Robust topology optimization considering part distortion and process variability in additive manufacturing

Komini, Ludian; Langelaar, Matthijs; Kriegesmann, Benedikt

DOI
10.1016/j.advengsoft.2023.103551

Publication date
2023

Document Version
Final published version

Published in
Advances in Engineering Software

Citation (APA)
Komini, L., Langelaar, M., & Kriegesmann, B. (2023). Robust topology optimization considering part distortion and process variability in additive manufacturing. *Advances in Engineering Software*, *186*, Article 103551. https://doi.org/10.1016/j.advengsoft.2023.103551

Advances in Engineering Software 186 (2023) 103551

Contents lists available at ScienceDirect

Advances in Engineering Software

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

Robust topology optimization considering part distortion and process variability in additive manufacturing

Ludian Komini [a], Matthijs Langelaar [b], Benedikt Kriegesmann [c,*]

[a] TBF Marti AG, Sernftalstrasse 2, 8762 Schwanden, Switzerland
[b] TU Delft, Mekelweg 2, 2628 CD Delft, The Netherlands
[c] Hamburg University of Technology, Am Schwarzenberg-Campus 4, Hamburg, Germany

ARTICLE INFO

Keywords:
Topology optimization
Additive manufacturing simulation
Part distortion
Uncertainty quantification
Robust design

ABSTRACT

This paper presents a method to consider uncertainties in the distortion prediction of additive manufacturing processes within robust topology optimization. The random variable of the stochastic additive manufacturing process is the inherent thermomechanical strain, typically determined by process characterization experiments. The value of the inherent strain *per se* encompasses uncertainty due to differences between characterization and production geometries, uncontrolled process variations, hatching pattern choices, and other effects not captured in layer-by-layer part-scale additive manufacturing simulation. These effects can be represented by different inherent strains for each realized layer, therefore, model the variability in part distortion. Instead of employing a more detailed simulation approach, requiring significantly more process data and computation time, our method aims at generating a robust design in this setting, to obtain parts exhibiting reduced distortion regardless of uncertainties in distortion prediction. The formulation benefits from the superposition potential within the employed process simulation. For robust optimization, the expected part distortion and its estimated variance are included in the standard density-based topology optimization algorithm's objective function. The effectiveness of the approach is demonstrated by simultaneously optimizing structural performance combined with a minimized additive manufacturing part distortion under uncertain process conditions.

1. Introduction

Topology Optimization (TO) and Additive Manufacturing (AM) have achieved an unprecedented scale of application in industry and academia in the last three decades. On the one hand, TO alone has reached a mature status as an efficient and versatile computational design process. On the other hand, AM has established itself as a highly flexible and economically viable fabrication alternative for end-use parts. The advantages of AM have boosted the production of complex topology-optimized parts, which would otherwise be impossible to produce using classical methods [1]. These developments have encouraged researchers to explore strategic techniques to bring these two distinct disciplines together [2].

Considering the challenges of AM fabrication, extensions of TO procedures to reduce support requirements through control of overhanging regions have received the most attention (see, e.g. [3–5]). Proposed methods typically only involve geometric criteria of the design that can be evaluated at a low computational cost. However, for a variety of potentially fatal defects (e.g., overheating, distortion, recoater collision predictions, etc.), a computationally demanding AM process simulation is essential. Nonetheless, the physical simulation of the AM process alone has achieved significant improvement in the context of distortion prediction, allowing practical part-scale simulations ([6]). Additionally, several recent studies have proposed incorporating AM process simulation into TO framework (see, e.g., [7–11] and the references therein).

Another important fact to highlight, in this regard, is that considerable variability of the distortion prediction is attributed to the presence of uncertainties in the AM process parameters. The comparison results in [12] reveal that deterministic models are not trustworthy for practical usages due to large variations in the prediction of the distortion during the AM process simulation. This fact indicates a clear need for a framework for the robust optimization of parts, given the uncertainty present in the AM parameters. The consideration of uncertainties within a design optimization can be carried out in various ways [13], for instance, by treating the uncertain parameters as stochastic variables (probabilistic approach) or as bounded parameters (see, e.g., [14,15]). Using probabilistic approaches and targeting to reduce the variability of some response functions is referred to as robust design optimization

* Corresponding author.
E-mail addresses: ludian.komini@tbfmartiag.ch (L. Komini), benedikt.kriegesmann@tuhh.de (B. Kriegesmann).

https://doi.org/10.1016/j.advengsoft.2023.103551
Received 8 May 2023; Received in revised form 1 September 2023; Accepted 18 October 2023
Available online 4 November 2023

and, in combination with topology optimization, as robust topology optimization (RTO) (see, e.g., [16–18]). Many works consider the applied load (direction) as a random variable [19,20], and some also study material stiffness properties and/or geometric deviations [16–18]. In a few studies, manufacturing simulations alone have been carried out probabilistically (e.g., [21]). Still, AM manufacturing variables have thus far yet to be embedded into a robust topology optimization.

In this paper, we address this gap by considering uncertainties in the inherent-strain-based mechanical AM process simulation and embed this into a robust topology optimization. The considered uncertainties arise due to differences between inherent strain characterization experiments and the actual part geometry, uncontrolled process variations, hatching pattern choices, and other physical effects not captured in the layer-by-layer AM simulation. In this study, we assume that the aggregation of these effects can be represented by different inherent strains (IS) for each realized layer, which can be combined by the employed layer-by-layer process simulation. From the point of view of the simulation process, the magnitude of the layer-wise inherent strains is treated as stochastic. Consequently, the outcome of the AM simulation is a distribution of distortion fields. We focus on a single aggregated distortion measure for simplicity but without loss of generality. An example of layer-wise distortion control to prevent recoater collisions in TO can be found in [7,11]. In this paper, we consider a multiobjective minimization problem, combining both structural compliance and global distortion using the weighted sum approach. The distortion minimization ensures the compatibility and functionality of the part with its surroundings. Following [6], *mutatis mutandis*, the AM process simulation under linearity assumptions is broken down into decoupled process steps, each of which comprises the simulation of one layer addition. In this setting, the principle of superposition clears the way to leverage parallelization while preserving the accuracy cf. [6].

Our contribution is a novel stochastic Design for Additive Manufacturing (DfAM) framework. We propose an effective incorporation of the uncertainty in the selected AM process parameter, a fast stochastic AM process simulation, and its integration into the established density-based TO framework.

We use the p-mean of the displacement field to measure the distortion quantity resulting from the AM process simulation. Further, we propose, implement and validate three formulations for the stochastic distortion quantity:

1. *Monte Carlo estimation with superimposed realizations and sensitivity, viz. MC* ($p = 2$).
2. *Monte Carlo estimation with superimposed realizations, viz. MC* ($p > 2$).
3. *First-Order-Second-Moment estimation of the statistical quantities and sensitivity, viz. (FOSM).*

The first two approaches presented in this work are based on the Monte Carlo simulation of all the realizations, and they benefit from the computational breakdown coming from the superposition exploited in the AM process simulation. The difference between the first and the second formulation is that when $p = 2$, we can also superimpose each sensitivity calculation, thus further reducing the algorithm's overall computational cost.

The last formulation of the stochastic process, namely *FOSM*, is based on the Taylor series expansion of the statistical quantities, mean value and standard deviation, around the mean value of the process parameter. In this way, in a single loop, i.e., for only one realization we determine the mean value, the variance of the distortion quantity and their sensitivity [18].

The different formulations are studied using 2D numerical examples, since, in spite of the parallelization of the AM distortion prediction, computational cost remains high. Nevertheless, no fundamental obstacle prevents the extension of the proposed approaches to 3D.

This paper is organized as follows. First, we outline the problem definition in Section 2. Section 2.3 provides the fundamentals of the AM simulation method used in this work. Section 2.4 explains how an aggregate solver for distortion prediction is constructed by assembling independent solutions for each layer partition of the entire model. Subsequently, Section 2.5 outlines how this concept extends to multiple realizations of the process simulation and the associated robust optimization problem is formulated. Section 3 contains all the derivations of the stochastic quantities and their subsequent sensitivities. In Section 4 these quantities are arranged in an algorithm and its performance is studied using several examples in Section 5. Ultimately, Section 6 concludes this paper. All additional derivations can be found in Appendices A and D.

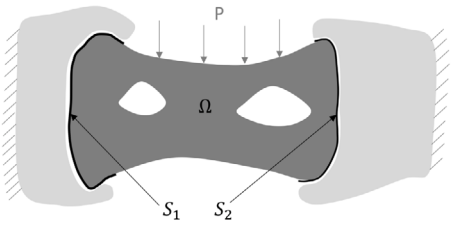

Fig. 1. Illustration of the considered problem setting. The solid gray color Domain Ω indicates the part that needs to be optimized and 3D-printed, while the light gray regions indicate the parts that need to be compatible. The optimized part's interfaces (S_1 and S_2) need to remain compatible with the adjacent parts.

2. Problem setting

2.1. Deterministic formulation

Our goal is to build a new DfAM framework, which will facilitate the study of the robust design of a topology-optimized structure, minimizing stochastic AM process part-distortion, see Fig. 1. Let us, as a starting point, consider a representative problem, the minimization of the compliance C for a structural part.

$$\min_{\boldsymbol{x}} C(\boldsymbol{x}) = \boldsymbol{u}_o(\boldsymbol{x})^T \boldsymbol{K}(\boldsymbol{x}) \boldsymbol{u}_o(\boldsymbol{x}) \tag{1}$$

s.t.:

$$\frac{\boldsymbol{V}(\boldsymbol{x})}{\boldsymbol{V}_0} \leq v_{frac}$$

$$\boldsymbol{K}(\boldsymbol{x})\boldsymbol{u}_o(\boldsymbol{x}) = \boldsymbol{p}_l \tag{2}$$

$$0 \leq x_e \leq 1 \qquad \text{for all } x_e \text{ in } \boldsymbol{x}$$

Here x_e represents the elements' density and is a component of the vector $\boldsymbol{x}$, which comprises the elements' density of the whole design domain, v_{frac} is the maximum volume fraction, $\boldsymbol{u}_o$ is the displacement field from the operation, $\boldsymbol{K}$ is the global stiffness matrix, $\boldsymbol{V}(\boldsymbol{x})$ is the volume of the ultimate design layout, $\boldsymbol{V}(0)$ is the initial volume of the design domain and $\boldsymbol{p}_l$ is the external load vector. In this paper, the minimization of the compliance of 3D-printed components is considered.

The SIMP interpolation given in Eq. (3) has been used for the parametrization of the discretized design domain [22].

$$E_e(x_e) = E_{min} + x_e^{p_e}(E_0 - E_{min}). \tag{3}$$

For a component to fit in an assembly, see Fig. 1, it is essential to limit its distortion due to the printing process, particularly at part–part interfaces. Thus, while this part is additively manufactured, we also want that it features minimum distortion. We name this distortion quantity Γ. The distortion effect is not considered in computing the compliance as it is small enough to be negligible for this matter, but big enough to compromise the compatibility of the part with its adjacent elements. This yields a multi-objective problem, which is scalarized by weighted summation with weight κ_1 as shown in Eq. (4). Here, the compliance C is divided by the applied load in order to scale it to

the same order of magnitude as the distortion measure Γ. We refer to this case as deterministic distortion minimization, or in short, the deterministic case.

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \kappa_1 C(\boldsymbol{x}) + (1-\kappa_1)\Gamma(\boldsymbol{x}) \tag{4}$$

s.t:

$$\begin{aligned} &\frac{\boldsymbol{V}(\boldsymbol{x})}{\boldsymbol{V}_0} = v_{frac} \\ &\boldsymbol{K}(\boldsymbol{x})\boldsymbol{u}_o(\boldsymbol{x}) = \boldsymbol{p}_l \\ &0 \le x_e \le 1 \qquad \text{for all } x_e \text{ in } \boldsymbol{x} \end{aligned} \tag{5}$$

$$\begin{aligned} &\boldsymbol{K}^k(\boldsymbol{x})\boldsymbol{u}_j^k(\boldsymbol{x}) = f_j^k(\boldsymbol{x}) \\ &k = 1...N \\ &j = 1...m \end{aligned} \tag{6}$$

where:

k	index of the layer.
N	total number of layers.
j	index of the realization.
m	total number of realizations.

Here Eq. (6) represents the layer-by-layer AM distortion simulation following the linearized inherent strain approach [6], involving k steps which we discuss in Section 2.3. Note that in this paper, no Einstein summation convention is used for repeated indices unless explicitly mentioned.

2.2. Stochastic formulation

Using a probabilistic formulation, we acknowledge the fact that the AM process simulation comprises uncertainty, which will propagate to the estimation of the distortion quantity Γ. To this end, we introduce an integrated stochastic AM process simulation in Section 2.3.

There are numerous applications in which it is desirable that a topology-optimized structure is robust with respect to AM-induced distortion. In the present problem setting, we assume that the magnitude of the inherent strain is uncertain. However, based on expert knowledge and experience, a distribution of the stochastic magnitude parameter can be provided. Furthermore, the stochastic parameter, i.e., the magnitude of the inherent strain, is assumed to vary per layer, which corresponds to the finest resolution of the employed AM process model.

To this end, we introduce the nominal inherent strain $\Delta\epsilon^k$, as well as a stochastic scaling factors α^k of the k-th process step, where the scaling factors α^{k_1} and α^{k_2} of different process steps k_1 and k_2 may correlate. Hence, the vector $\boldsymbol{\alpha}$ is the random vector containing the correlated random factors for all steps. This random vector can be considered as a discretized one-dimensional random field in the build direction. The approach implies the assumption that the scaling factor α^k is constant in each process step k. This is certainly a strong simplification, but this assumption is a prerequisite for the approaches proposed in this paper and consistent with layer-by-layer modeling.

Under these considerations, following, e.g. [18], we define the subsequent constrained stochastic optimization problem with the objective function $f(\boldsymbol{x}, \boldsymbol{\alpha})$ as the weighted summation of the compliance and a term composed of the mean and standard deviation, of the distortion quantity, see Eq. (7).

$$\min_{\boldsymbol{x}} \kappa_1 C(\boldsymbol{x}) + (1-\kappa_1)\left[\kappa_2 \mu_\Gamma(\boldsymbol{x}) + (1-\kappa_2)\sigma_\Gamma(\boldsymbol{x})\right] \tag{7}$$

s.t:

$$\begin{aligned} &\frac{\boldsymbol{V}(\boldsymbol{x})}{\boldsymbol{V}_0} = v_{frac} \\ &\boldsymbol{K}(\boldsymbol{x})\boldsymbol{u}_o(\boldsymbol{x}) = \boldsymbol{p}_l \\ &\boldsymbol{K}^k(\boldsymbol{x})\boldsymbol{u}_j^k(\boldsymbol{x}) = f_j^k(\boldsymbol{x}) \\ &k = 1...N \\ &j = 1...m \\ &0 \le x_e \le 1 \qquad \text{for all } x_e \text{ in } \boldsymbol{x} \\ &\mu_\Gamma(\boldsymbol{x}) = E[\Gamma(\boldsymbol{x}, \boldsymbol{\alpha})] \\ &\sigma_\Gamma(\boldsymbol{x}) = \sqrt{E[\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}) - \mu_\Gamma]^2} \\ &\boldsymbol{\alpha} \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2) \end{aligned} \tag{8}$$

The distortion quantity $\Gamma(\boldsymbol{x}, \boldsymbol{\alpha})$ now is a function of the random vector $\boldsymbol{\alpha}$.

In this work, we consider that the compliance $C(\boldsymbol{x})$ is not exposed to uncertainty for the same reason as stated earlier. Ergo, the compliance formulation and its sensitivity are taken the same as in the deterministic compliance minimization problem.

$$\frac{\partial C(\boldsymbol{x})}{\partial \boldsymbol{x}_e} = -p_e x_e^{p_e - 1}(E_0 - E_{min})\boldsymbol{u}^T \mathbf{K}_0 \boldsymbol{u} \tag{9}$$

Here, $\mathbf{K}_0$ is the element stiffness matrix for an element with unit Young's modulus. The new components of the objective function and their respective sensitivity are $\mu_\Gamma(\boldsymbol{x})$, $\sigma_\Gamma(\boldsymbol{x})$, $\frac{\partial \sigma_\Gamma(\boldsymbol{x})}{\partial x_e}$ and $\frac{\partial \mu_\Gamma(\boldsymbol{x})}{\partial x_e}$. The formulation of these quantities, and their algorithmic implementation, is the focus of this work. Since these quantities are a direct outcome of the AM process simulation, we first present in Section 2.3 its mathematical representation, together with the adequate simplifications, which we leverage to facilitate its practical implementation.

2.3. Additive manufacturing process modeling: fundamentals

Using a high-fidelity simulation of a stochastic AM process in an optimization framework requires excessive computational effort. This computation intensity is due to both the temporal and spatial scales involved and the fact that the exact physics of the AM process are governed by nonlinear equations, which need to be solved for each realization at each optimization iteration. In light of this challenge, certain simplifications are introduced, that facilitate an approximate (fast) solution scheme, rather than a strictly accurate (computationally intense) one. A first and common simplification is to consider the printing process as a sequence of layer additions, instead of resolving the detailed scanning sequence used. The driving physical event of the AM process is the heat flow generated during the processing of each layer. This heat flow distributes the thermal energy that the newly added layer contains, to the surrounding environment, i.e., already-built layers and the base plate. High-temperature gradients result in a build-up of thermal stress and, consequently, distortion of the part. This thermo-mechanical process is influenced by many different parameters, such as hatching pattern choices, scan strategy (path and speed), pre-heating temperature, layer thickness, beam size and power, powder size, and morphology. The ultimate effect can be estimated to some extent using a fully coupled transient thermal process simulation. However, this has a formidable computational cost. Moreover, all the influencing parameters encompass uncertainties [23]. It is accepted that not all physical effects correlated to these processes can be simulated precisely [6,23]. To overcome the computational bottleneck and enable consideration of AM process effects within TO, a simplified AM process model is required. The Inherent Strain (IS), representing a compound effect of all the physical events and their uncertainties, has been established as a reasonable compromise between accuracy and computational cost, and has been applied in deterministic TO studies involving AM distortion measures (e.g., [9,11]). In the IS method, the residual plastic strain, is considered at the hatching region when

the new layer is activated. From this strain, the thermal stresses and their consequent effects are computed. We follow the scheme proposed by Munro et al. [6], which exploits a linearized version of the IS modeling approach. Details are provided in the next subsection. The IS is typically obtained from characterization experiments with test geometries [23]. Note that differences between the actual part and test geometry may already result in differences in actual local IS values. This uncertainty is one of the motivations to use a robust optimization approach in conjunction with the IS approach.

The AM process is decoupled into linear sub-processes for each layer without compromising accuracy. Afterward, the superposition of all individual sub-process deformations obtains the final thermo-mechanical deformation. Recognizing that the individual process step deformations scale with IS magnitude, we perform these simulations using nominal IS values. This recognition allows us to construct realizations with different IS magnitudes at low cost simply using the superposition of these nominal process step deformations, scaled appropriately.

This concept forms the core concept of the computational model for stochastic AM process simulation, where the uncertainty of the process model is introduced in the IS magnitude.

2.4. Additive manufacturing process simulation — nominal simulation

In this section, the formulation of the AM process simulation is given. An illustrative structure, presented in Fig. 2, has been partitioned into several sub-models, each of which consists of one lumped layer built during the AM process. Assuming that there is a finite element discretization of the mechanical equation, representing the single process on each sub-model, the AM-like process's simulation follows the element birth and death method. It consists of iteratively activating layers of finite elements in the mesh to simulate the process of incrementally adding solid material to the structure. In this model, the physics involved contains only the IS as an external effect. Furthermore, the elements of every single layer are added simultaneously, and each single layer addition can then be solved separately.

For the implementation of the process described above material, and geometrical, linear assumptions from the spirit of Munro et al. [6] are utilized, with adjustments for efficient stochastic process consideration as discussed in the next subsection.

Nevertheless, only the thermal stress, triggered by the inherent strain activated in the individual hatching regions of the newly added layer, will propagate in the built layers [23]. These considerations, together with linearity assumptions, allow the decoupling of the AM process, which comprises the incremental layer activation. Each layer addition is simulated separately while applying a nominal value for the driving physical parameter, i.e., the inherent strain (IS). We call this the nominal AM process simulation. Hence, under these assumptions, a holistic view of the physical process can be seen as two main sub-steps for each layer activation, see [6].

- The stress-free merging of the newly added layer to the last layer, i.e., the liquid (melted) layer, is placed on the previous sequential layer.
- The addition effect coming from the newly added layer.

Excluding any other external forces during the printing process, the incremental displacement field will be calculated only from the solidification and the contraction of the newly added layer, while considering all the previous layers at the fully solid-state [6].

From an implementation viewpoint, the equilibrium equations are mapped to each decoupled process. Each of them represents the deformation due to the deposition of the $k-$th layer. Following standard finite element procedures (see e.g., [24]), for an ordinary realization, each displacement field $\boldsymbol{u}^k$ of the added layer k is evaluated as follows:

$$\boldsymbol{K}^k(\boldsymbol{x})\boldsymbol{u}^k(\boldsymbol{x}) = \boldsymbol{f}^k(\boldsymbol{x}) \tag{10}$$

$$\boldsymbol{K}^k(\boldsymbol{x}) = \mathop{\mathrm{A}}_{e=1}^{n_e} \boldsymbol{k}_e^k(x_e) \tag{11}$$

$$\boldsymbol{f}^k(\boldsymbol{x}) = \mathop{\mathrm{A}}_{e=1}^{n_e} \boldsymbol{f}_e^k(x_e) \tag{12}$$

where:

$\mathop{\mathrm{A}}_{e=1}^{n_e}$ assembly operation.

$\boldsymbol{k}_e^k(x_e) = \int_V \boldsymbol{B}^T \boldsymbol{D}(x_e)\boldsymbol{B}dv$ element stiffness matrix.

$\boldsymbol{f}_e^k(x_e) = \int_V \boldsymbol{B}^T \boldsymbol{D}_e(x_e)\Delta\boldsymbol{\epsilon}^k dv$ nominal element force vector.

$\boldsymbol{D}_e(x_e) = \boldsymbol{D}_{min} + x_e^{p_e}\left(\boldsymbol{D}_{min} - \boldsymbol{D}_0\right)$ constitutive.

The strain matrix B:

$$\mathbf{B} = \begin{bmatrix} \frac{\partial N_i}{\partial \xi} & 0 & \frac{\partial N_{i+1}}{\partial \xi} & 0 & \frac{\partial N_{i+2}}{\partial \xi} & 0 & \frac{\partial N_{i+3}}{\partial \xi} & 0 \\ 0 & \frac{\partial N_i}{\partial \eta} & 0 & \frac{\partial N_{i+1}}{\partial \eta} & 0 & \frac{\partial N_{i+2}}{\partial \eta} & 0 & \frac{\partial N_{i+3}}{\partial \eta} \\ \frac{\partial N_i}{\partial \eta} & \frac{\partial N_i}{\partial \xi} & \frac{\partial N_{i+1}}{\partial \eta} & \frac{\partial N_{i+1}}{\partial \xi} & \frac{\partial N_{i+2}}{\partial \eta} & \frac{\partial N_{i+2}}{\partial \xi} & \frac{\partial N_{i+3}}{\partial \eta} & \frac{\partial N_{i+3}}{\partial \xi} \end{bmatrix}$$

i is the index of the shape function N_i.

ξ *and* η are the local coordinate in the ξ *and* η direction.

The entire displacement field for N additive manufacturing steps can be obtained by applying the superposition principle:

$$\boldsymbol{u} = \sum_{k=1}^{N} \boldsymbol{u}^k \tag{13}$$

The displacement field is represented in a vector and it expands as the number of layers increases. Therefore, a reference vector of the displacement field will be allocated with the size equal to the design domain's total DOFs. The displacement field increments will be summed up at this reference vector after each layer addition, consistent with their DOFs.

2.5. Additive manufacturing stochastic process

In Section 2.4 we clarify the linear behavior of our formulation underlying each layer addition step, and that the superposition principle holds. This formulation is extended naturally to an arbitrary number of realizations m following the same principle. The key feature of this formulation is that the displacement field of a distinct process step k scales linearly with the scaling factors α^k of the inherent strains.

To account for the variability of the IS, α^k will act as a scaling factor. First, we run the simulation for $j = 0$ and $\alpha_0^k = 1$ (i.e., a vector filled with unity values), which would generate the nominal distortion field $\boldsymbol{u}^k$. In a Monte Carlo simulation with m realization, each $\boldsymbol{u}_j^k$ is obtained from the linear operation between the scaling factor α_j^k and the nominal distortion field $\boldsymbol{u}^k$.

$$\boldsymbol{u}_j^k = \alpha_j^k \boldsymbol{u}^k \quad \text{for } j = 1...m \tag{14}$$

Consecutively, the total displacement field at a given realization is computed as follows.

$$\boldsymbol{u}_j = \sum_{k=1}^{N} \alpha_j^k \boldsymbol{u}^k \quad \text{for } j = 1...m \tag{15}$$

A graphic visualisation of the superposition of the displacement field is given in Fig. 3. Following this principle, each realization can be generated from the nominal AM simulation in Section 2.4. The total displacement field of each realization can be obtained using the superposition introduced.

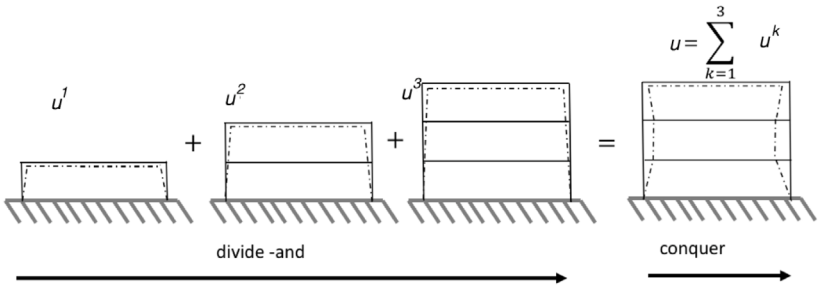

Fig. 2. The decoupling of the layerwise AM-process, illustrated with a representative part (left-hand side) where k indicates the layer number. This separation scheme is made for the nominal simulation. The effect of each layer deposition step is superimposed on the final AM part (right-hand side).

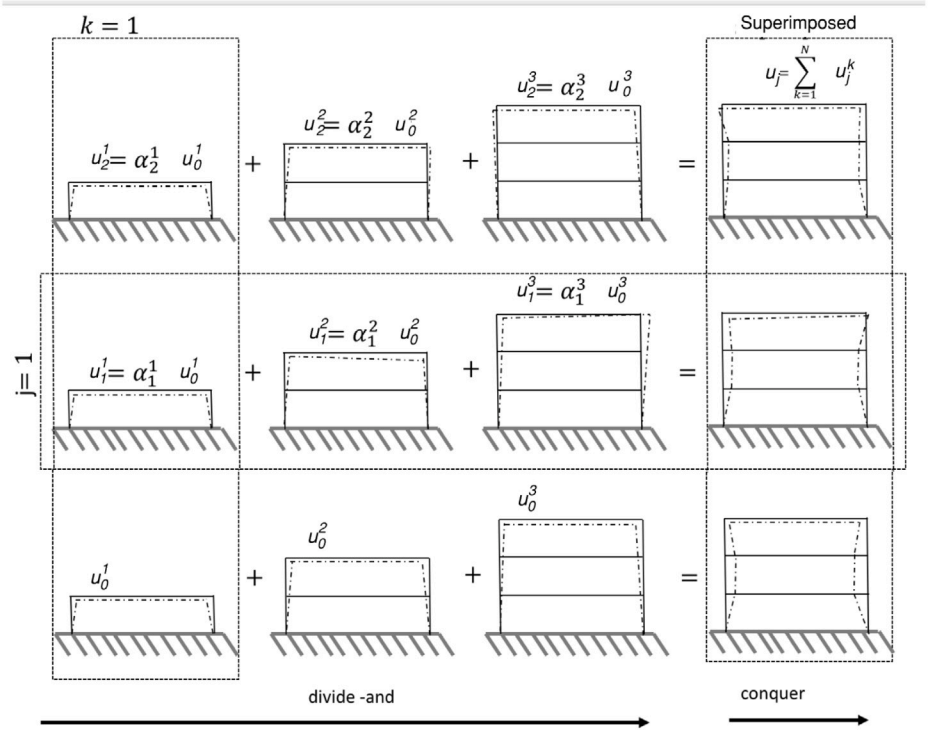

Fig. 3. The superposition of each realization. The first row from the bottom indicates the nominal simulation. Each row thereafter, from bottom to top, represents the superposition of each realization.

3. Distortion minimization objective function and sensitivity formulation

3.1. Deterministic objective function formulation

As a basis for our formulation, we take Eq. (15) outlined in Section 2.5 to formulate the displacement field $\boldsymbol{u}_j$. Given a realization j of the scaling factors $\boldsymbol{\alpha}_j$, the distortion quantity Γ_j is evaluated as a p-mean of the displacement field coming from the AM process at the interface(s) where we choose to control distortion, see Eqs. (16)–(18).

$$\boldsymbol{v}_j = \boldsymbol{\Theta}\boldsymbol{u}_j \tag{16}$$

In Eq. (16), the matrix $\boldsymbol{\Theta}$ is a diagonal matrix containing node density x_{node}. Whereas the nodal density is calculated as the weighted value of the element's density $\boldsymbol{x}_e$ that surrounds the analyzed node.

$$\boldsymbol{\Theta} = \begin{cases} x_{node} = \frac{\sum_{e=1}^{IntEl} x_e}{IntEl} & \text{for interface DOFs} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

where: $IntEl$ is the number of elements intersecting with the analyzed node located at the targed edges (see Fig. 5 and Fig. 8).

We can define the distortion quantity as the $p-$mean of the vector $\boldsymbol{v}_j$.

$$\Gamma_j = \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{1/p} \quad for\ j = 1...m \tag{18}$$

where $\circ$ indicates the Hadamard operator, which in this case implies p-power computed component-wise (when placed at exponential) and the product when it is placed in the middle.

The vector $\mathbf{1}$ is filled with ones and has the same size as $\boldsymbol{v}_j$.

We include a division by the number of interface nodes N_n, to decouple the distortion quantity from mesh refinements. For a more detailed explanation, see Appendix D.

3.2. Robust objective function

We propose three different formulations for the problem outlined in Section 2, minimization of the distortion under stochastic process along with compliance. The distortion measure $\Gamma(\boldsymbol{\alpha})$ is a function of the random vector $\boldsymbol{\alpha} \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$ with probability density function $f(\boldsymbol{\alpha})$. The mean and the variance of the distortion measure $\Gamma(\boldsymbol{\alpha})$ are given

as:

$$\mu_\Gamma = \int_{-\infty}^{+\infty} \Gamma(\boldsymbol{\alpha}) f(\boldsymbol{\alpha}) d\boldsymbol{\alpha} \tag{19}$$

$$\sigma_\Gamma = \int_{-\infty}^{+\infty} [\Gamma(\boldsymbol{\alpha}) - \mu_\Gamma]^2 f(\boldsymbol{\alpha}) d\boldsymbol{\alpha} \tag{20}$$

As a subsequent step, we estimate these quantities using the Monte Carlo (*MC*) method and the First-Order-Second-Moment (*FOSM*) method. For the *MC*-method we distinguish between two cases, $p = 2$ and $p > 2$, as will be explained in the subsequent section. The probabilistic approach is embedded in the overall density-based topology optimization algorithm.

3.3. Monte Carlo approach

The FEM procedure, by its very nature, is discrete. Therefore, the displacement fields coming from FE are discrete and consequently, the distortion quantity Γ is evaluated from discrete quantities, the displacement fields. Eventually, the statistical characteristics of the distortion quantities are evaluated with respect to the random vector $\boldsymbol{\alpha}$ as follows.

$$\mu_\Gamma(\boldsymbol{x}) = E[\Gamma(\boldsymbol{x}, \boldsymbol{\alpha})] \approx \frac{1}{m-1} \sum_{j=1}^{m} \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j) \tag{21}$$

$$\begin{aligned} \sigma_\Gamma(\boldsymbol{x}) &= \sqrt{E[\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}) - \mu_\Gamma]^2} \\ &\approx \sqrt{\frac{1}{m-1} \sum_{j=1}^{m} (\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j) - \mu_\Gamma(\boldsymbol{x}))^2} \end{aligned} \tag{22}$$

From these equations, the derivatives of the mean value $\frac{\partial \mu_\Gamma}{\partial x_e}$ and standard deviation $\frac{\partial \sigma_\Gamma^2}{\partial x_e}$, given in Eqs. (23) to (25).

$$\frac{\partial \mu_\Gamma(\boldsymbol{x})}{\partial x_e} = E\left[\frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha})}{x_e}\right] \tag{23}$$

$$\frac{\partial \sigma_\Gamma(\boldsymbol{x})}{\partial x_e} = \frac{1}{2\sigma_\Gamma} \frac{\partial \sigma_\Gamma^2(\boldsymbol{x})}{\partial x_e} \tag{24}$$

where

$$\frac{\partial \sigma_\Gamma^2(\boldsymbol{x})}{\partial x_e} = E\left[\frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha})}{\partial x_e} - \frac{\partial \mu_\Gamma(\boldsymbol{x})}{\partial x_e}\right]^2 \tag{25}$$

To compute the sensitivity of the response $\frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)}{\partial x_e}$, given realization j, we use the adjoint formulation. Hence, the augmented distortion measurement is given as $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}_j)$.

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}_j) = \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j) + \sum_{k=1}^{N} \alpha_j^k \lambda_j^{kT} \left(\boldsymbol{K}^k \boldsymbol{u}^k - \boldsymbol{f}^k\right) \tag{26}$$

where λ_j^{kT} is the vector containing Lagrange multipliers. Here we keep α_j^k out of the brackets to facilitate the isolation of $\sum_{k=1}^{N} \alpha_j^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}$ later in the derivations, which are given in Appendix A.

Subsequently, we obtain the operator $\frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)}{\partial x_e}$ and all the derivations are given in Appendix A, Eqs. (51) to (56).

$$\begin{aligned} \frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)}{\partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j \\ &+ \left(\frac{1}{N_n}\right)^{1/p} \sum_{k=1}^{N} \alpha_j^k \lambda_j^{kT} \left(\frac{\partial \boldsymbol{K}^k}{\partial x_e} \boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \end{aligned} \tag{27}$$

Eq. (27) is then organized in a compact form.

$$\frac{\partial \Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)}{\partial x_e} = \boldsymbol{P}_j + \left(\frac{1}{N_n}\right)^{1/p} \sum_{k=1}^{N} \alpha_j^k \lambda_j^{kT} \boldsymbol{S}^k \tag{28}$$

where:

$$\boldsymbol{P}_j = \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j \tag{29}$$

In Eq. (28) we can superimpose $\boldsymbol{P}_j$ using $\boldsymbol{u}_j$ and $\boldsymbol{v}_j$, whereas $\boldsymbol{S}^k$ is invariant to the realizations, and it is defined in Eq. (30).

$$\boldsymbol{S}^k = \frac{\partial \boldsymbol{K}^k}{\partial x_e} \boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e} \tag{30}$$

where:

$$\frac{\partial \boldsymbol{K}^k}{\partial x_e} = \frac{\partial \boldsymbol{k}_e}{\partial x_e} = -p_e x_e^{p_e - 1} \underbrace{\int_V \boldsymbol{B}^T \boldsymbol{D}_0 \boldsymbol{B} d\upsilon}_{\boldsymbol{k}_0} = -p_e x_e^{p_e-1} \boldsymbol{k}_0 \tag{31}$$

and

$$\frac{\partial \boldsymbol{f}^k}{\partial x_e} = \frac{\partial \boldsymbol{f}_e^k}{\partial x_e} = -p_e x_e^{p_e - 1} \underbrace{\int_V \boldsymbol{B}^T \boldsymbol{D}_0 \Delta \epsilon_e^k d\upsilon}_{f_0^k} = -p_e x_e^{p_e-1} \boldsymbol{f}_0^k \tag{32}$$

Ultimately, the last component λ^k is obtained by the solution of Eq. (33) and its superposition is studied in detail in Sections 3.3.1 and 3.3.2.

$$\boldsymbol{K}^k \lambda_j^k = -\left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{\Theta} \boldsymbol{v}_j^{\circ(p-1)} \tag{33}$$

Eventually, we compute the complete sensitivity of $\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)$ through superposition, and subsequently, compute the sensitivity of the mean value, variance and standard deviation, using respectively Eqs. (23)–(25).

3.3.1. Sensitivity formulation for $p = 2$

For the case when $p = 2$, the right-hand side of the adjoint formulation given in Eq. (33) becomes a linear combination of α_j^β and $\boldsymbol{u}^\beta$, see Eq. (34). Thus, it can easily be decoupled. To discern the advantages of this case, we expand and substitute $p = 2$ to $\boldsymbol{v}_j^{p-1}$ which gives the adjoint system of equations as a linear summation on the right-hand side. Considering Eqs. (15) and (16), this reduces the order of $\boldsymbol{v}$ to a linear combination of α_j^β and $\boldsymbol{u}^\beta$, thus allowing for the superposition of the solution of Eq. (34).

$$\boldsymbol{K}^k \lambda_j^k = -\left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{-\frac{1}{2}} \boldsymbol{\Theta}^2 \sum_{\beta=1}^{N} \alpha_j^\beta \boldsymbol{u}^\beta \tag{34}$$

At each layer k and with system matrix $\boldsymbol{K}^k$, we compute λ^k for the summation of the total displacement field $\sum_{\beta=1}^{N} \alpha_j^\beta \boldsymbol{u}^\beta$. This displacement field is computed from the nominal AM-simulation settings. The kth index on the left-hand side of Eq. (34) varies during the solution, whereas the index β on the right-hand side is fixed to the summation of the total number of layers N. Thus, to avoid any ambiguity, we have purposely introduced a new notation β on the right-hand side for the index of the layer.

$$\boldsymbol{K}^k \lambda_j^k = -\left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{-\frac{1}{2}} \boldsymbol{\Theta}^2 \sum_{\beta=1}^{N} \alpha_j^\beta \boldsymbol{u}^\beta \tag{35}$$

We perceive that the combination of $\alpha_j^\beta \boldsymbol{u}^\beta$ is linear. To superimpose λ_j^k for each realization $\boldsymbol{\alpha}_j^\beta$ we proceed as follows. First, we solve for $\lambda^{\beta,k}$ at each layer k for the displacement field of each nominal layer-up process $\boldsymbol{u}^\beta$, see Eq. (36). Afterward, we generate λ_j^k for each realization as the linear combination of α_j^β and $\lambda^{\beta,k}$, Eq. (37).

$$K^k \lambda_j^{\beta,k} = -\left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{-\frac{1}{2}} \boldsymbol{\Theta}^2 \boldsymbol{u}^\beta \quad for \quad k = 1...N \tag{36}$$

$$\lambda_j^k = \sum_{\beta=1}^{N} \alpha_j^\beta \lambda^{\beta,k} \tag{37}$$

Thus, in this case, we solve only the adjoint system for the nominal AM process simulation, and then we can superimpose the solutions for all the other realizations λ_j^k, thus leading to $N \times N$ number of equations to be solved. This is, therefore, a sound approach when the number of realizations is smaller than the number of layers.

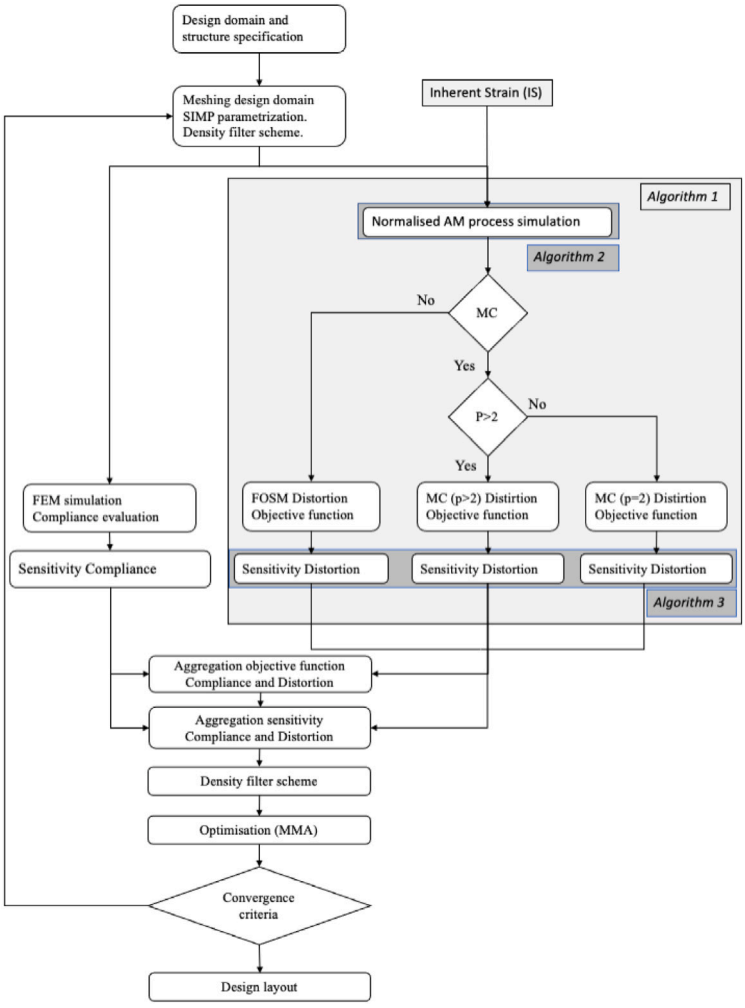

Fig. 4. Algorithm workflow: integration of the stochastic AM-process simulation into the traditional density-based TO.

3.3.2. Sensitivity formulation for p>2

For the case when the exponent, $p > 2$, in Eq. (34) there is no significant improvement in the derivations and computational speed-up since the right-hand side of Eq. (33) is highly nonlinear and hard to decouple. The solution of the adjoint needs to be done at each layer k for all the realizations m, which leads to a total $m \times N$ number of system of equations to be solved.

3.4. FOSM objective function formulation

The computational cost of performing a Monte Carlo simulation is enormous if $p > 2$. An alternative, less expensive approach is the First-Order-Second-Moment (*FOSM*) method. To estimate the mean value and variance of the distortion measure, Γ is approximated by a Taylor series expansion at the mean value vector $\boldsymbol{\mu_\alpha}$ of the random vector $\boldsymbol{\alpha}$.

$$\Gamma(\boldsymbol{\alpha}) = \Gamma(\boldsymbol{\mu_\alpha}) + \sum_{k=1}^{N} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k} \left(\alpha^k - \mu_\alpha^k\right) + \frac{1}{2} \sum_{k_1=1}^{N} \sum_{k_2=1}^{N} \frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_1} \partial \alpha^{k_2}} \left(\alpha^{k_1} - \mu_\alpha^{k_1}\right) \left(\alpha^{k_2} - \mu_\alpha^{k_2}\right) + \cdots \tag{38}$$

This approximation is used to approximate Eqs. (19) and (20), respectively. Note k_1 and k_2 are equal to k and here they also indicate the layer number.

Considering only the first-order terms of the Taylor series in Eq. (38) they provide respectively the approximation of the mean value and the variance w.r.t. the random vector $\boldsymbol{\alpha}$.

$$\begin{aligned} \mu_\Gamma &\approx \Gamma(\boldsymbol{\mu_\alpha}) \\ \sigma_\Gamma^2 &\approx \sum_{k_1=1}^{N} \sum_{k_2=1}^{N} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_1}} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_2}} \operatorname{cov}\left(\alpha^{k_1}, \alpha^{k_2}\right) \end{aligned} \tag{39}$$

where $\operatorname{cov}\left(\alpha^{k_1}, \alpha^{k_2}\right)$ is the covariance of two entries of the input vector $\boldsymbol{\alpha}$. In Eq. (39) the distortion quantity $\Gamma(\mu)$ is computed using Eqs. (14), (15) and (18) at the mean value $\mu_{\boldsymbol{\alpha}}$ of the scaling factor $\boldsymbol{\alpha}$.

Computing the variance with Eq. (39) requires the derivatives of $\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k}$, which are given in Eq. (40).

$$\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k} = \frac{1}{p} (\mathbf{1}^T \boldsymbol{v}_j)^{\frac{1}{p}-1} \mathbf{1}^T p \boldsymbol{v}_j^{\circ(p-1)} \circ \boldsymbol{\Theta} \boldsymbol{u}^k \tag{40}$$

For the derivation of this equation, refer to Appendix B, with the end form Eq. (60).

3.5. FOSM sensitivity formulation

The computation of the sensitivity for the mean value simply equals the deterministic derivative, evaluated for the mean value of the input parameter vector, i.e., executing Eq. (40) for $\boldsymbol{\mu_\alpha}$. For an efficient computation of the gradient of the variance σ_Γ^2, we build the augmented

formulation $\chi(\boldsymbol{x},\boldsymbol{\alpha})$ using the adjoint method.

$$\begin{aligned}\chi(\boldsymbol{x},\boldsymbol{\alpha}) &= \sigma_I^2(\boldsymbol{x}) + \sum_{k=1}^{N} \alpha^k \lambda^{kT} \left(\boldsymbol{K}^k \boldsymbol{u}^k - \boldsymbol{f}^k\right) \\ &= \sum_{k_2=1}^{N}\sum_{k_2=1}^{N} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_2}} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_2}} \operatorname{cov}\left(\alpha^{k_2}, \alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N} \alpha^k \lambda^{kT} \left(\boldsymbol{K}^k \boldsymbol{u}^k - \boldsymbol{f}^k\right)\end{aligned} \tag{41}$$

After further derivations, given in Appendices B and C, we obtain the sensitivity as:

$$\frac{\partial \sigma_I^2(\boldsymbol{x})}{\partial x_e} = \boldsymbol{M}_1 + \sum_{k=1}^{N} \alpha^k \lambda^{kT} \boldsymbol{S}^k \tag{42}$$

where $\boldsymbol{S}^k$ is the same as in Eq. (30) evaluated at $\boldsymbol{\mu_\alpha}$, and $\boldsymbol{M}_1$ is given as follows.

$$\boldsymbol{M}_1 = 2 \sum_{k_2=1}^{N}\sum_{k_2=1}^{N} \left(\frac{1}{N_n}\right)^{1/p} \boldsymbol{A}_s \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_2}} \operatorname{cov}\left(\alpha^{k_2}, \alpha^{k_2}\right) \tag{43}$$

and λ^k is computed from the solution of the adjoint system of equations at each layer k.

$$\boldsymbol{K}^k \lambda^k = \left[-2 \sum_{k_1=1}^{N}\sum_{k_2=1}^{N} \left(\frac{1}{N_n}\right)^{1/p} \boldsymbol{B}_s \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^{k_2}} \operatorname{cov}\left(\alpha^{k_1}, \alpha^{k_2}\right)\right]^T \tag{44}$$

4. Algorithm workflow and computational cost

In this section, we organize the algorithm workflow in relation to each formulation of our framework, see Fig. 4.

Important features for each method respectively are the number of systems of equations needed to be solved for the objective function and sensitivity, as these mainly drive the computational cost. All the results are summarized in Table 1. Each method requires the same number of systems of equations to be solved for evaluation of the objective function, precisely N. *FOSM* requires the least number of equations to be solved for the sensitivity, namely N systems of equations. The *MC* ($p > 2$) case requires the most significant number of equations to be solved for sensitivity $N{\times}m$, whereas for the other case, *MC* ($p = 2$), this number is smaller, namely (N^2). This case is more computationally efficient when the number of realizations m is smaller than the number of layers N. Yet, this computational improvement comes at a cost in memory allocation. Therefore, the use of each method is more case-related.

5. Numerical examples

5.1. Test setup

The performances of the proposed optimization formulations are presented and analyzed in this section. To confirm the minimization of the distortion quantity, we compare the proposed formulations with the case when only compliance is minimized. From the latter case, we obtain a reference distortion. Afterward, we run the optimization while accounting for the AM process simulation without variability in the process parameter, i.e., deterministic distortion minimization. We name this case shortly *deterministic case*. Hence, to illustrate the robustness of designs obtained with our framework, we compare the proposed stochastic formulations with the *deterministic case*, as well as the comparison of the stochastic formulations *FOSM* and *MC* with each other. As a demonstrative example, we take a cantilever beam with the dimensions of length $l = 150$ mm, height $h = 50$ mm and thickness of $d = 10$ mm, see Figs. 5 and 8. Due to the considerable computational cost, as well as the clarity of presentation of results, all examples are in 2D.

Algorithm 1: Stochastic AM- simulation

Input : $\boldsymbol{x}$ -density field, $\boldsymbol{\alpha}_j$-control parameters and building orientation.

Output: statistical measures of the distortion and its gradient

$\boldsymbol{\Theta}$ *and* $\frac{\partial \boldsymbol{\Theta}}{\partial x_e}$ // Compute nodal densities matrix and their derivative (17), (47)

$U[dofs, k]$, $\boldsymbol{S}^k \leftarrow$ ***Algorithm.2*** // Displacement field

if *Monte Carlo* **then**

  if $p \prec 2$ **then**

    for $j \leftarrow 1$ **j to** $\leftarrow m$ (*no. realizations*) **do**

      Γ_j // Comp. (18)

    end for

  else

    for $j \leftarrow 1$ **j to** $\leftarrow m$ (*no. realizations*) **do**

      Γ_j, P_j // Comp. (18), (19)

    end for

  end if

else

  $\Gamma, \boldsymbol{M}_1$ // Comp. (39), (43),

end if

$\frac{\partial \Gamma_j(\boldsymbol{x})}{x_e}$ *or* $\frac{\partial \Gamma(\boldsymbol{x})}{x_e} \leftarrow$ ***Algorithm.3*** // Gradient

$\mu_\Gamma, \sigma_\Gamma, \frac{\partial \mu_\Gamma}{\partial x_e}$ *and* $\frac{\partial \sigma_\Gamma}{\partial x_e}$ // Comp. (21) or (22), (23), (24)

return $\underbrace{\kappa_2 \mu_\Gamma(x_e) + (1-\kappa_2)\sigma_\Gamma(x_e)}_{part-distortion}$ $\underbrace{\kappa_2 \frac{\partial \sigma_\Gamma}{\partial x_e} + (1-\kappa_2)\frac{\partial \sigma_\Gamma}{\partial x_e}}_{sensitivity}$

Algorithm 2: Stochastic AM-process Objective function

Input : $\boldsymbol{x}$ -density field, $\boldsymbol{\alpha}_j$-control parameters and building orientation.

Output : $\boldsymbol{u}_{[dofs,k]}$ & $\boldsymbol{S}^k$ -displacement field and its sub-part of the sensitivity.

$\boldsymbol{\Theta}$ *and* $\frac{\partial \boldsymbol{\Theta}}{\partial x_e}$ // Retrieve nodal densities matrix

for $k \leftarrow 1$ **to** $k \leftarrow N$ (*no. layers*) **do**

  Last(k) & *built*(k) // Identify the lastly-added and already-built layer

  Δf^k; $\frac{\partial \Delta f}{\partial x_e}$; // Comp. (12), (32)

  $\Delta \boldsymbol{K}^k$; $\frac{\partial \Delta \boldsymbol{K}}{\partial x_e}$; // Comp. (11), (31)

  $\boldsymbol{u}_{[dofs,k]} \leftarrow \Delta \boldsymbol{u}^k$ // Solve (10) for $\Delta \boldsymbol{u}^k$ and store for layer k

  $\boldsymbol{S}^k$ // Compute from (30) and store for layer-k

  empty Δf^k; $\frac{\partial \Delta f}{\partial x_e}$; $\boldsymbol{K}^k$

end for

return $\boldsymbol{u}_{[dofs,k]}$ & $\boldsymbol{S}^k$

Table 1
Number of systems of equations needed to be solved for the objective function and sensitivity considering the number of realizations m and the number of layers N.

Method	*MC* ($p > 2$)	*MC* ($p = 2$)	*FOSM*
Objective function	N	N	N
Sensitivity	$m \times N$	$N \times N$	N

Material properties are Young's modulus $E = 115$ GPa and Poisson's ratio $\nu = 0.3$. For AM-process simulation we borrow the IS ($\Delta\epsilon = 0.05$) from the use case example presented in [6]. Plane strain formulation is used for the FEM implementation. We normalize the compliance by the applied load and account for the objective function contribution to the structural analysis. Consequently, we get the magnitude of the compliance comparable to the distortion coming from the AM-process with the unit [mm]. For the SIMP interpolation we set the penalization

Algorithm 3: Sensitivity

Input : $\boldsymbol{x}, \boldsymbol{\alpha}_j, \boldsymbol{P}_j, \boldsymbol{M}_1$.

Output: $\frac{\partial \Gamma_j(x_e)}{x_e}$ *or* $\frac{\partial \Gamma(x_e)}{x_e}$

for $k \leftarrow 1$ **to** $k \leftarrow N\,(no.layers)$ **do**
  $built(k)$ // Identify already-built layer
  if *Monte Carlo* **then**
    if *P=2* **then**
      for $\beta \leftarrow 1$ **to** $\beta \leftarrow N\,(no.layers)$ **do**
        $\lambda_{[dofs,k,\beta]} \leftarrow \lambda^{\beta,k}$ // Comp. (36) and store-layer β
      end for
      for $j \leftarrow 1$ j **to** $\leftarrow m\,(no.realizations)$ **do**
        $\lambda_j^k \leftarrow \sum_{\beta=1}^{N} \boldsymbol{\alpha}_j^{\beta} (\lambda^{\beta,k})^T$ // Comp. (37)
        $\frac{\partial \Gamma_j(x_e)}{x_e}$ // Comp. (28)
      end for
    else
      for $j \leftarrow 1$ j **to** $\leftarrow m\,(no.realizations)$ **do**
        $\lambda_{[dofs,k,j]} \leftarrow \lambda_j^k$ // Comp. (36)
        $\frac{\partial \Gamma_j(x_e)}{x_e}$ // Comp. (28)
      end for
    end if
  else
    λ_j^k // Comp. (44)
    $\frac{\partial \Gamma_j(\boldsymbol{x})}{x_e}$ // Comp. (28)
  end if
end for
return $\frac{\partial \Gamma_j(\boldsymbol{x})}{x_e}$ *or* $\frac{\partial \Gamma(\boldsymbol{x})}{x_e}$

$p_e = 3.5$, which is fixed for all cases, and for the density filter scheme we choose the radius to $r_{min} = 1.7$ elements. We choose a low r_{min} parameter to allow for a reasonably black-and-white design layout in this context. The volume fraction is set to 0.5. As a penalization factor for the p-mean we choose $p = 12$ and $p = 2$. The optimization is performed for each of the following cases while keeping the same parameters using MMA from [25] and as convergence criteria are used, the threshold volume change of 0.05 and as termination criteria 200 iterations. To summarize, the cases that are considered are as follows:

- *Only compliance*
- *Deterministic case*
- *FOSM*
- *MC ($p = 2$)*
- *MC ($p > 2$)* precisely *($p = 12$)*

We start from the half-full solid design, i.e., $x_e = 0.5$. As a scaling factor we assume α_j^k to have a normal distribution with mean $\mu = 1$ and standard deviation $\sigma = 0.1$, i.e. $\alpha_j^k \sim \mathcal{N}(1, 0.1)$. Despite that a coefficient of variation of 10% is not unreasonable, there is no real justification for this choice. In practical application, the stochastic distribution of α should originate from experimental data. To test the method on a different example, we vary the orientation angle of the 3D-building process and add a distributed load on the considered edge (see Table 5).

5.2. First example—printing orientation 0°

The cantilever beam example given in Fig. 5 with a 0° orientation for the AM process simulation is considered as a first example. The degrees of freedom on the left-hand side are fixed, and a single load $P = 80kN$ is applied to the upper right corner. The short edges of the domain are taken as interfaces of interest where distortion is measured. The design domain is discretized by $\Omega = 300 \times 120$ elements in the

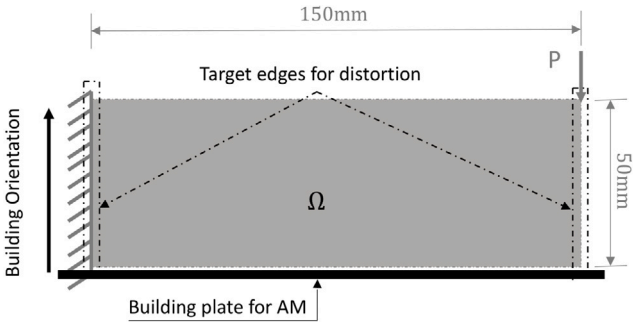

Fig. 5. Case 0° printing orientation: The cantilever beam. The loading condition is indicated by gray color, and the strong black color marks the AM process.

respective horizontal and vertical directions. The number of layers is fixed to the number of vertical elements, particularly 120 layers. For *MC* simulation, 100 realizations are taken into consideration. Note that $m < N$, therefore sensitivity analysis for $MC(p = 2)$ will be less costly than $MC(p > 2)$. The weighting factor κ_1 is set to 0.2 implying that compliance is 20% of the objective, and distortion parts 80%. On the other hand, the other weighting factor κ_2 is set to 0.4, which likewise means splitting 40% of the weight on the mean distortion and 60% on the standard deviation. First, we optimize the structure only for compliance minimization. Afterward, the optimization is performed while including both, compliance and part-distortion, and we refer to this as the *deterministic case*. Ultimately, the latter optimization is performed taking into consideration the uncertainties in IS while employing our three proposed stochastic methods. Comparisons of the design layout are given in Fig. 6, together with the distorted configurations of the optimized designs, obtained from the nominal AM-process simulation.

For the following comparisons, we refer to the case where we minimize only compliance to evaluate how the distortion is suppressed when it is compared to the *deterministic case*. Eventually, to evaluate the robustness of our formulation, we compare the *deterministic case* with the proposed stochastic methods of this work. Note that only the distortion of the left and right edges of the design is relevant to this optimization, distortions within the domain are not included. The motivation for this is that in practical applications it is typically the edge of a structure that represents the interface to another structure and therefore, the deviation from the ideal structure should be controlled here. This can be accomplished by modifying the selection matrix $\boldsymbol{\Theta}$ accordingly. As evident in the zoomed-in, right-hand side, edges taken into consideration in Fig. 6, all the cases when distortion is included into the optimization process, i.e., case (b) to (e), lead to a different blueprint, and significantly lower distortion values, when compared to the case when only compliance is minimized, case (a). This also holds for the inner part of the design domain, which is not explicitly considered in the optimization, since it is coupled with the distortions at the edges. Furthermore, it is observed that the pairs *deterministic case* with *FOSM* and *MC ($p = 2$)* with *MC ($p>2$)* generate a similar layout with respect to each other. This pattern comes from the fact that for *the FOSM* approach, less standard deviation is quantified into the objection function, as evident in Table 3. Thus, this does not lead to much deviation from the deterministic case. On the other hand, for the cases *MC ($p=2$)* with *MC ($p>2$)*, more contribution comes from the standard deviation to the objective function, therefore, leading to a more similar design layout.

The results in Fig. 6, all show overhanging regions that would require support to enable printing. This is accepted in this study, as our focus is on including and studying the effect of AM process uncertainty. If desired, various methods exist in the literature to control overhanging regions and/or add support structures.

Noticeable in Fig. 7, during the early iterations, the compliance quantity decays rapidly for each case, whereas the distortion quantity

Table 2
Case 0° printing orientation: comparisons of the statistical quantities of the distortion with the deterministic case and the compliance with the case when only compliance is minimized using *MC* with 100 realizations.

Method	$\mu_{Dist.}$	w.r.t (1)	w.r.t (2)	$\sigma_{Dist.}$	w.r.t (1)	w.r.t (2)	Compliance	
(1) Only compliance	11.875			4.68×10^{-1}			9.8071	
(2) Deterministic case	7.7442	↓ 35.6%		3.2×10^{-1}	↓ 31.4%		10.1732	↑ 3.7%
(3) *FOSM*	2.844	↓ 76%	↓ 62.7%	2.52×10^{-1}	↓ 46.16%	↓ 21.4%	10.8203	↑ 10.3%
(4) *MC (p=2)*	1.45	↓ 88.7%	↓ 81%	3.6×10^{-2}	↓ 92.2%	↓ 88.6%	11.2650	↑ 14.8%
(5) *MC (p>2)*	0.45	↓ 96.2%	↓ 94%	1.2×10^{-2}	↓ 92%	↓ 88.6%	11.2650	↑ 14.8%

Table 3
Case 0° printing orientation: comparisons of the statistical quantities of the distortion while evaluated with *FOSM* and *MC* with 100 realizations.

Method	μ_{FOSM}	μ_{MC}		σ_{FOSM}	σ_{MC}	
FOSM	2.78	2.844	↑ 2.2%	2.09×10^{-2}	2.52×10^{-1}	↑ 91.7%

spikes. Afterward, due to local reconfiguration of the design layout, the distortion quantities decay swiftly for each method too. This switch of magnitudes happens because the compliance contribution is larger than the distortion quantity at the initial state, and so does the respective sensitivity. Hence, the algorithm will emphasize the compliance contribution more than the distortion measure during early iterations. After approximately 50 iterations, the optimizer balances these two quantities to a comparable magnitude. Thereafter, both quantities follow a slow, yet steady, decay. However, each case has some fluctuation in the distortion quantity throughout the iterations.

While comparing the deterministic case with the stochastic cases, not only the standard deviation is strongly minimized, but also the principal distortion quantity, the mean value, is further minimized. In this example, the *MC (p>2)* result shows the highest robustness against distortion. Of the robust cases, the *FOSM* case shows the least robustness improvement, with a standard deviation reduction of 21.47% w.r.t. the deterministic case, as opposed to 88.65% and 96.1% reductions for both *MC* cases. All these improvements in the distortion quantity come with a slight increase in compliance. More specifically, for this particular example, we see a 3.7% increase in compliance for the deterministic case, 10.3% for *FOSM* and 14.8% for *MC* ($p > 2$) . This is expected and can be seen as the cost of the increased robustness against AM-induced distortions. The statistical quantities of the distortion summarized in Table 2 are evaluated using *MC* simulation on the final design. We do so because *FOSM* lacks accuracy on capturing all these quantities precisely. Because the *FOSM* method is used for graphically quantifying statistical quantities in the optimization process for the *FOSM* case, we do a comparison of their statistical quantities using MC and the results are summarized in Table 3. For the mean value, the difference between *FOSM* and *MC* distortion evaluation is small: 2.6% for the deterministic design, and 5% for the *FOSM* design. Considering the far lower computational cost of *FOSM*, this is acceptable. However, the standard deviation is 91% larger for both designs, which is a large difference. This may explain why the *FOSM* design also showed only modest improvements in robustness compared to the *MC* cases (Table 2).

Finally, it is noted that in this particular case, we may see a significant improvement of the distortion because for the compliance minimization design case the majority of the elements associated with the interface nodes of interest become void, on the left and right boundary of the domain. This also removes their contribution from the distortion measure. For this reason, a second example is considered where this is much less the case.

5.3. Second example—printing orientation 90°

This section presents the second application of the proposed approach. The cantilever beam is loaded with a distributed load of 8 kN/m on the right edge, see Fig. 8. In this case, a significant difference is also that the AM process simulation is performed while the design domain is orientated at 90°.

In this example, the design domain is discretized by $\Omega = 90 \times 300$ elements in respectively horizontal and vertical directions. The number of layers is fixed to the number of vertical elements, in this case, 300 layers. For *MC* simulation, 100 realizations are taken into consideration.

Note that again $m < N$ and by a larger factor than in Example 1, therefore again, sensitivity analysis for $MC(p = 2)$ will be less costly than $MC(p > 2)$. Multiobjective weights are kept the same as in the first example, i.e., $\kappa_1 = 0.2$ and $\kappa_2 = 0.4$.

Fig. 8 illustrates the initial domain. The other parameters are assigned the same values as used in the first example too. Therefore, the goal is to find a design layout, which has both, (1) minimal compliance and (2) minimal distortion' statistical components. The optimization is again performed for the five formulations, and the design layouts are given in Fig. 9 while maintaining the same order as in Example 1.

From Fig. 10 we observe that the mean value and the standard deviation have a smoother behavior when it is compared to the first example. This behavior is because the edge taken into consideration in this case always remains solid with respect to compliance minimization. Hence, there is no sudden large void, as opposed to the first example, which leads to sharp decay in the distortion quantity followed by continued fluctuations.

For this problem, see Fig. 9, it is observed that designs have converged to a clear solid/void solution. Furthermore, in this example, all the cases when distortion is included into the optimization process lead to a different blueprint, i.e., the edges have darker blue, hence less distortion. This is observed when these blueprints are compared to the case when only compliance is minimized. Unlike the first example, it is observed that only the pairs *MC (p=2)* with *MC (p>2)* generate a similar layout. The other robust optimization cases, *FOSM*, generates a different design layout when compared to *deterministic case* and *MC (p=2) & (p>2)*. This pattern comes from the fact that for the *FOSM* the standard deviation is underestimated less in the objective function, when compared to the first example, see Table 3. On the other hand, for the cases *MC (p=2)* and *MC (p>2)* do not suffer from this inaccuracy, therefore, leading to a more similar design layout and different from the other two cases.

More specifically, for this particular example, we see 47.5% increase in compliance for the deterministic case, 48.2% for *FOSM*, 70% for *MC (p=2)* and 50% for *MC* ($p > 2$) . The results are summarized in Table 4.

Clearly, for this example, distortion control is more demanding, and a stronger trade-off is needed w.r.t. the compliance of the design. In comparison to the deterministic case, *FOSM* again shows only modest robustness improvements (2.7% reduction of the mean distortion, and 10.2% reduction in standard deviation). As before, the *MC* formulations achieve larger distortion robustness gains. *MC (p=2)* shows again the strongest performance with 81% and 87% reductions in mean and

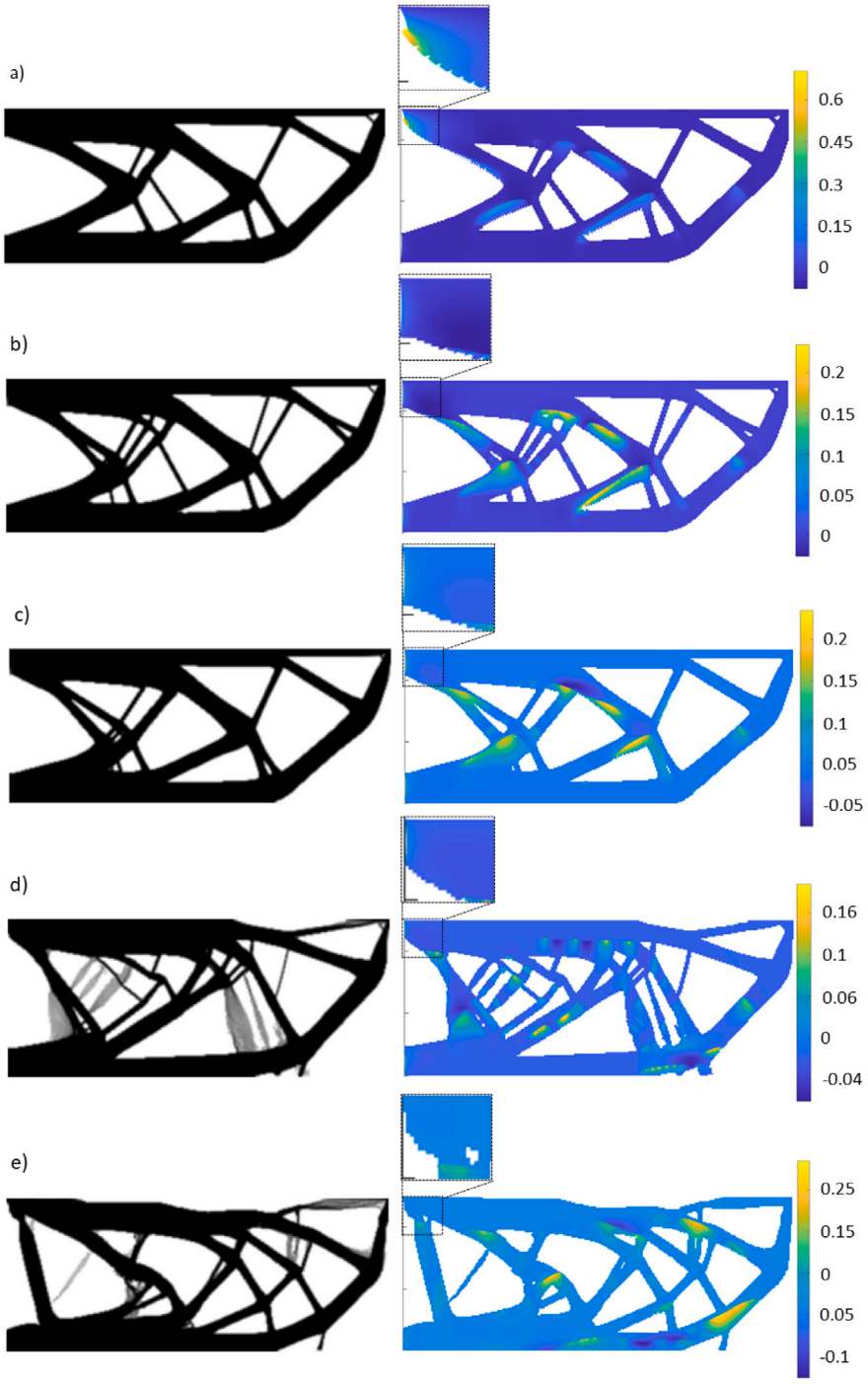

Fig. 6. The optimized design layout for respectively (a) Only compliance ; (b) Deterministic case; (c) *FOSM*; (d) *MC* ($p > 2$) and (e) *MC (p=2)*. On the right-hand side are given the deformed structures from the fabrication process, with distortion magnitude in mm. The AM process simulation is performed while the design domain is orientated 0°.

Table 4
Case 90° printing orientation: comparisons of the statistical quantities of the distortion with the deterministic case and the compliance with the case when only compliance is minimized using *MC* with 100 realizations.

Method	$\mu_{Dist.}$	w.r.t (1)	w.r.t (2)	$\sigma_{Dist.}$	w.r.t (1)	w.r.t (2)	Compliance	
(1) Only compliance	6.01			8×10^{-2}			77.53	
(2) Deterministic case	4.77	↓ 20%		3.9×10^{-2}	↓ 51.3%		114.42	↑ 47.5%
(3) *FOSM*	4.5691	↓ 23%	↓ 2.7%	3.5×10^{-2}	↓ 56.2%	↓ 10.2%	114.9	↑ 48.2%
(4) *MC (p=2)*	0.8843	↓ 41%	↓ 81%	5.1×10^{-3}	↓ 93.6%	↓ 86.9%	132.4	↑ 70%
(5) *MC (p>2)*	3.77	↓ 37%	↓ 20.9%	2.09×10^{-2}	↓ 73.8%	↓ 46.4%	116.88	↑ 50%

Table 5
Case 90° printing orientation: comparisons of the statistical quantities of the distortion while evaluated with *FOSM* and *MC* with 100 realizations.

Method	μ_{FOSM}	μ_{MC}		σ_{FOSM}	σ_{MC}	
FOSM	4.36	4.569	↑ 4.4%	3.8×10^{-3}	3.5×10^{-2}	↑ 89.1%

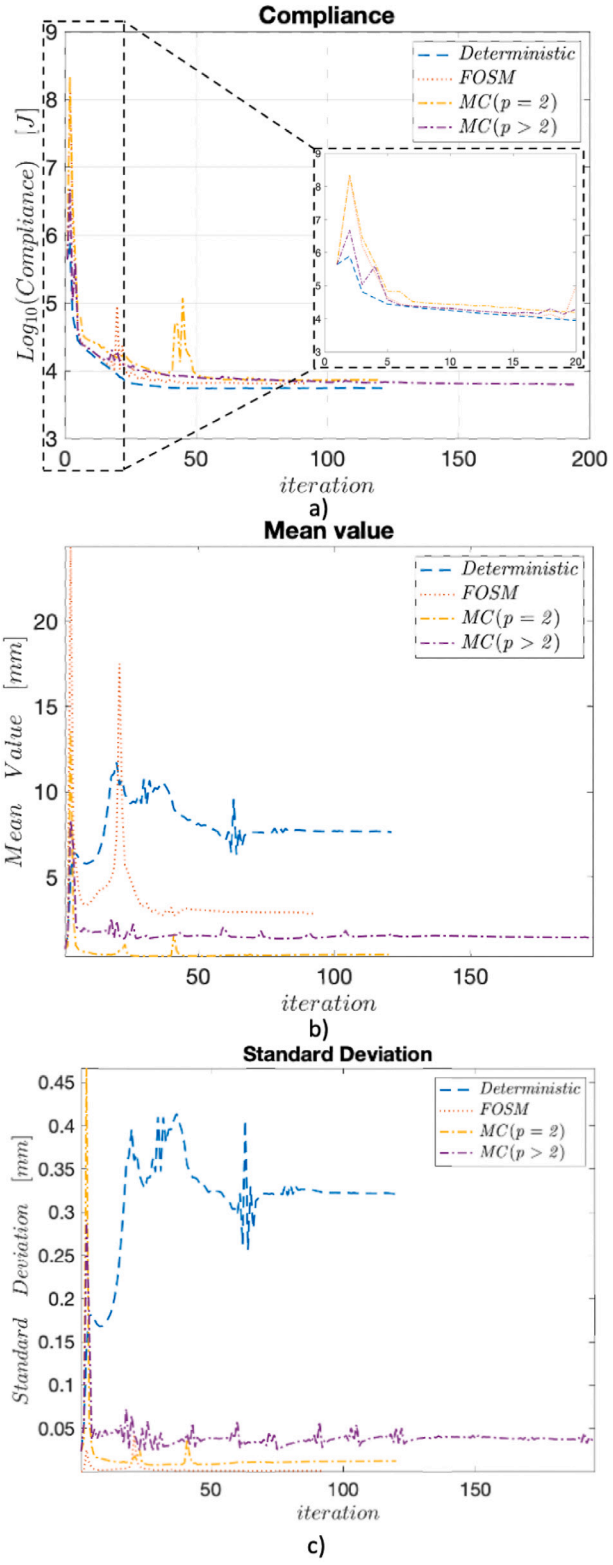

Fig. 7. Case 0° printing orientation: The evolution of (a) compliance, and (b) the mean value for the distortion quantity together with (c) standard deviation.

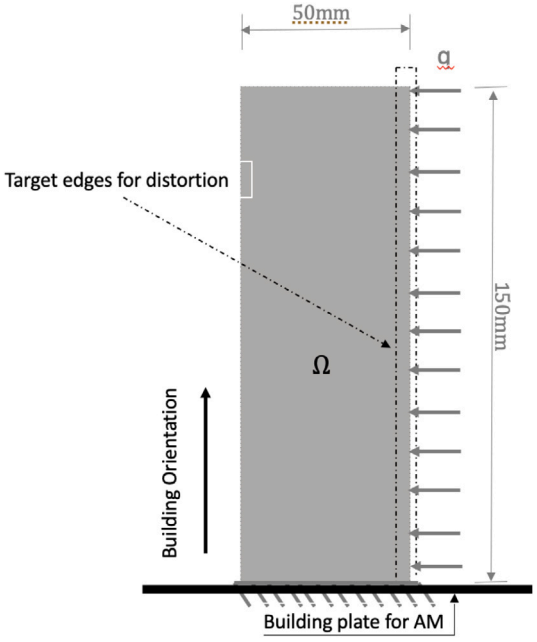

Fig. 8. Case 90° printing orientation: Design domain and the loading scheme. The distributed loading condition is indicated by gray color, and the strong black color marks the AM process.

standard deviation, respectively. *MC* ($p > 2$) also performs clearly better than *FOSM*, but with less statistical distortion quantity reductions of 21% and 46.4%, respectively.

6. Conclusion

Controlling process-induced part distortion is an important topic in design for Additive Manufacturing (AM). Failing to limit distortion in critical regions leads to unfit parts and even build failures. While AM process simulations such as the inherent strain method can be used to predict distortion, there is inherent uncertainty regarding the applicability of certain calibrated strains for a new part. This paper proposes the first method to address this challenge by including the inherent uncertainty within the AM process model into a multiobjective topology optimization, aimed at maximizing both part performance and robustness against process-induced distortion.

The inclusion of stochastic AM process simulation provides an enhanced and effective means to control part distortion and provides new avenues in robust design.

The two considered robust formulations, namely, Monte Carlo (MC) and First-Order-Second-Moment ($FSOM$), perform satisfactorily in optimizing a distortion measure simultaneously with structural compliance.

In this paper, the inherent strain magnitude in each deposited layer is chosen as a stochastic variable to demonstrate this concept. In combination with a linearized inherent strain-based AM process simulation, significant computational gains were obtained to limit the additional cost of a robust versus a deterministic optimization. The proposed approach implies the assumption that the random parameters are constant in each layer, and only vary in build direction. If a full

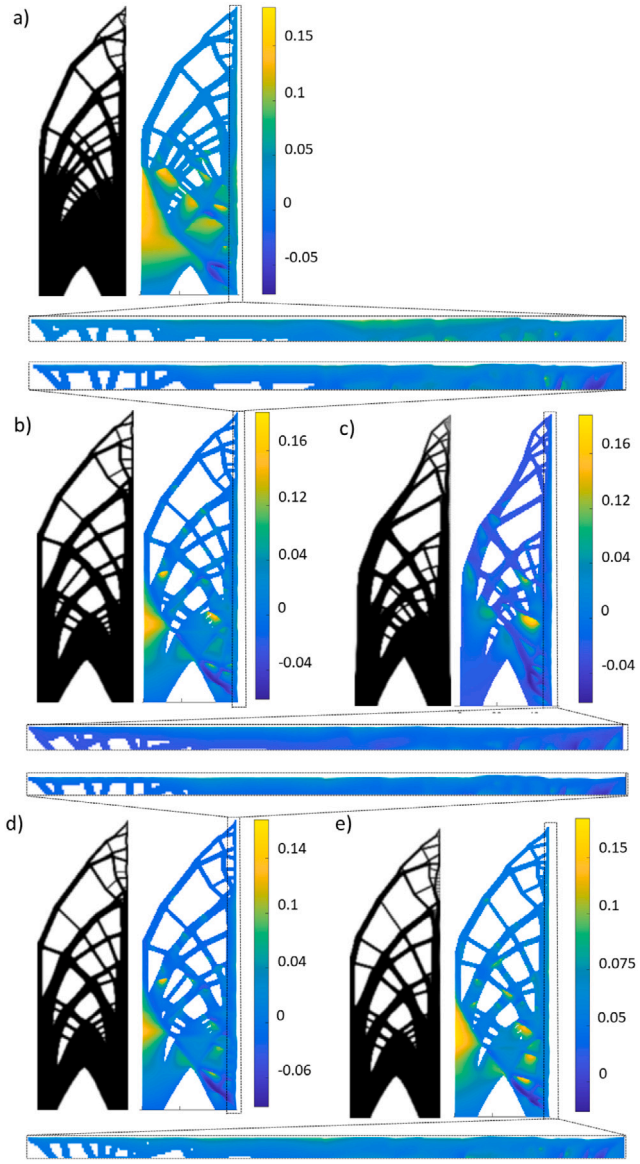

Fig. 9. The optimized design layout for respectively (a) Only compliance (b) Deterministic case; (c) *FOSM*; (d) *MC (p=2)*, (e) *MC (p > 2)* . On the right-hand side, the distortion from the fabrication process is shown, with distortion magnitude in mm. The AM process simulation is performed for a design domain orientated 90° with respect to the longest dimension.

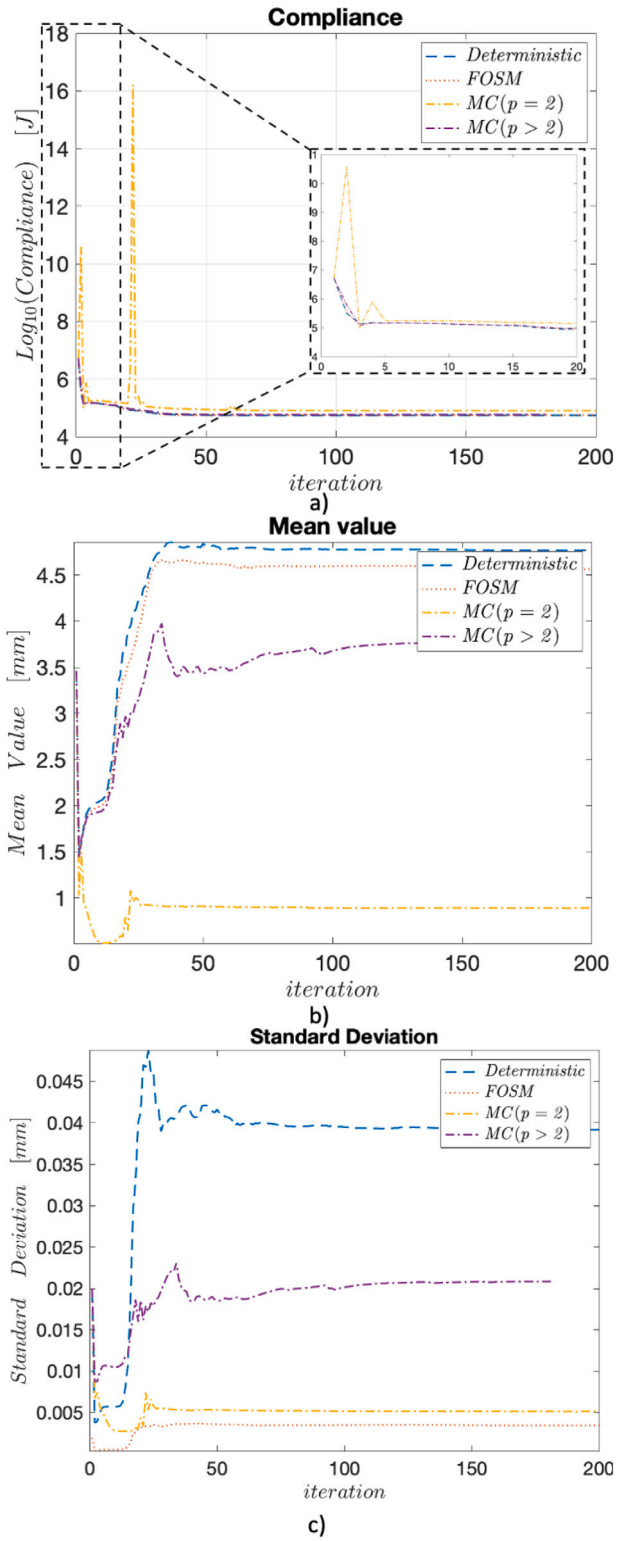

Fig. 10. Case 90° printing orientation: The evolution of (a) compliance, and (b) the mean value for the distortion quantity together with (c) standard deviation.

two- or three-dimensional random field is considered, we do not see an alternative to using plain Monte Carlo simulations.

The adversarial nature of compliance and part distortion induces the optimizer to frequently reconfigure the design layout during the initial iterations. It is observed that improvements in the distortion measure come with an increase in compliance.

Compared to a compliance-only design, all proposed methods are capable of achieving significant distortion reductions. Compared to a deterministic case, including robustness leads to both a lower mean and standard deviation of the distortion measure. Furthermore, it was found that the *FOSM* method for standard deviation estimate was considerably less accurate than an *MC* reference. This explains the less significant reductions in standard deviations obtained in robust optimization of the part topology using *FOSM*, relative to results using *MC*.

In terms of computational cost, each method requires the same number of systems of equations to be solved for the evaluation of the objective function, proportional to the number of layers N. Regarding the sensitivity analysis, *FOSM* requires the least number of additional systems of equations to be solved for the sensitivity, namely N. The MC cases, in general, require N^2 additional systems of equations to be solved for the sensitivities. An exception holds for the case where

the distortion measure is defined as a 2-norm. In that case, *MC* case is more computational efficient when the number of realizations m is smaller than the number of layers N, but this improvement comes at a memory cost.

This study involves only 2D examples due to the high computational effort associated with the general *MC* formulation. As the *MC* formulation using a 2-norm distortion measure showed good performance and reduced computational effort, as future work, it is recommended to develop a 3D version of the algorithm based on this approach. While robust optimization against part distortion is more expensive than a deterministic approach, large distortion reductions have been observed. Accounting for uncertainty in AM process models may well be a more practical direction than increasing the fidelity and computational cost of the models themselves.

CRediT authorship contribution statement

Ludian Komini: Methodology, Software, Writing, Visualization. **Matthijs Langelaar:** Conceptualization, Resources, Writing, Supervision. **Benedikt Kriegesmann:** Conceptualization, Writing, Supervision.

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors wish to express their gratitude to Professor Svanberg for generously providing his MMA code.

The first author would like to extend sincere courtesy to Professor Niko Pojani for generously shaping the first author's initial insights on the topic of optimization.

Appendix A

Here we derive the sensitivity for one realization using the adjoint method. Although the quantity of interest $\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)$ is dependent on both $\boldsymbol{x}$ and $\boldsymbol{\alpha}_j$, for the sake of brevity, we acknowledge this, and we keep these parameters outside the left-hand side of the equation notation. First, we decompose the functions $\Gamma(\boldsymbol{x}, \boldsymbol{\alpha}_j)$, and write their respective derivatives.

$$\begin{aligned}
&\Gamma_j = \left(\frac{1}{N_n}\right)^{1/p} (L_j)^{\frac{1}{p}} && \frac{\partial \Gamma_j}{\partial L_j} = \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p} (L_j)^{\frac{1}{p}-1} \\
&L_j = \mathbf{1}^T \boldsymbol{f}_j && \frac{\partial L_j}{\partial \boldsymbol{f}_j} = \mathbf{1}^T \\
&\boldsymbol{f}_j = \boldsymbol{v}_j^{\circ(p)} && \frac{\partial \boldsymbol{f}_j}{\partial \boldsymbol{v}_j} = p \boldsymbol{v}_j^{\circ(p-1)} \\
&\boldsymbol{v}_j = \boldsymbol{\Theta} \boldsymbol{u}_j && \frac{\partial \boldsymbol{v}_j}{\partial x_e} = \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j + \boldsymbol{\Theta} \frac{\partial \boldsymbol{u}_j}{\partial x_e} \\
&\boldsymbol{u}_j = \textstyle\sum_{k=1}^{N} \alpha_j^k \boldsymbol{u}^k && \frac{\partial \boldsymbol{u}_j}{\partial x_e} = \textstyle\sum_{k=1}^{N} \alpha_j^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}
\end{aligned} \tag{45}$$

Then, we use the chain rule to formulate the sensitivity w.r.t. design parameter x_e.

$$\frac{\partial \Gamma_j}{\partial x_e} = \frac{\partial \Gamma_j}{\partial L_j} \frac{\partial L_j}{\partial \boldsymbol{f}_j} \frac{\partial \boldsymbol{f}_j}{\partial \boldsymbol{v}_j} \circ \frac{\partial \boldsymbol{v}_j}{\partial x_e} \tag{46}$$

We then substitute the sub-parts of the chain rule, Eq. (45), to Eq. (46) and we get the following equation.

$$\begin{aligned}
\frac{\partial \Gamma_j}{\partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} p e^T \left(\boldsymbol{v}_j^{\circ(p-1)} \circ \frac{\partial \boldsymbol{v}_j}{\partial x_e}\right) \\
&= \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \mathbf{1}^T \left(\boldsymbol{v}_j^{\circ(p-1)} \circ \frac{\partial \boldsymbol{v}_j}{\partial x_e}\right)
\end{aligned} \tag{47}$$

where:

$$\frac{\partial \boldsymbol{v}_j}{\partial x_e} = \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j + \boldsymbol{\Theta} \frac{\partial \boldsymbol{u}_j}{\partial x_e} \tag{48}$$

$$\boldsymbol{\Theta} = \begin{cases} x_{node} & \text{for interface DOFs} \\ 0 & \text{otherwise} \end{cases} \tag{49}$$

$$\frac{\partial \boldsymbol{\Theta}}{\partial x_e} = \begin{cases} \frac{\partial x_{node}}{\partial x_e} & \text{for interface DOFs} \\ 0 & \text{otherwise} \end{cases} \tag{50}$$

Afterward, we build the adjoint formulation for one realization, the nominal simulation, upon which we will then superimpose all the other realizations.

$$\mathcal{L}_j = \Gamma_j + \sum_{k=1}^{N} \alpha_j{}^k \boldsymbol{\lambda}^{kT} \left(K^k \boldsymbol{u}^k - \boldsymbol{f}^k\right) \tag{51}$$

Then we proceed with further derivations w.r.t x_e.

$$\begin{aligned}
\frac{\partial \mathcal{L}_j}{\partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} p \mathbf{1}^T \left(\boldsymbol{v}_j^{\circ(p-1)} \circ \frac{\partial \boldsymbol{v}_j}{\partial x_e}\right) \\
&\quad + \left(\frac{1}{N_n}\right)^{1/p} \sum_{k=1}^{N} \alpha_j^k \boldsymbol{\lambda}_j^{kT} \left(\frac{\partial \boldsymbol{K}^k}{\partial x_e} \boldsymbol{u}^k + K^k \frac{\partial \boldsymbol{u}^k}{\partial x_e} - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right)
\end{aligned} \tag{52}$$

Note: The operation $\mathbf{1}^T \left(\boldsymbol{v}_j^{\circ(p-1)} \circ \frac{\partial \boldsymbol{v}_j}{\partial x_e}\right)$ can be rewritten as a normal vector multiplication $\boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{v}_j}{\partial x_e}$, thus we eliminate Hadamard product.

$$\begin{aligned}
\frac{\partial \mathcal{L}_j}{\partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{v}_j}{\partial x_e} \\
&\quad + \sum_{k=1}^{N} \alpha_j^k \boldsymbol{\lambda}_j^{kT} \left(\frac{\partial K^k}{\partial x_e} \boldsymbol{u}^k + \boldsymbol{K}^k \frac{\partial \boldsymbol{u}^k}{\partial x_e} - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right)
\end{aligned} \tag{53}$$

After rearrangements, we group according to $\alpha_j^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}$.

$$\begin{aligned}
&\frac{\partial \mathcal{L}_j}{\partial x_e} \\
&= \sum_{k=1}^{N} \underbrace{\left(\left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \boldsymbol{\Theta} + \boldsymbol{\lambda}_j^{kT} \boldsymbol{K}^k\right)}_{=0 \text{ adjoint equation}} \\
&\alpha_j^k \frac{\partial \boldsymbol{u}^k}{\partial x_e} \\
&+ \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j \\
&+ \sum_{k=1}^{N} \alpha_j^k \boldsymbol{\lambda}_j^{kT} \left(\frac{\partial \boldsymbol{K}^k}{\partial x_e} \boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right)
\end{aligned} \tag{54}$$

Solving the adjoint system of equations given in Eq. (55) at each layer k, for all the realizations j we get $\boldsymbol{\lambda}_j^k$.

$$\boldsymbol{K}^k \boldsymbol{\lambda}_j^k = -\left(\frac{1}{N_n}\right)^{1/p} \left(e^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{\Theta} \boldsymbol{v}_j^{\circ(p-1)} \tag{55}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}_j}{\partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \left(\mathbf{1}^T \boldsymbol{v}_j^{\circ(p)}\right)^{\frac{1}{p}-1} \boldsymbol{v}_j^{\circ(p-1)^T} \frac{\partial \boldsymbol{\Theta}}{\partial x_e} \boldsymbol{u}_j \\
&\quad + \sum_{k=1}^{N} \alpha_j^k \boldsymbol{\lambda}_j^{kT} \left(\frac{\partial \boldsymbol{K}^k}{\partial x_e} \boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right)
\end{aligned} \tag{56}$$

After computing the multipliers with Eq. (55), we substitute $\boldsymbol{\lambda}_j^k$ in Eq. (56), and eventually the final form of the sensitivity will be as obtained.

Appendix B

Here, we write the direct derivative of the variance σ_f^2 of Γ for the *FOSM* approach. To compute the variance σ_f^2, given in Eq. (57), we need $\frac{\partial \Gamma(\mu_\alpha)}{\partial \alpha^k}$.

$$\sigma_\Gamma^2 = \sum_{k_1=1}^{N}\sum_{k_2=1}^{N} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \boldsymbol{\alpha}^{k_1}} \frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial \boldsymbol{\alpha}^{k_2}} \operatorname{cov}\left(\alpha^{k_1}, \alpha^{k_2}\right) \tag{57}$$

Hence, we derive $\frac{\partial \Gamma}{\partial \alpha^k}$ using the chain rule

$$\begin{aligned}
&\Gamma = \left(\tfrac{1}{N_n}\right)^{1/p} (L)^{\frac{1}{p}} && \frac{\partial \Gamma}{\partial L} = \left(\tfrac{1}{N_n}\right)^{1/p} \tfrac{1}{p}(L)^{\frac{1}{p}-1}\\
&L = \mathbf{1}^T \boldsymbol{f} && \frac{\partial L}{\partial \boldsymbol{f}} = \mathbf{1}^T\\
&\boldsymbol{f} = \boldsymbol{v}^{\circ(p)} && \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}} = p\boldsymbol{v}^{\circ(p-1)}\\
&\boldsymbol{v} = \boldsymbol{\Theta}\boldsymbol{u} && \frac{\partial \boldsymbol{v}}{\partial \boldsymbol{u}} = \boldsymbol{\Theta}\\
&\boldsymbol{u} = \textstyle\sum_{k=1}^{N} \alpha^k \boldsymbol{u}^k && \frac{\partial \boldsymbol{u}}{\partial \alpha^k} = \boldsymbol{u}^k
\end{aligned} \tag{58}$$

$$\begin{aligned}
\frac{\partial \Gamma}{\partial \alpha^k} &= \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1} \frac{\partial L}{\partial \alpha^k}\\
\frac{\partial \Gamma}{\partial \alpha^k} &= \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1} \frac{\partial L}{\partial \boldsymbol{f}} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}} \circ \frac{\partial \boldsymbol{v}}{\partial \boldsymbol{u}} \frac{\partial \boldsymbol{u}}{\partial \alpha^k}
\end{aligned} \tag{59}$$

$$\frac{\partial \Gamma}{\partial \alpha^k} = \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1} \mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\boldsymbol{u}^k \tag{60}$$

As a next step, we compute $\frac{\partial \Gamma}{\partial x_e}$ using again the chain rule, and we substitute it in Eq. (60) to obtain $\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e}$, which is needed to solve the adjoint system in Eq. (73).

$$\frac{\partial \Gamma}{\partial x_e} = \frac{\partial \Gamma}{\partial L}\frac{\partial L}{\partial \boldsymbol{f}}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}} \circ \frac{\partial \boldsymbol{v}}{\partial x_e} \tag{61}$$

$$\begin{aligned}
&\Gamma = \left(\tfrac{1}{N_n}\right)^{1/p} (L)^{\frac{1}{p}} && \frac{\partial \Gamma}{\partial L} = \left(\tfrac{1}{N_n}\right)^{1/p} \tfrac{1}{p}(L)^{\frac{1}{p}-1}\\
&L = \mathbf{1}^T \boldsymbol{f} && \frac{\partial L}{\partial \boldsymbol{f}} = \mathbf{1}^T\\
&\boldsymbol{f} = \boldsymbol{v}^{\circ(p)} && \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}} = p\boldsymbol{v}^{\circ(p-1)}\\
&\boldsymbol{v} = \boldsymbol{\Theta}\boldsymbol{u} && \frac{\partial \boldsymbol{v}}{\partial x_e} = \frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u} + \boldsymbol{\Theta}\frac{\partial \boldsymbol{u}}{\partial x_e}\\
&\boldsymbol{u} = \textstyle\sum_{k=1}^{N} \alpha^k \boldsymbol{u}^k && \frac{\partial \boldsymbol{u}}{\partial x_e} = \sum_{k=1}^{N} \alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}
\end{aligned} \tag{62}$$

After we substitute the sub-parts of the chain rule, particularly Eq. (62), into Eq. (61) we get

$$\begin{aligned}
\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left(\frac{\partial L}{\partial x_e}\right)\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)}\\
&\circ \boldsymbol{\Theta}\boldsymbol{u}^k\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p\frac{\partial \boldsymbol{v}^{\circ(p-1)}}{\partial x_e} \circ \boldsymbol{\Theta}\boldsymbol{u}^k\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\frac{\partial \boldsymbol{u}^k}{\partial x_e}
\end{aligned} \tag{63}$$

Afterwards, we isolate $\alpha^k \sum_{k=1}^{N} \frac{\partial \boldsymbol{u}^k}{\partial x_e}$ and we get:

$$\begin{aligned}
&\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e} =\\
&\left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left[\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right]\\
&\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\boldsymbol{u}^k\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1}\left[\mathbf{1}^T p(p-1)\boldsymbol{v}^{\circ(p-2)} \circ \left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right]\\
&\circ \boldsymbol{\Theta}\boldsymbol{u}^k\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left(\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\boldsymbol{u}^k\right)\\
&\left[\mathbf{1}^{\mathrm{T}} \mathrm{p}\boldsymbol{v}^{\circ(p-1)} \circ \left(\boldsymbol{\Theta}\sum_{k=1}^{N}\alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}\right)\right]\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p(p-1)\boldsymbol{\Theta}\boldsymbol{u}^k \circ \boldsymbol{v}^{\circ(p-2)}\\
&\circ \left[\boldsymbol{\Theta}\sum_{k=1}^{N}\alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}\right]\\
&+ \left(\frac{1}{N_n}\right)^{1/p} \frac{1}{p}\underbrace{\delta_{rs}\frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\frac{1}{\alpha^k}\sum_{k=1}^{N}\alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}}_{=0 \quad for \quad r\neq s}
\end{aligned} \tag{64}$$

where:

$$\delta rs = \begin{cases} 1 & \text{if } r = s \\ 0 & \text{otherwise} \end{cases} \tag{65}$$

Because we want to isolate $\frac{\partial \boldsymbol{u}^k}{\partial x_e}$ to solve the adjoint equation we rewrite Eq. (65) while grouping relevant terms for our purpose.

$$\begin{aligned}
\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e} &= \left(\frac{1}{N_n}\right)^{1/p}\\
&\underbrace{\left\{\begin{array}{l} \frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left[\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right]\\ \mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\boldsymbol{u}^k\\ + \frac{1}{p}(L)^{\frac{1}{p}-1}\left[\mathbf{1}^T p(p-1)\boldsymbol{v}^{\circ(p-2)} \circ \left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right] \circ \boldsymbol{\Theta}\boldsymbol{u}^k \end{array}\right\}}_{\boldsymbol{A}_s}\\
&+ \underbrace{\left\{\begin{array}{l} \frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left(\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\boldsymbol{u}^k\right)\\ \left(\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}\right)\\ + \left[\begin{array}{l} \frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p(p-1)\boldsymbol{\Theta}\boldsymbol{u}^k \circ \boldsymbol{v}^{\circ(p-2)} \circ \boldsymbol{\Theta}\\ + \underbrace{\delta_{rs}\frac{1}{\alpha^k}\frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)} \circ \boldsymbol{\Theta}}_{=0 \text{ for } r\neq s} \end{array}\right] \end{array}\right\}}_{\boldsymbol{B}_s}\\
&\left(\frac{1}{N_n}\right)^{1/p}\sum_{k=1}^{N}\alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e}
\end{aligned} \tag{66}$$

Hence, we observe the compact form of $\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e}$.

$$\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial \alpha^k \partial x_e} = \left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{A}_s + \left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{B}_s\sum_{k=1}^{N}\alpha^k \frac{\partial \boldsymbol{u}^k}{\partial x_e} \tag{67}$$

where $\boldsymbol{A}_s$ and $\boldsymbol{B}_s$ are given respectively in Eq. (69) and Eq. (71). Since $\frac{\partial \boldsymbol{u}^k}{\partial x_e}$ is expensive, we use the adjoint method for the variance.

$$\boldsymbol{A}_s = \left\{ \begin{array}{l} \underbrace{\frac{1}{p}\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left[\mathbf{1}^T p\boldsymbol{v}^{p-1}\circ\left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right] \mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)}\circ\boldsymbol{\Theta}\boldsymbol{u}^k}_{\boldsymbol{A}_1} \\ \underbrace{+\frac{1}{p}(L)^{\frac{1}{p}-1}\left[\mathbf{1}^T p(p-1)\boldsymbol{v}^{\circ(p-2)}\circ\left(\frac{\partial \boldsymbol{\Theta}}{\partial x_e}\boldsymbol{u}\right)\right] \circ\boldsymbol{\Theta}\boldsymbol{u}^k}_{\boldsymbol{A}_2} \end{array} \right\} \tag{68}$$

$$\boldsymbol{A}_s = \boldsymbol{A}_1 + \boldsymbol{A}_2 \tag{69}$$

$$\boldsymbol{B}_s = \left\{ \begin{array}{l} \underbrace{p\left(\frac{1}{p}-1\right)(L)^{\frac{1}{p}-2}\left(\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)}\circ\boldsymbol{\Theta}\boldsymbol{u}^k\right) \left(\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)}\circ\boldsymbol{\Theta}\right)}_{\boldsymbol{B}_1} \\ + \left[\begin{array}{l} \underbrace{\frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p(p-1)\boldsymbol{\Theta}\boldsymbol{u}^k\circ\boldsymbol{v}^{\circ(p-2)}\circ\boldsymbol{\Theta}}_{\boldsymbol{B}_2} \\ \underbrace{+\delta_{rs}\frac{1}{\alpha^k}\frac{1}{p}(L)^{\frac{1}{p}-1}\mathbf{1}^T p\boldsymbol{v}^{\circ(p-1)}\circ\boldsymbol{\Theta}}_{=0 \text{ for } r\neq s} \end{array} \right]_{\boldsymbol{B}_3} \end{array} \right\} \tag{70}$$

$$\boldsymbol{B}_s = \boldsymbol{B}_1 + \boldsymbol{B}_2 + \boldsymbol{B}_3 \tag{71}$$

Appendix C

Given the augmented formulation $\chi(\boldsymbol{x},\boldsymbol{\alpha})$ of variance ${\sigma_I}^2(\boldsymbol{x})$ computed using the adjoint method:

$$\begin{aligned} \chi(\boldsymbol{x},\boldsymbol{\alpha}) &= \sigma_I^2(\boldsymbol{x}) + \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\boldsymbol{K}^k\boldsymbol{u}^k - \boldsymbol{f}^k\right) \\ &= \sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_1}}\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\boldsymbol{K}^k\boldsymbol{u}^k - \boldsymbol{f}^k\right) \end{aligned} \tag{72}$$

We differentiate χ w.r.t. the design parameter x_e:

$$\begin{aligned} \frac{\partial \chi(\boldsymbol{x},\boldsymbol{\alpha})}{\partial x_e} &= 2\sum_{k_1=1}^{\mathrm{N}}\sum_{k_2=1}^{N}\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial x_e\partial\alpha^{k_1}}\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\frac{\partial \boldsymbol{K}^k}{\partial x_e}\boldsymbol{u}^k + \boldsymbol{K}^k\frac{\partial \boldsymbol{u}^k}{\partial x_e} - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \end{aligned} \tag{73}$$

In Eq. (73) we substitute $\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^k\partial x_e} = \left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{A}_s + \left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{B}_s\sum_{k=1}^{N}\alpha^k\frac{\partial \boldsymbol{u}^k}{\partial x_e}$, given in Eq. (66) in Appendix A, for the mixed partial derivative $\frac{\partial^2 \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^k\partial x_e}$.

$$\begin{aligned} \frac{\partial \chi(\boldsymbol{x},\boldsymbol{\alpha})}{\partial x_e} &= 2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\left(\boldsymbol{A}_s + \boldsymbol{B}_s\sum_{k=1}^{N}\alpha^k\frac{\partial \boldsymbol{u}^k}{\partial x_e}\right) \\ &\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\frac{\partial \boldsymbol{K}^k}{\partial x_e}\boldsymbol{u}^k + \boldsymbol{K}^k\frac{\partial \boldsymbol{u}^k}{\partial x_e} - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \end{aligned} \tag{74}$$

Afterwards, we rearrange Eq. (74).

$$\begin{aligned} \frac{\partial \chi(\boldsymbol{x},\boldsymbol{\alpha})}{\partial x_e} =& 2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{A}_s\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\frac{\partial \boldsymbol{K}^k}{\partial x_e}\boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \\ &+ 2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{B}_s\left(\sum_{k=1}^{N}\alpha^k\frac{\partial \boldsymbol{u}^k}{\partial x_e}\right) \\ &\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\boldsymbol{K}^k\frac{\partial \boldsymbol{u}^k}{\partial x_e}\right) \end{aligned} \tag{75}$$

Here we isolate $\sum_{k=1}^{N}\alpha^k\frac{\partial \boldsymbol{u}^k}{\partial x_e}$.

$$\begin{aligned} \frac{\partial \chi(\boldsymbol{x},\boldsymbol{\alpha})}{\partial x_e} =& 2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{A}_s\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\frac{\partial \boldsymbol{K}^k}{\partial x_e}\boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \\ &+ \left[\underbrace{2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{B}_s\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) + {\lambda^k}^T\boldsymbol{K}^k}_{=0 \text{ (adjoint equation)}}\right] \\ &\sum_{k=1}^{N}\alpha^k\frac{\partial \boldsymbol{u}^k}{\partial x_e} \end{aligned} \tag{76}$$

Demanding the term in brackets to equal zero provides the adjoint system to be solved. In the end, we obtain the sensitivity of $\sigma_I^2(\boldsymbol{x})$.

$$\begin{aligned} \frac{\partial \sigma_I^2(\boldsymbol{x})}{\partial x_e} &= \frac{\partial \chi(\boldsymbol{x},\boldsymbol{\alpha})}{\partial x_e} = \\ &2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\left(\frac{1}{N_n}\right)^{1/p}\boldsymbol{A}_s\frac{\partial \Gamma(\boldsymbol{\mu_\alpha})}{\partial\alpha^{k_2}}\operatorname{cov}\left(\alpha^{k_1},\alpha^{k_2}\right) \\ &+ \sum_{k=1}^{N}\alpha^k{\lambda^k}^T\left(\frac{\partial \boldsymbol{K}^k}{\partial x_e}\boldsymbol{u}^k - \frac{\partial \boldsymbol{f}^k}{\partial x_e}\right) \end{aligned} \tag{77}$$

and $\boldsymbol{\lambda}^k$ is obtained from the solution of Eq. (44) the adjoint system of equations.

Appendix D

Let us consider two formulations for the objective function namely, p-norm Y and p-mean Γ given respectively in Eqs. (78) and (79):

$$Y = \left(\mathbf{1}^T\boldsymbol{v}^{\circ(p)}\right)^{1/p} \tag{78}$$

$$\Gamma = \left(\frac{1}{N_n}\mathbf{1}^T\boldsymbol{v}^{\circ(p)}\right)^{1/p} \tag{79}$$

Here N_n is the number of nodes taken into consideration, which in the examples considered are the nodes along the vertical edges of the design space. For p-norm case, we have observed an increase in the distortion as we refine the mesh, i.e., more nodal contributions are summed, which is an artefact. Therefore, to solve this issue, we simply divide the p-norm quantity by the number of nodes, so we decouple the distortion quantity from the mesh refinement and call this new formulation p-mean.

References

[1] Plocher J, Panesar A. Review on design and structural optimisation in additive manufacturing: Towards next-generation lightweight structures. Mater Des 2019;183:108164.

[2] Zegard T, Paulino GH. Bridging topology optimization and additive manufacturing. Struct Multidiscip Optim 2015;53:175–92.

[3] Langelaar M. Topology optimization of 3D self-supporting structures for additive manufacturing. Addit Manuf 2016;12:60–70.

[4] van de Ven E, Maas R, Ayas C, Langelaar M, van Keulen F. Overhang control based on front propagation in 3D topology optimization for additive manufacturing. Comput Methods Appl Mech Engrg 2020;369:113169.

[5] Liu Y, Zhou M, Wei C, Lin Z. Topology optimization of self-supporting infill structures. Struct Multidiscip Optim 2021;63(5):2289–304.

[6] Munro D, Ayas C, Langelaar M, van Keulen F. On process-step parallel computability and linear superposition of mechanical responses in additive manufacturing process simulation. Addit Manuf 2019;28:738–49.

[7] Allaire G, Jakabcin L. Taking into account thermal residual stresses in topology optimization of structures built by additive manufacturing. Math Models Methods Appl Sci 2018.

[8] Allaire G, Bihr M, Bogosel B. Support optimization in additive manufacturing for geometric and thermo-mechanical constraints. Struct Multidiscip Optim 2020;61:2377–99.

[9] Pellens J, Lombaert G, Michiels M, Craeghs T, Schevenels M. Topology optimization of support structure layout in metal-based additive manufacturing accounting for thermal deformations. Struct Multidiscip Optim 2020;61:2291–303.

[10] Wang WM, Munro D, Wang CCL, van Keulen F, Wu J. Space-time topology optimization for additive manufacturing. Struct Multidiscip Optim 2020;61:1–18.

[11] Misiun G, van de Ven E, Langelaar M, Geijselaers H, van Keulen F, van den Boogaard T, Ayas C. Topology optimization for additive manufacturing with distortion constraints. Comput Methods Appl Mech Engrg 2021;386:114095.

[12] Gholaminezhad I, Assimi H, Jamali A, Vajari DA. Uncertainty quantification and robust modeling of selective laser melting process using stochastic multi-objective approach. Int J Adv Manuf Technol 2016;86:1425–41.

[13] Schuëller G, Jensen H. Computational methods in optimization considering uncertainties – An overview. Comput Methods Appl Mech Engrg 2008;198(1):2–13, Computational Methods in Optimization Considering Uncertainties.

[14] Elishakoff I, Kriegesmann B, Rolfes R, Hühne C, Kling A. Optimization and antioptimization of buckling load for composite cylindrical shells under uncertainties. AIAA J 2012;50:1513–24.

[15] Wang L, Liang J, Wu D. A non-probabilistic reliability-based topology optimization (NRBTO) method of continuum structures with convex uncertainties. Struct Multidiscip Optim 2018;58(6):2601–20.

[16] Schevenels M, Lazarov B, Sigmund O. Robust topology optimization accounting for spatially varying manufacturing errors. Comput Methods Appl Mech Engrg 2011;200(49–52):3613–27.

[17] Lazarov B, Schevenels M, Sigmund O. Topology optimization considering material and geometric uncertainties using stochastic collocation methods. Struct Multidiscip Optim 2012;46(4):597–612.

[18] Kriegesmann B, Lüdeker J. Robust compliance topology optimization using the first-order second-moment method. Struct Multidiscip Optim 2019;60.

[19] Maute K, Frangopol DM. Reliability-based design of MEMS mechanisms by topology optimization. Comput Struct 2003;81(8):813–24, K.J Bathe 60th Anniversary Issue.

[20] Kharmanda G, Olhoff N, Lemaire M. Reliability-based topology optimization. Struct Multidiscip Optim 2004;26:295–307.

[21] Liebisch M, Hein R, Wille T. Probabilistic process simulation to predict process induced distortions of a composite frame. CEAS Aeronaut. J. 2018;9.

[22] Bendsøe M, Sigmund O. Topology optimization: Theory, methods and applications. Springer; 2004.

[23] Bugatti M, Semeraro Q. Limitations of the inherent strain method in simulating powder bed fusion processes. Addit Manuf 2018;23:329–46.

[24] Bathe K-J. Finite element procedures: Klaus-Jurgen Bathe. NJ Klaus-Jürgen Bathe, cop: Englewood Cliffs; 2006.

[25] Svanberg K. The method of moving asymptotes—a new method for structural optimization. Internat J Numer Methods Engrg 1987;24(2):359–73.