for this optimization was 7 days.

Acknowledgements

The research of J. Desai has been carried out in the framework of IRT SystemX, Paris-Saclay, France, and therefore granted with public funds within the scope of "Programme d'Investissements d'Avenir". The research of G. Allaire is partly supported by the ANR Shapo project. The authors thank Chiara Nardoni from IRT SystemX for her precious support and fruitful discussions on the



(a) (b)

Figure 25: 3D column: boundary conditions (left), initial shape (right)



(a) Linear elasticity (b) Imposed displacement (60) (c) Imposed displacement (61)

Figure 26: Final shapes obtained from the initialization in Fig. 25b for different models and imposed displacements

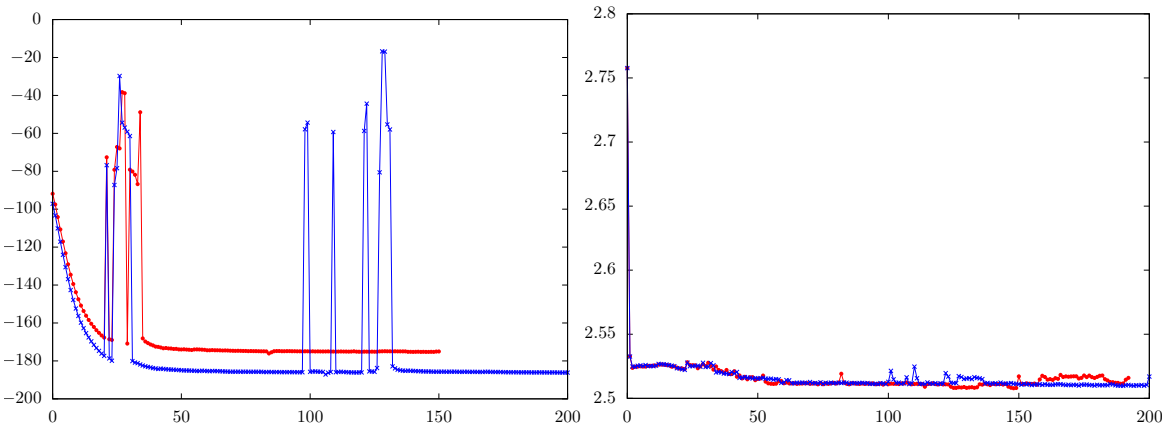
topic of this paper.

References

[1] G. Allaire, C. Dapogny, R. Estevez, A. Faure, and G. Michailidis. Structural optimization under overhang constraints imposed by additive manufacturing technologies. *Journal of Computational Physics*, 351:295–328, 2017.

[2] G. Allaire, C. Dapogny, and F. Jouve. *Shape and topology optimization*, volume 22 of *Handbook of Numerical Analysis*. Elsevier, 2021. in Geometric partial differential equations, part II, A. Bonito and R. Nochetto eds., pp.1-132.

[3] G. Allaire, F. Jouve, and A.M. Toader. A level-set method for shape optimization. *Comptes Rendus Mathématique*, 334(12):1125–1130, 2002.



(a) Objective function (49) v/s iterations

(b) Volume v/s iterations

Figure 27: Convergence history for the shapes in Fig. 26b (in red) and 26c (in blue)

[4] G. Allaire, F. Jouve, and A.M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

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

[6] L. Ambrosio and V.M. Tortorelli. Approximation of functional depending on jumps by elliptic functional via t-convergence. *Communications on Pure and Applied Mathematics*, 43(8):999–1036, 1990.

[7] L. Ambrosio and V.M. Tortorelli. On the approximation of free discontinuity problems. *Bollettino dell'unione matematica italiana*, 6:105–123, 1992.

[8] O. Amir. A topology optimization procedure for reinforced concrete structures. *Computers & Structures*, 114:46–58, 2013.

[9] H. Amor, J.J. Marigo, and C. Maurini. Regularized formulation of the variational brittle fracture with unilateral contact: Numerical experiments. *Journal of the Mechanics and Physics of Solids*, 57:1209–1229, 2009.

[10] S. Balay, S. Abhyankar, M. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. Gropp, D. Karpeyev, D. Kaushik, M. Knepley, D. May, L. Curfman McInnes, R. Mills, T. Munson, K. Rupp, P. Sanan, B. Smith, S. Zampini, H. Zhang, and H. Zhang. Petsc users manual. Technical report, 2019.

[11] C. Barbarosie and S. Lopes. A generalized notion of compliance. *Comptes Rendus Mécanique*, 339(10):641–648, 2011.

[12] P. Bažant and G. Pijaudier-Cabot. Measurement of characteristic length of nonlocal continuum. *Journal of Engineering Mechanics*, 115(4):755–767, 1989.

[13] A. Benallal and J.J. Marigo. Bifurcation and stability issues in gradient theories with softening. *Modelling and Simulation in Materials Science and Engineering*, 15(1):S283, 2006.

[14] M. Bendsoe and O. Sigmund. *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2013.

[15] B. Bourdin. Numerical implementation of the variational formulation for quasi-static brittle fracture. *Interfaces and free boundaries*, 9(3):411–430, 2007.

[16] B. Bourdin, G.A. Francfort, and J.J. Marigo. The variational approach to fracture. *Journal of elasticity*, 91(1-3):5–148, 2008.



(a) Iteration 0 (b) Iteration 21 (c) Iteration 23

(d) Iteration 25 (e) Iteration 60 (f) Iteration 139

Figure 28: Damage variable α at the final time T plotted for several optimization iterations for the shape in Fig. 26b

[17] A. Braides. *Approximation of free-discontinuity problems*. Number 1694. Springer Science & Business Media, 1998.

[18] C. Bui, C. Dapogny, and P. Frey. An accurate anisotropic adaptation method for solving the level set advection equation. *International Journal for Numerical Methods in Fluids*, 70(7):899–922, 2012.

[19] M. Burger. A framework for the construction of level set methods for shape optimization and reconstruction. *Interfaces and Free boundaries*, 5(3):301–329, 2003.

[20] A. Chambolle. An approximation result for special functions with bounded deformation. *Journal de mathématiques pures et appliquées*, 83(7):929–954, 2004.

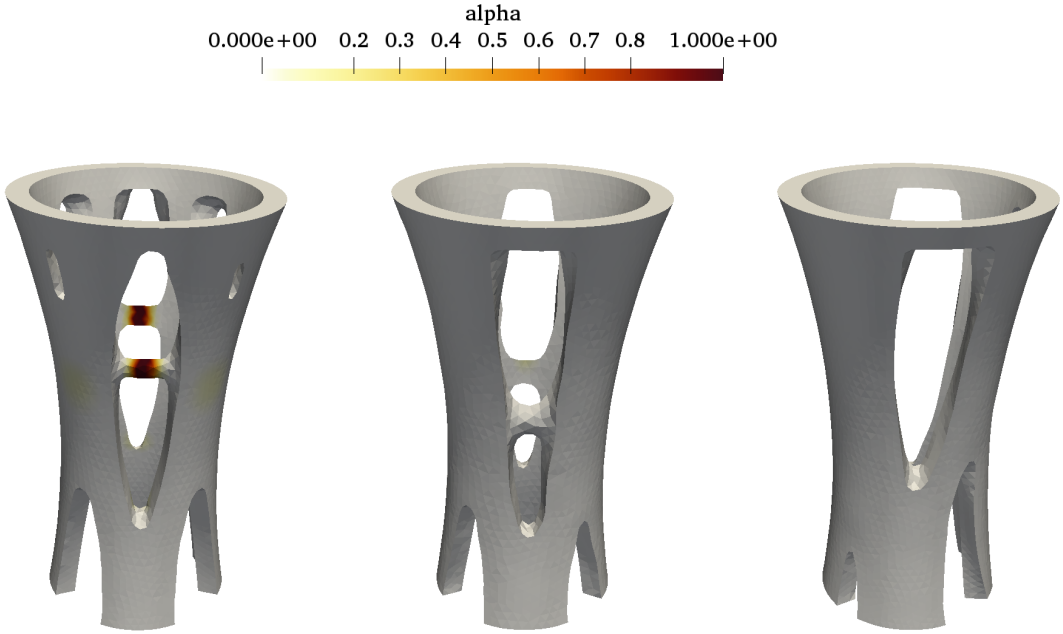


Figure 29: Damage variable α at the final time T for the optimized shapes in Fig. 26 and the imposed displacement (61)

[21] A. Chambolle, G.A. Francfort, and J.J. Marigo. When and how do cracks propagate? *Journal of the Mechanics and Physics of Solids*, 57(9):1614–1622, 2009.

[22] A. Chambolle, G.A. Francfort, and J.J. Marigo. Revisiting energy release rates in brittle fracture. *Journal of Nonlinear Science*, 20(4):395–424, 2010.

[23] Antonin Chambolle, Alessandro Giacomini, and Marcello Ponsiglione. Crack initiation in brittle materials. *Arch. Ration. Mech. Anal.*, 188(2):309–349, 2008.

[24] C. Comi and U. Perego. Fracture energy based bi-dissipative damage model for concrete. *International journal of solids and structures*, 38(36-37):6427–6454, 2001.

[25] J. Céa. Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût. *ESAIM: Mathematical Modelling and Numerical Analysis*, 20(3):371–402, 1986.

[26] D. Da and J. Yvonnet. Topology optimization for maximizing the fracture resistance of periodic quasi-brittle composites structures. *Materials*, 13(15):3279, 2020.

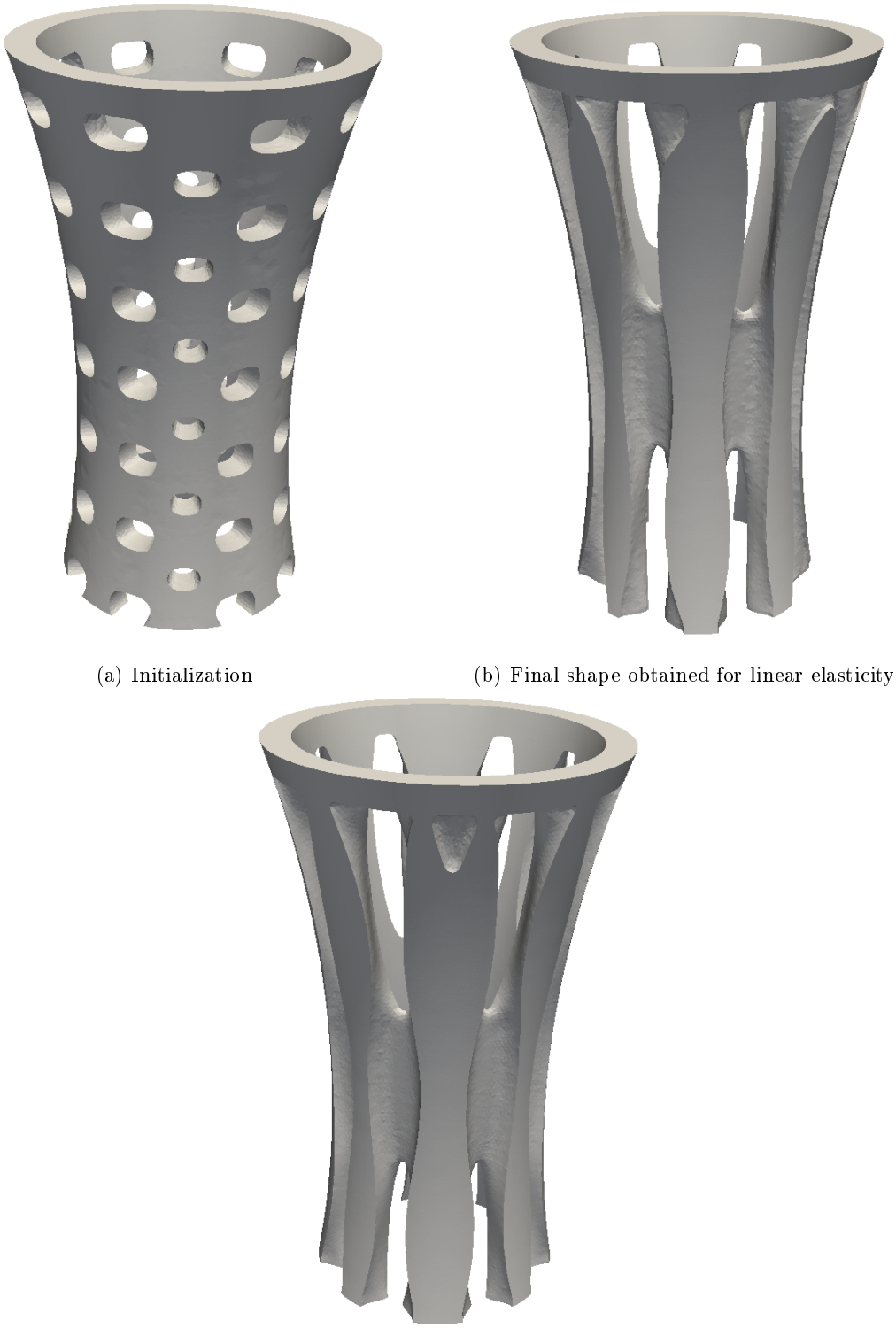
[27] C. Dapogny, C. Dobrzynski, and P. Frey. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics*, 262:358–378, 2014.

[28] C. Dapogny and P. Frey. Computation of the signed distance function to a discrete contour on adapted triangulation. *Calcolo*, 49(3):193–219, 2012.

[29] J. Desai. *Topology optimization in contact, plasticity, and fracture mechanics using a level-set method*. Phd thesis, to appear, Université de Paris, 2021.

[30] J. Desai, G. Allaire, and F. Jouve. Topology optimization in plasticity with hardening using a level-set method. *Structural and Multidisciplinary Optimization (to appear)*, 2021.

[31] F. Feppon, G. Allaire, C. Dapogny, and P. Jolivet. Topology optimization of thermal fluid–structure systems using body-fitted meshes and parallel computing. *Journal of Computational Physics*, 417:109574, 2020.



(a) Initialization

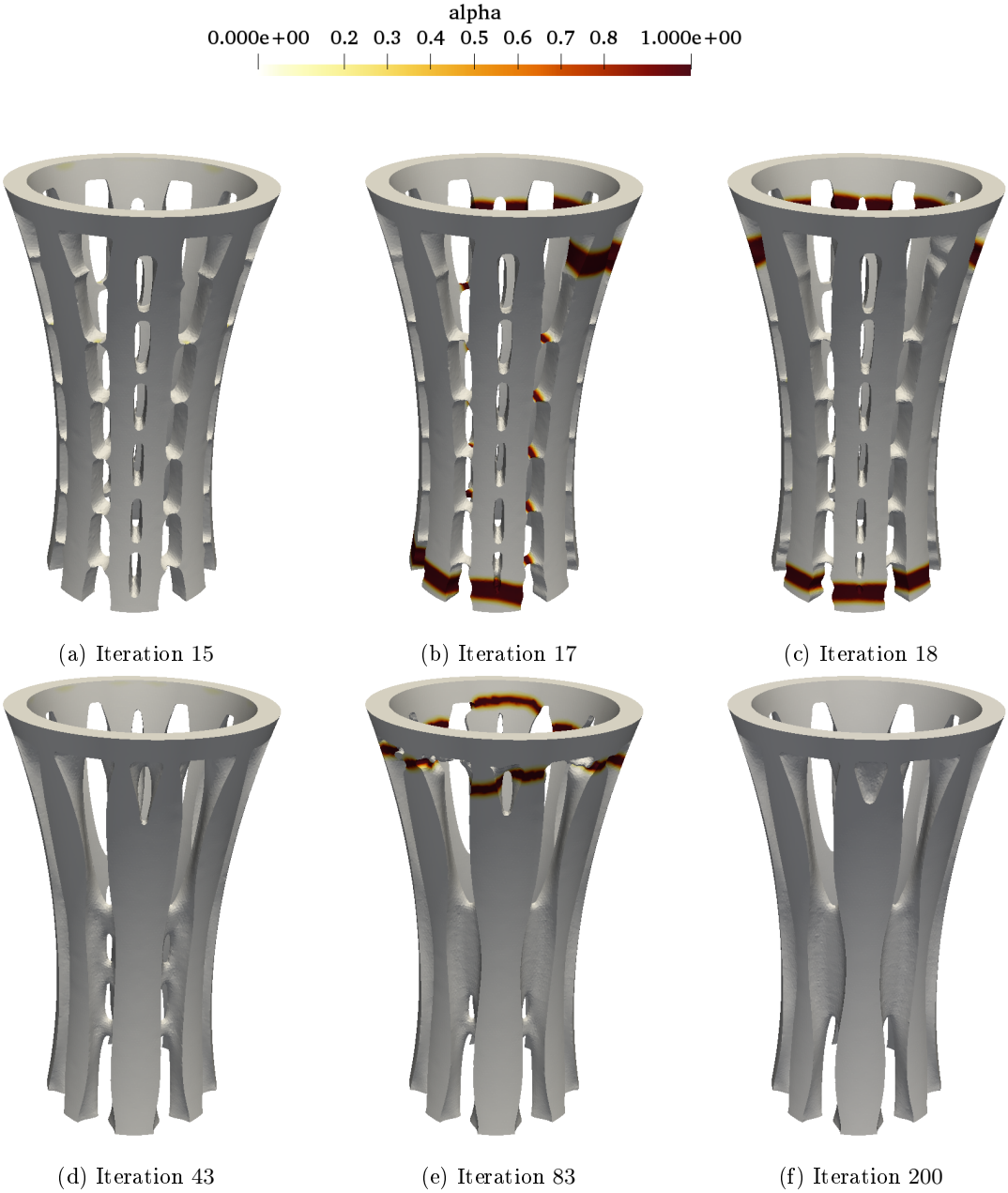
(b) Final shape obtained for linear elasticity

(c) Final shape obtained for damage model

Figure 30: Column shapes for a fine mesh and the imposed displacement (61)

[32] G.A. Francfort and J.J. Marigo. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids*, 46(8):1319–1342, 1998.

[33] N. Gaudillière, R. Duballet, C. Bouyssou, A. Mallet, Ph. Roux, M. Zakeri, and J. Dirrenberger. Chapter 3 - building applications using lost formworks obtained through large-scale additive



(a) Iteration 15 (b) Iteration 17 (c) Iteration 18

(d) Iteration 43 (e) Iteration 83 (f) Iteration 200

Figure 31: Damage variable α at the final time T plotted for several optimization iterations for the shape in Fig. 30c

manufacturing of ultra-high-performance concrete. In J.G. Sanjayan, A. Nazari, and B. Nematollahi, editors, *3D Concrete Printing Technology*, pages 37–58. Butterworth-Heinemann, 2019.

[34] A. Giacomini. Ambrosio-Tortorelli approximation of quasi-static evolution of brittle fractures. *Calculus of Variations and Partial Differential Equations*, 22(2):129–172, 2005.

[35] A.A. Griffith. VI. the phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 221(582-593):163–198, 1921.

[36] F. Hecht. New development in FreeFEM++. *Journal of numerical mathematics*, 20(3-4):251–266, 2012.

[37] A. Henrot and M. Pierre. *Shape Variation and Optimization*. EMS Tracts in Mathematics Vol. 28, 2018.

[38] K.A. James and H. Waisman. Topology optimization of structures under variable loading using a damage superposition approach. *International Journal for Numerical Methods in Engineering*, 101(5):375–406, 2015.

[39] P. Jolivet. FreeFEM tutorial. `http://jolivet.perso.enseeiht.fr/FreeFem-tutorial/`, 2020.

[40] Z. Kang, P. Liu, and M. Li. Topology optimization considering fracture mechanics behaviors at specified locations. *Structural and Multidisciplinary Optimization*, 55(5):1847–1864, 2017.

[41] G. Karypis and V. Kumar. Metis: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. 1997.

[42] J.J. Marigo. Constitutive relations in plasticity, damage and fracture mechanics based on a work property. *Nuclear Engineering and Design*, 114(3):249–272, 1989.

[43] C. Miehe, F. Welschinger, and M. Hofacker. Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field fe implementations. *International Journal for Numerical Methods in Engineering*, 83(10):1273–1311, 2010.

[44] F. Mignot and J.P. Puel. Optimal control in some variational inequalities. *SIAM Journal on Control and Optimization*, 22(3):466–476, 1984.

[45] D.B. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 1989.

[46] Q.S. Nguyen. Bifurcation and post-bifurcation analysis in plasticity and brittle fracture. *Journal of the Mechanics and Physics of Solids*, 35(3):303–324, 1987.

[47] S. Osher and R. Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153. Springer Science & Business Media, 2006.

[48] K. Pham, H. Amor, J.J. Marigo, and C. Maurini. Gradient damage models and their use to approximate brittle fracture. *International Journal of Damage Mechanics*, 20(4):618–652, 2011.

[49] J. Sokolowski and J.P. Zolésio. *Introduction to shape optimization*. Springer, 1992.

[50] E. Tanné, T. Li, B. Bourdin, J.J. Marigo, and C. Maurini. Crack nucleation in variational phase-field models of brittle fracture. *Journal of the Mechanics and Physics of Solids*, 110:80–99, 2018.

[51] M. Wang, X. Wang, and D. Guo. A level set method for structural topology optimization. *Computer methods in applied mechanics and engineering*, 192(1-2):227–246, 2003.

[52] C. Wu, J. Fang, S. Zhou, Z. Zhang, G. Sun, G.P. Steven, and Q. Li. Level-set topology optimization for maximizing fracture resistance of brittle materials using phase-field fracture model. *International Journal for Numerical Methods in Engineering*, 121(13):2929–2945, 2020.