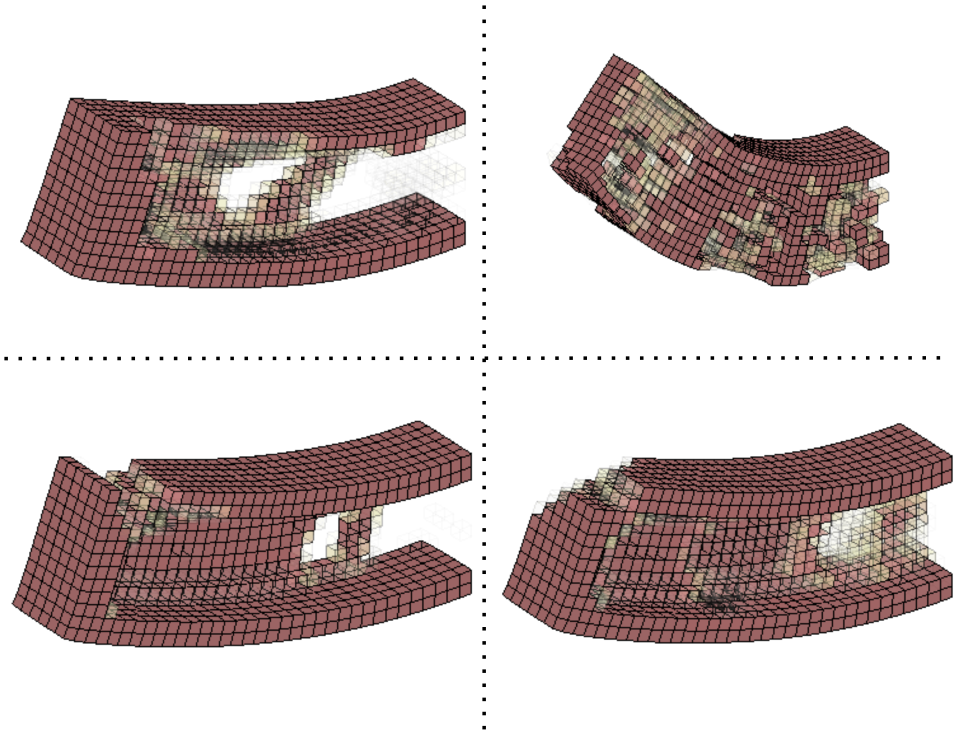
A Topology Optimization Framework considering Sintering in Design for Additive Manufacturing



Master Thesis
Christian Troelsgaard & Frederik Tobias Elmstrøm
Design of Mechanical Systems
Aalborg University
31/05/2024

AALBORG UNIVERSITY
STUDENT REPORT

The Faculty of Engineering and Science
Mechanical Engineering and Manufacturing
Fibigerstræde 16
9220 Aalborg Øst
http://www.mp.aau.dk

Title:

A Topology Optimization Framework considering Sintering in Design for Additive Manufacturing

Semester:

4^{th} Semester

Semester theme:

Master thesis

ECTS:

30 Points

Project period:

01/02 - 31/05 2024

Supervisors:

Erik Lund | el@mp.aau.dk

Project members:

Frederik Tobias Elmstrøm
Christian Troelsgaard

Issue: 1
Pages: 80
Date of submission: 31/05-2024

Signatures

Christian Troelsgaard

Christian Troelsgaard

Frederik Tobias Elmstrøm

Abstract

3D printing is a well-known and proven rapid prototyping tool often used to develop novel products. There is, however, an industrial desire to transform 3D printing from a rapid prototyping technology into an industrially viable manufacturing process. The development of 3D printing technology and materials has realized new manufacturing capabilities, including metal binder jetting 3D printing. This new technique is not matured and ready for industrial application due to, e.g., pitfalls regarding part tolerances. The metallic parts printed by the metal binder jetting process are porous, brittle and weak, and are strengthened and densified through a thermal sintering process. During the sintering process, the printed parts shrink and densify due to diffusional mechanisms activated by sintering. The parts further deform due to viscoplasticity induced by the elevated temperatures of sintering and distort undesirably due to manufacturing process variations. To counteract these undesired deformations and distortions the designed parts are compensated before printing.

Under the hypothesis that 3D printed parts that have to be compensated less, better adhere to the desired tolerances after sintering, a custom topology optimization tool has been developed to redesign or strengthen already designed parts against sintering-induced deformation. To realize the topology optimization tool, the deformations are modeled through a 3D geometrically non-linear finite element analysis combined with a non-linear viscoplastic semi-implicit material model. The quadratic convergence properties of the applied Newton-Raphson solver are kept by updating the tangent stiffness consistently with the stress. The non-linear finite element analysis is implemented in a custom MATLAB program, which has been compared against a 1D analytical solution and the commercial program ANSYS. The topology optimization is performed using the gradient-based MMA optimizer, linear and projection filtering along with continuation strategies. The design sensitivities of the topology optimization problem are analytically derived using the path-dependent adjoint method. The derivation and implementation of the design sensitivities in the MATLAB program are validated by forward difference approximation.

Topology optimization is performed on the well-known MBB beam benchmark geometry and an industrial test case where an already designed part has to be further strengthened. Since topology optimization within sintering is a novel field, four objective functions are proposed and tested. It was shown that three of the objective functions resulted in designs that reduced deformation and distortion for the MMB beam. The industrial test case was performed with two different objective functions which returned two similar designs where only small differences can be recognized. Determining which objective function to use might require further experimental analysis.

Contents

Preface

Literature is referenced to via the Harvard method. In-text references are sourced by Last name [Year] whereas references for sections are referenced to by [Last name, year]. If a reference is placed after the last period in a text section and the following section of text starts on a new line, the reference applies to the entire text section, e.g.:. [Last name, year] References to equations are given when referring to the equation. For tables and figures, references are given in the table or figure caption. A list of references is supplied on the last pages.

Tables and figures are numbered via their corresponding chapter, i.e. 'Tab 5.4' is the fourth table in Chapter 5. Equations are referred to by e.g. 'Eq. (4.2)' and figures as 'Fig. 3.2'.

Symbols, abbreviations, and mathematical notation are listed on the following pages. The list is not extensive, and other symbols or abbreviations may be introduced. Furthermore, subscripts may differentiate between the same symbol used multiple times. Some symbols are repeated but are then generally used in different chapters.

Thanks to Grundfos and Sintex for their collaboration during this project, especially Tom Jæger and Frederik Skafsgaard for their insights and assistance.

Present work has been supervised by Professor Erik Lund at Aalborg University.

Nomenclature

Notation

Δx	Increment of x.
$\dot{x}$	First time derivative of x.
$\frac{\partial f}{\partial x}$	Partial derivative of f wrt. x.
$\frac{df}{dx}$	Full derivative of f wrt. x.
∞	Infinity.
$[\,]$	Unit or matrix.
$\lVert x \rVert_2$	2-norm of x.
$\mathcal{A}$	Assembly operator.
$\mathcal{O}(n)$	Order of accuracy.
$\square^T$	Transpose of vector or matrix.
$\square'$	Deviatoric part of tensor.
$\tilde{\tilde{x}}$	Projection filtered design variable.
$\tilde{x}$	Linear Pseudo-Density filtered design variable.
$\{\,\}$	Vector.
f	Arbitrary function.

Symbols introduced in chapter 3

α	Surface energy.
η	Uniaxial viscosity.
ν	Poisson's ratio.
ϕ	Shear viscosity moduli.
ψ	Bulk viscosity moduli.
σ	Cauchy stress.
ε	Green Lagrange strain.
A	Pre-exponential constant.
d	Relative density.
E	Young's modulus.
G	Grain-size.
P_L	Sintering stress.
Q	Activation energy.
R	Universial gas constant.
T	Temperature.

Symbols introduced in chapter 4

(η, ξ, ζ)	Natural coordinatesystem.
(x, y, z)	Global coordinatesystem.
$\left[\hat{\mathbf{S}}\right]$	Stress matrix.
$[\mathbf{A}]$	Deformation derivative matrix.
$[\mathbf{B}]$	Kinematic matrix.
$[\mathbf{D}]$	Material stiffness matrix.
$[\mathbf{F}]$	Deformation gradient.
$[\mathbf{G}]$	Shape function derivative matrix.

$[\mathbf{H}]$	Boolean matrix.
$[\mathbf{I}]$	Identity matrix.
$[\mathbf{K}]$	Global stiffness matrix.
$[\mathbf{k}]$	Element stiffness matrix.
$[\mathbf{N}]$	Shape functions.
$\{\boldsymbol{\theta}\}$	Deformation derivative vector.
$\{\mathbf{a}\}$	Element nodal displacements.
$\{\mathbf{F}\}$	Global force vector.
$\{\mathbf{f}\}$	Element force vector.
$\{\mathbf{R}\}$	Global residual.
$\{\mathbf{r}\}$	Element global residual.
$\{\mathbf{S}\}$	2nd Piola-Kirchhoff stress.
$\{\mathbf{U}\}$	Global nodal displacements.
$\{\mathbf{u}\}$	Continuous displacements.
$\{\mathbf{x}^\star\}$	Deformed configuration.
$\{\mathbf{x}\}$	Reference configuration.
w	Gauss-integration weight.

Symbols introduced in chapter 5

ω	Smoothing constant.
φ	Smoothing function.
$\{\boldsymbol{\kappa}\}$	Internal variables.
$\{\mathbf{H}\}$	Material residual.
$\{\mathbf{q}\}$	Pseudo-load vector.
t	Time.

Symbols introduced in chapter 6

β	Projection parameter.
η	Projection cut-off.
κ	Interpolation gradients.
ρ	Material bulk density.
$\{\mathbf{g}\}$	Inequality constraints.
$\{\mathbf{x}\}$	Design variables.
C	Compliance.
F	Objective function.
G	Filter weight factor.
p	Material penalization factor.
r	Filter radius.
V	Element volume.

Symbols introduced in chapter 7

$\hat{f}$	Langrangian function.
$[\boldsymbol{\Theta}]$	Deviatoric operator.
$\{\boldsymbol{\lambda}\}, \{\boldsymbol{\mu}\}$	Lagrange multipliers.
$\{\mathbf{A}\}$	Trace vector.
$\{\mathbf{c}\}$	Local variables.
v	Interpolation function.

Abbreviations

CPU	Computational Processing Unit.
DOF	Degree Of Freedom.
FDA	Forward Difference Approximation.
FEA	Finite Element Analysis.

GCMMA	Globally Convergent Method of Moving Asymptotes.
GNLFEA	Geometrically Non Linear Finite Element Analysis.
LHS	Left Hand Side.
MBB	Messerschmitt-Bölkow-Blohm.
MBJ	Metal Binder Jetting.
MMA	Method of Moving Asymptotes.
NLFEA	Non Linear Finite Element Analysis.
NLPI	Non-Linear Penalization or Interpolation.
NR	Newton-Raphson.
PD	Pseudo Density.
RAMP	Rational Approximation of Material Parameters.
RD	Relative Density.
RHS	Right Hand Side.
SIMP	Solid Isotropic Material with Penalization.
SOVS	Skorohod-Olevsky Viscous Sintering.
TO	Topology Optimization.

1 Introduction

The ability to selectively build a given part from a material feedstock in a layerwise manner has been possible since the 1980's. This technology is often called Three-Dimensional (3D) Printing, layered manufacturing, rapid prototyping or more recently additive manufacturing. The first 3D printers only printed with polymeric materials and the specimens they could manufacture were prototypes, presentation parts or proof-of-concept parts due to their poor mechanical properties. As the 3D printing technology has matured more materials are available to print including metals, short- and continuous fiber-reinforced polymers and ceramics. The continuous development of 3D printers and their materials means that a transition from rapid prototyping to functional additive manufacturing has occurred and 3D printed parts are now included in commercial products. [Bandyopadhyay and Bose, 2016]
Polymeric 3D printing is also available for hobby use due to continuous development and reduction in the price of 3D printing hardware and polymeric material.

The possibility of 3D printing with metals has freed the design constraints that have traditionally been present for metal parts which previously have been manufactured by subtractive processes such as drilling, milling and turning. Since 3D printing can manufacture designs not traditionally available it opens the door for engineers to use more exotic design tools like topology optimization to design structures with high stiffness- and strength-to-mass ratio. These exotic designs can include internal voids or lattice structures that traditionally have been impossible to manufacture. High-performance structures such as 3D printed topology optimized structures are of interest in the aerospace and automotive industries but also in the dental and medical industries the prospect of designing and manufacturing user-customized implants is promising. [Bandyopadhyay and Bose, 2016]

A specific type of metal 3D printing is called Metal Binder Jetting (MBJ). The MBJ 3D printing process is attractive due to its relatively low cost [Li et al., 2020] while having high productivity [Zhang et al., 2021]. The MBJ process does not need supporting structures to print overhanging structures, like most other metal 3D printing processes, due to its printing process [Zhang et al., 2021]. Hence a costly post-processing step of removing integrated supporting structures can be skipped for MBJ 3D printing. This project's collaborative partner Grundfos is interested in exploring new methods to improve the geometrical accuracy of MBJ 3D printed parts. Thus this work will solely be focused on 3D printing with the MBJ process. In the following, an introduction will be given to the MBJ process and some of the pitfalls currently limiting its applicability.

1.1 Metal binder jetting

The primary mechanism of 3D printing by binder jetting is the binding of loose powder to a single structure. The powder can be both polymeric or metallic. The binder is a fluid and acts similarly to glue by bonding the metallic powder into a structure.

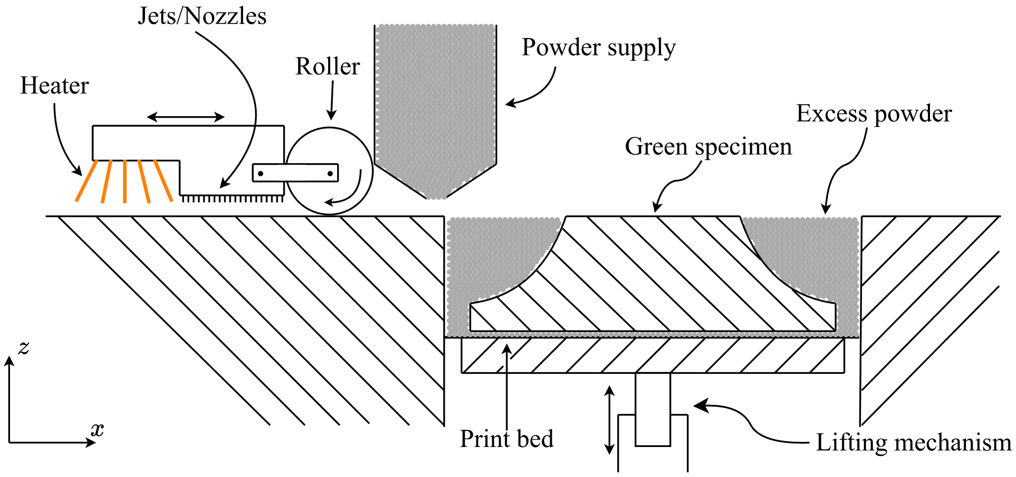


Figure 1.1. A sketch of the primary internal components in a MBJ 3D printer

In Fig. 1.1 the internal components of a MBJ 3D printer have been sketched. The printing process performed by the MBJ 3D printer is a sequence of layer-wise actions that are repeated throughout the z-direction. Initially, the print bed is lowered by an amount equal to the desired layer thickness. A layer of metallic powder is then distributed on the print bed. The roller ensures the powder-layer thickness is uniform throughout the print volume. The jets (or nozzles) then selectively deposit small volumes of binder fluid on the loose powder layer. The wetted metallic powder is then briefly heated to increase the strength of the printed part by curing the binder fluid. This layer-wise process is then repeated until the entire part has been printed. The as-printed part is often called a green part and is both porous and brittle. The green part has a Relative Density (RD) of 50%-65% when manufactured by MBJ. The RD is a measure of the volume embodied by metallic powder in the green part. When the part has been printed, the green specimen is removed from the printer and any excess metallic powder is removed in a process called de-powdering. [Borujeni et al., 2022]
Due to the low RD of the green part and the low strength of the binder, de-powdering is a delicate process and care must be taken not to damage or deform the green specimen.

To strengthen the green part it goes through a thermal treatment called sintering which is further discussed in Chap. 3. The sintering process aims to densify and increase the strength of the 3D-printed part [Borujeni et al., 2022]. Sintering is performed in furnaces that can heat the 3D printed part to temperatures near the materials' melting point. Ideally, the 3D-printed part should reach the bulk density of the constituent material. Since sintering is a mass-conserving process, the increase in density must be reacted by a decrease in the volume of the printed part. Thus, the green parts' external dimensions shrink during sintering as the RD increases. In addition to deformation due to shrinkage, undesired deformation and distortion can occur during sintering due to viscoplasticity, inhomogeneous initial RD distribution, uneven heating and non-uniform friction between the 3D-printed part and the sintering setter [Borujeni et al., 2022]. The sintering setter is typically a flat ceramic plate on which the 3D-printed specimens are placed during sintering.

1.1.1 Sintering deformation predictability

Currently, designing and manufacturing a part with 3D MBJ printing is a multi-step process with many concurrent design modifications based on deformation feedback from the printed and sintered specimens. This process is sketched in Fig. 1.2. This process is costly in both time and resources since the result of a design modification has to be printed and sintered before it can be evaluated. Therefore, it is proposed to pursue automatic design modifications and thereby (partly) remove the heuristic manual design modifications.

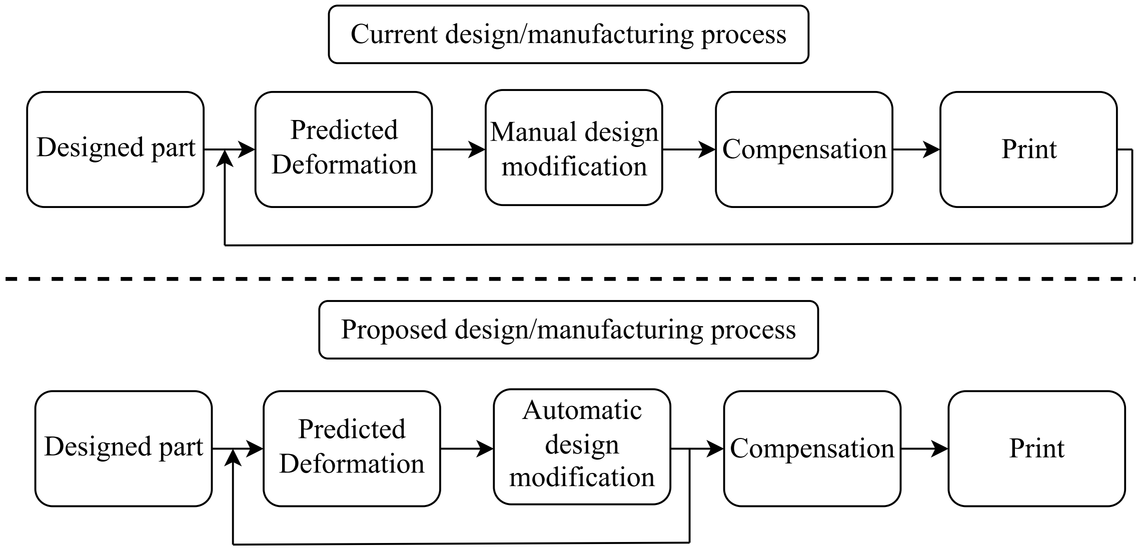

Current design/manufacturing process
Designed part
Predicted Deformation
Manual design modification
Compensation
Print
Proposed design/manufacturing process
Designed part
Predicted Deformation
Automatic design modification
Compensation
Print

Figure 1.2. Flowcharts of two different design/manufacturing processes

Numerical modeling of the deformation induced during sintering has proven difficult to predict. Several authors contribute low predictability to anisotropic shrinkage, caused due to inhomogeneous green density [Lee et al., 2022][Borujeni et al., 2023][Rios et al., 2021]. Anisotropic shrinkage during sintering can cause inaccurate predictions of geometric features, such as holes [Zago et al., 2021]. In general, larger discrepancies are seen between shrinkage in the build direction (along the z-axis in Fig. 1.1), compared to in-plane. In Borujeni et al. [2023], the accuracy of deformation compensation for several different specimens with distinct geometric features was analyzed. By taking the mean deformation compensation of the specimen and comparing it to the percentage of the final sintered geometry that complies with the tolerances, Fig. 1.3 can be constructed.

As seen from Fig. 1.3, there is an approximately inversely linear correlation between the mean compensation and the percentage of the geometry that complies with tolerances. This is further indicated in the works by Zhang et al. [2021], where the geometries with

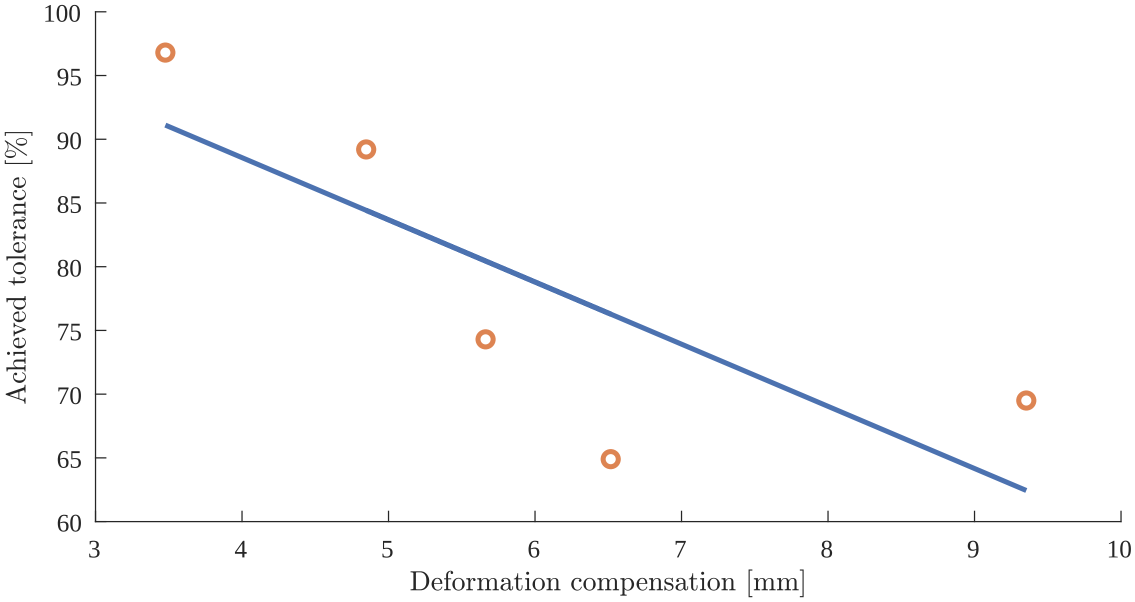


Figure 1.3. Correlation between achieved tolerance and mean deformation compensation. Data from Borujeni et al. [2023].

the least amount of overhang (and hereby, the least amount of deformation) showed a better correlation with the Finite Element Analysis (FEA) predicted deformation.

A proposed design process is also sketched in Fig. 1.2 where based on the predicted deformation, an algorithm can make changes to the designed part automatically. The changes should follow the geometry observations from Zhang et al. [2021], i.e. if the geometry responds to sintering with lower deformation, the predicted deformation is better. A type of algorithmic 'automatic' design change could be topology optimization used to determine e.g. the stiffest structure. Topology optimization is discussed further in Chap. 6. The proposed process could result in fewer resources wasted on printing and sintering parts that are not accepted due to excessive deformation or warping and could be less time-consuming. To increase the industrial relevance of MBJ, accurately predicting deformations and complying with tolerances during sintering is of interest. To the author's knowledge, no study has currently been conducted on reducing the required compensation of an MBJ-printed component, by altering the geometry.

1.2 Summary of previous work

As discussed in Sec. 1.1.1, the numerical models used to predict the deformation of MBJ 3D printed specimens due to sintering may sometimes be inaccurate. Improving this accuracy was the motivation for previous work [Troelsgaard and Elmstrøm, 2023]. The contents of this report are a continuation of the work on the 3$^{\text{rd}}$ semester by [Troelsgaard and Elmstrøm, 2023]. The previous work entailed inverse modeling to determine material parameters that lead to numerical sintering deformation models matching observed deformation states of MBJ 3D printed parts after sintering.

To model the deformation due to sintering of MBJ 3D printed parts, multiple macroscopic phenomenological sintering models were presented. Further, the limitations of the sintering models were discussed wrt. to experimental effects from the literature. Assumptions such as isotropy and homogeneous RD distribution limit their accuracy when compared to experimental deformation and RD distribution. In the previous work, the sintering material model used by ANSYS [ANSYS, 2023] was chosen such that the implementation into a finite element program could be verified against an ANSYS sintering analysis.

A Python-based open-source, educational and research-based program PyFEM [de Borst et al., 2023] [de Borst et al., 2012] was used as a baseline and modified to incorporate sintering simulation. The PyFEM program supports geometrically non-linear formulations of finite element analysis which is necessary due to large thermally induced creep deformation by the sintering process. A standard Newton-Raphson solver was used to solve the finite element problem. The PyFEM code was written by object-oriented programming. This meant that it was easy to read and modify the code but increased the difficulty of debugging. Further, several performance issues were encountered with Python and PyFEM which severely limited the model size that could be analyzed.

A custom elastoviscoplastic material model was developed based on sintering theory and implemented into the PyFEM program. The material model utilized a stress-consistent tangent operator, thus ensuring quadratic convergence properties of the NR solver. The material model was compared to a sintering simulation in ANSYS and showed decent correlation. Some differences were identified between the responses but could be due to differences in element formulation and implementation.

The inverse modeling optimization problem was formulated as a minimization problem of a sum of least square residuals. The residuals expressed the deformation error between the sintered MBJ 3D printed part and the sintering deformation analysis model. The Levenberg-Marquardt optimizer was used to solve the inverse modeling problem. Finite difference approximated sensitivities were used to supply the optimizer with gradient information. Due to time constraints, analytical sensitivities were not derived. The inverse modeling problem was never applied to deformation data obtained from a printed and sintered specimen so only numerical test cases were considered.

Multiple numerical test cases showed that from a set of 'wrong' material parameters, the optimizer could change these such that, in one case, the optimized material parameters were within 1% of the known values. Multiple numerical tests were performed on a simple cantilever beam geometry with different amounts of displacement information supplied to the optimizer. The conclusion being, that by strategically choosing the points at which deformation is measured, the optimizer can perform well with little deformation information. All but one numerical test was performed with two material parameters as design variables. A single numerical test was performed with four material parameters as design variables however this test did not perform well and likely got stuck in a local optimum.

The experience and knowledge of this previous work are the foundation for the following extension of the predictability of deformations during sintering.

2 Problem statement

In the introduction, it was shown that sintering specimens that are less susceptible to deformation due to sintering are more often within the required tolerances. A method to redesign or reinforce current specimens such that less deformation due to sintering occurs is sought. Thus, the following problem statement will be the focus of the remainder of this report:

> ***How can density-based topology optimization be used to reduce the deformation of MBJ 3D-printed components due to sintering?***

Due to time constraints, certain aspects of sintering will not be considered in this project's scope. These aspects impact the printed specimens' deformation due to sintering, but their effects will not be considered in the following work. The delimitations are listed below:

- Friction due to contact between print and ceramic plate is omitted.
- Gravity is only calculated at the start of the simulation.
- Anisotropic shrinkage is omitted.
- Non-homogeneous RD distribution is omitted.
- Isothermal heating is assumed.

The remainder of the report will be structured as follows.
In Chap. 3, both microscopic and macroscopic sintering theory will be presented along with sintering material parameters. In Chap. 4, geometrically non-linear finite element theory is introduced. In Chap. 5, the non-linear viscoplastic sintering material model will be presented along with verification by comparing it with an analytical 1D solution and a commercial program. In Chap. 6, the concepts of topology optimization are introduced, including filtering, material interpolation, minimum length scale, design-dependent loads, optimizers, and continuation approaches. In Chap. 7, the design sensitivity analysis is performed where the analytical gradients used by the gradient-based optimizer are derived. In Chap. 8, a set of numerical test cases will be presented along with the optimized designs by the developed topology optimization framework. Finally, in Chap. 9 and Chap. 10 discussion and conclusion of the report's contents will be presented along with future work perspectives.

3 Sintering theory

Sintering is a thermal treatment process in which porous materials are heated to near-melting temperatures, causing the particles to fuse, eliminating inter-particle voids (pores) and forming a solid structure. It is driven by reducing the structure's total energy by decreasing the net surface area and, thereby, surface energy through various diffusional mechanisms. For a comprehensive overview of the micromechanical sintering mechanisms, the reader is referred to Rahaman [2015] and German [1996]. However, in the current work, the macro mechanical response of the structure subject to sintering is of interest.

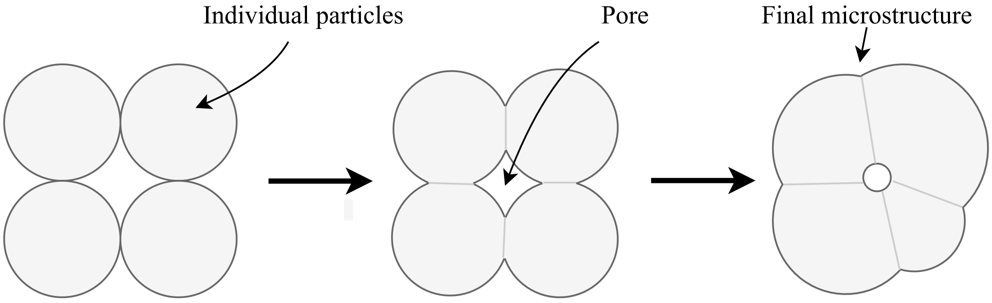


Figure 3.1. Schematic of the evolution of microstructure during sintering.

The continuum theory of sintering, developed in Skorokhod [1972] and Olevsky [1997], synthesizes the microstructural mechanisms into a set of phenomenological continuum equations that describe the structure's macroscopic deformation and shrinkage. Herein, it is assumed that the material consists of a solid phase with evenly distributed pores, cf. Fig. 3.1. The solid phase consists of individual particles with isotropic, viscous and incompressible behavior. Therefore, the overall response is isotropic, and the macromechanical response can be modeled using FEA. This method of modeling the constitutive properties of materials subject to sintering is often called the Skorohod-Olevsky Viscous Sintering ($SOVS$) model in the literature.[Braginsky et al., 2003][Olevsky et al., 2000]

3.1 Constitutive model

Due to its more straightforward implementation and determination of material parameters, a linear viscous version of SOVS is used extensively in the literature to model sintering for both ceramics [Braginsky et al., 2003][Argüello et al., 2009][Shi et al., 2023], metals [Sahli et al., 2015][Al-Qudsi et al., 2015][Zhang, 2005][Song et al., 2006] and specifically for MBJ components [Lee et al., 2022][Borujeni et al., 2022][Zhang et al., 2021]. The linear viscous constitutive equation can in tensor-notation be written as in Eq. (3.1) [Zhang et al., 2021].

$$\dot{\varepsilon}_{ij}^{vp} = \frac{1}{2\eta\phi}\sigma'_{ij} + \frac{1}{18\eta\psi}(\sigma_{kk} - 3P_L)\delta_{ij} \tag{3.1}$$

$\dot{\varepsilon}_{ij}^{vp}$ is the viscoplastic strain rate tensor, σ'_{ij} is the deviatoric stress tensor, σ_{kk} is the trace of the stress tensor, P_L is the effective sintering stress, ϕ and ψ is the effective shear- and bulk viscosity moduli, respectively, given in Eq. (3.2) [Shi et al., 2023].

$$\phi = d^2 \qquad\qquad \psi = \frac{2d^3}{3(1-d)} \tag{3.2}$$

d is the relative density (RD), the amount of solid phase and η is the uniaxial viscosity of the material. Fitting experimental rheological data would generally find the latter [Olevsky et al., 2000]. The viscosity is a thermally activated process and it is therefore reasonable to assume an Arrhenius-type relation as in Eq. (3.3) [Zhang, 2005].

$$\eta = A_\eta \exp\left(\frac{Q_\eta}{RT}\right) \tag{3.3}$$

A_η is the pre-exponential factor, R is the universal gas constant, T is the absolute temperature and Q_η is a exponential factor or activation energy.
RD is governed by mass conservation and can, therefore, be modeled using Eq. (3.4) which expresses the densification rate [Argüello et al., 2009].

$$\dot{d} = -d\dot{\varepsilon}_{kk}^{vp} \tag{3.4}$$

The effective sintering stress is given by Eq. (3.5) [Zhang et al., 2021].

$$P_L = \frac{6\alpha}{G}d^2 \tag{3.5}$$

α is the surface energy of the particles and G is the grain-size. Grain growth for metal alloys can be assumed to be a thermally activated process and can therefore be modeled using an Arrhenius type equation as in Eq. (3.6) [Zhang et al., 2021].

$$\dot{G} = \frac{A_G}{G}\exp\left(-\frac{Q_G}{RT}\right) \tag{3.6}$$

A_G is a pre-exponential factor, and Q_G is an exponential factor or activation energy.
In combination Eq. (3.1)-Eq. (3.6) form the constitutive equations necessary to model the macromechanical strain due to sintering.

3.2 Material parameters

Determination of material parameters is critical to the accuracy of the numerical model, as was explored in Troelsgaard and Elmstrøm [2023].
Experimental investigations show a piecewise linear relationship between the logarithmic grain-growth rate $\dot{G}$ and the inverse of temperature [Zhang et al., 2021]. This therefore

results in a transition between two values for the activation energy, Q_G, at a specified transition temperature, T_G, as in Eq. (3.7).

$$Q_G = \begin{cases} Q_{G1}, & \text{if } T \leq T_G \\ Q_{G2}, & \text{otherwise} \end{cases} \tag{3.7}$$

Similarly, the axial shrinkage due to sintering is only detectable above a certain temperature threshold [Zhang, 2005]. Sinter-stress is therefore only calculated above a specified transition temperature, as in Eq. (3.8).

$$P_L = \begin{cases} \frac{6\alpha}{G} d^2, & \text{if } T \geq T_s \\ 0, & \text{otherwise} \end{cases} \tag{3.8}$$

Material parameters have been supplied by the collaborative partner, given in Tab. 3.1. These have previously been used for numerical simulations of sintering for MBJ-printed components and are therefore adopted in this work. Several of these parameters are standard values, used extensively in literature.

E	ν	ρ	d_0	A_η	Q_η	Q_{G1}	Q_{G2}	A_G	α	G_0	T_G	T_s
196e9	0.28	7995	0.595	77e7	2e4	315.8e3	5e4	7.39e-13	2	4.3e-6	1050	918

Table 3.1. Material parameters for test case. All are given in SI units.

A benchmark sinter schedule will be used, as given in Fig. 3.2.

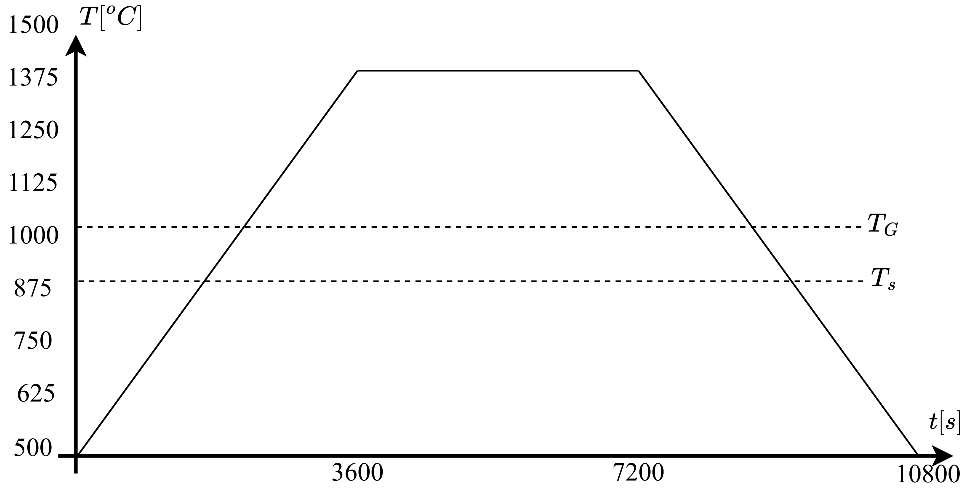


Figure 3.2. Benchmark sinter schedule.

The initial stages of sintering primarily evaporate the binder and form loose connections between particles. The physics regarding this preliminary sintering stage will not be modeled; thus, it is not important to include these low-temperature ranges in the sintering schedule. Therefore, only temperatures above 500 °C is included in Fig. 3.2. This shortens the solution time, which is one of the primary concerns regarding topology optimization.

The constitutive models explored in this section must be implemented in a FEA program to model the deformations of sintering. FEA theory will therefore be presented next.

4 Geometrically non-linear finite element theory

During sintering of MBJ 3D printed specimens, large deformations can occur due to viscoeffects caused by prolonged exposure to elevated temperatures, see Fig. 3.2. Hence, it is necessary to use a Geometrically Non-Linear Finite Element (GNLFE) program to model the deformation. The GNLFE theory reviewed in Sec. 4.1 is implemented in a custom MATLAB program. ANSYS is used as a pre-processor to discretize the structure and apply boundary conditions before the data is exported to the GNLFE program using ANSYSimport by [Pedersen, 2024]. The MATLAB implementation of the GNLFE program is inspired by the open-source research- and education-based FE program PyFEM [de Borst et al., 2023] [de Borst et al., 2012] also used in [Troelsgaard and Elmstrøm, 2023]. The PyFEM program is supplementary to the book by de Borst et al. [2012].

4.1 Kinematic formulation

This theory section is based on a combination of [Crisfield, 1991] and [de Borst et al., 2012]. The following index notation will be used in this section and the remainder of the report unless otherwise specified. The (i, j) superscript denotes the i^{th} load or time step and the j^{th} equilibrium iteration. The subscripts (e, k) denote the e^{th} element and the k^{th} integration point. If any of the super or subscripts are omitted, it means that the super or subscripts are either constant and equal for all symbols or the values are independent of the super or subscripts. The GNLFEA formulation used in this project is based on the total Lagrange formulation. Sintering is an inherent 3D process due to the volumetric shrinkage. Thus, the GNLFE code implemented in MATLAB uses 8-noded trilinear isoparametric 3D elements, as shown in Fig. 4.1. In Fig. 4.2, two different states of the same quadrilateral are shown. State 1 is the reference undeformed state and consists of a square with side lengths $dx_1 = dx_2$. The square of state 1 is subjected to a rotation, a translation and a displacement which is defined as state 2. The deformation between state 1 and state 2 can be described by the deformation gradient $[\mathbf{F}]$, defined as Eq. (4.2). [de Borst et al., 2012]

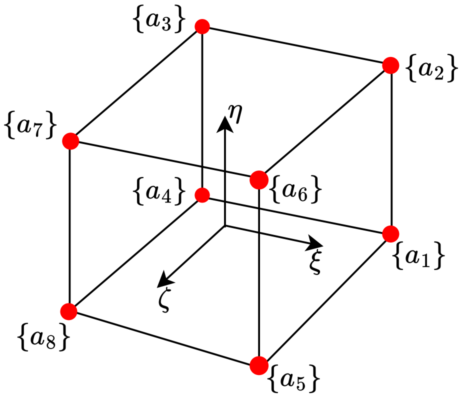


Figure 4.1. Isoparametric 3D element

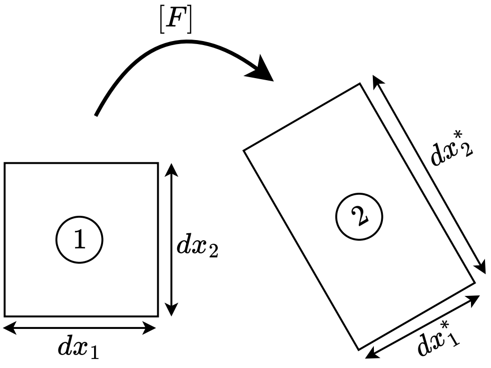


Figure 4.2. Deformation gradient $[\mathbf{F}]$

In GNLFEA applications, an appropriate strain measure must be used that does not produce straining for rigid body motion or rotation. The strain measure often used in GNLFEA is the Green-Lagrange strain formulation denoted by $\{\boldsymbol{\varepsilon}\}^{(i,j)}_{(e,k)}$. An energetically consistent stress measure must also be used when using the nonlinear Green-Lagrange formulation of strains. The 2nd Piola-Kirchhoff stress measure, denoted by $\{\mathbf{S}\}^{(i,j)}_{(e,k)}$, is often used in conjunction with the Green-Lagrange strains [de Borst et al., 2012].
The Green-Lagrange strain can be expressed by the deformation gradient $[\mathbf{F}]^{(i,j)}_{(e,k)}$ as described in Eq. (4.1).

$$\{\boldsymbol{\varepsilon}\}^{(i,j)}_{(e,k)} = \frac{1}{2}\left(\left([\mathbf{F}]^{(i,j)}_{(e,k)}\right)^T [\mathbf{F}]^{(i,j)}_{(e,k)} - [\mathbf{I}]\right) \tag{4.1}$$

In Eq. (4.1) $[\mathbf{I}]$ is the identity matrix. A visualization of the deformation gradient is given in Fig. 4.2. In matrix form, the deformation gradient can be described as Eq. (4.2).

$$[\mathbf{F}]^{(i,j)}_{(e,k)} = \frac{\partial\{\mathbf{x^*}\}}{\partial\{\mathbf{x}\}} \tag{4.2}$$

The deformation from state 1 to state 2 also can be described as $\{\mathbf{u}\} = \{\mathbf{x^*}\} - \{\mathbf{x}\}$ and the deformation gradient can be rewritten as Eq. (4.3) [de Borst et al., 2012].

$$[\mathbf{F}]^{(i,j)}_{(e,k)} = [\mathbf{I}] + \frac{\partial\{\mathbf{u}\}}{\partial\{\mathbf{x}\}} \tag{4.3}$$

$\{\mathbf{u}\}$ denotes the continuous displacements. The Green-Lagrange strains can be decomposed into a linear and a non-linear part as shown in Eq. (4.4). For small deformation, the non-linear strain component can often be neglected.

$$\{\boldsymbol{\varepsilon}\}^{(i,j)}_{(e,k)} = \{\boldsymbol{\varepsilon}_{\mathbf{L}}\}^{(i,j)}_{(e,k)} + \{\boldsymbol{\varepsilon}_{\mathbf{NL}}\}^{(i,j)}_{(e,k)} \tag{4.4}$$

The Green-Lagrange strains can also be described by a single matrix-vector relation, expressing the relation between the strains $\{\boldsymbol{\varepsilon}\}$ and the nodal displacements $\{\mathbf{a}\}$, see Eq. (4.5) [Crisfield, 1991].

$$\begin{aligned}\{\boldsymbol{\varepsilon}\}^{(i,j)}_{(e,k)} &= \{\boldsymbol{\varepsilon}_{\mathbf{L}}\}^{(i,j)}_{(e,k)} + \{\boldsymbol{\varepsilon}_{\mathbf{NL}}\}^{(i,j)}_{(e,k)} = \left([\mathbf{H}] + \frac{1}{2}[\mathbf{A}(\boldsymbol{\theta})]^{(i,j)}_{(e,k)}\right)\{\boldsymbol{\theta}\}^{(i,j)}_{(e,k)} \\ &= \left([\mathbf{H}] + \frac{1}{2}[\mathbf{A}(\boldsymbol{\theta})]^{(i,j)}_{(e,k)}\right)[\mathbf{G}]_{(e,k)}\{\mathbf{a}\}^{(i,j)}_{(e)}\end{aligned} \tag{4.5}$$

From Eq. (4.5) it can be seen that the deformation derivative components in vector form $\{\boldsymbol{\theta}\}_{(e,k)}^{(i,j)}$ also can be described by the derivatives of the shape functions $[\mathbf{G}]_{(e,k)}$ and the nodal displacements $\{\mathbf{a}\}_{(e)}^{(i,j)}$. The matrices in Eq. (4.5) are shown in Eqs. (4.6) to (4.8) for a 2D 4-noded element. The 2D matrices are shown for brevity. The GNLFEA analysis will be performed in 3D. See Crisfield [1991] for the 3D versions of the matrices in Eqs. (4.6) to (4.8). The $[\mathbf{H}]$ matrix is a Boolean bookkeeping matrix. The $[\mathbf{A}(\boldsymbol{\theta})]_{(e,k)}^{(i,j)}$ matrix contains deformation derivative components. The $\{\boldsymbol{\theta}\}_{(e,k)}^{(i,j)}$ vector also contains the deformation derivative components but in a vectorized structure.

$$[\mathbf{H}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \qquad [\mathbf{A}(\boldsymbol{\theta})]_{(e,k)}^{(i,j)} = \begin{bmatrix} \frac{\partial u}{\partial x} & 0 & \frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial u}{\partial y} & 0 & \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} & \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial x} \end{bmatrix} \tag{4.6}$$

$$\{\boldsymbol{\theta}\}_{(e,k)}^{(i,j)} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{Bmatrix} \quad [\mathbf{G}]_{(e,k)} = \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \tag{4.7}$$

The comma notation in Eq. (4.7) denotes differentiation, i.e. $N_{1,x}$ is the shape function describing node 1 in the element differentiated w.r.t. the undeformed global x-direction. The same direction along which a_1, a_3, a_5 and a_7 is given.

$$\{\mathbf{a}\}_{(e)}^{(i,j)} = \begin{Bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 \end{Bmatrix}^T \tag{4.8}$$

The entries in Eq. (4.8) are organized such that a_1, a_3, a_5 and a_7 are all nodal displacement along the global x-direction while a_2, a_4, a_6 and a_8 are nodal displacements along the global y-direction.

4.2 Newton Raphson solution

When performing GNLFEA the most important requirement is to ensure that the external and internal forces are in equilibrium. To quantify the force equilibrium, the out-of-balance force vector (or residual) is often used and can on element level be defined as Eq. (4.9).

$$\{\mathbf{r}\}_{(e)}^{(i,j)} = \sum_{k=1}^{n_{ipts}} \{\mathbf{r}\}_{(e,k)}^{(i,j)} w_{(e,k)} \tag{4.9}$$

The element residual $\{\mathbf{r}\}_{(e,k)}^{(i,j)}$ is integrated over the element volume using Gauss-quadrature by a weighted summation of the integration point contributions $\{\mathbf{r}\}_{(e,k)}^{(i,j)}$, given in Eq. (4.10).

$$\{\mathbf{r}\}_{(e,k)}^{(i,j)} = \{\mathbf{f_{int}}\}_{(e,k)}^{(i,j)} - \{\mathbf{f_{ext}}\}_{(e,k)} \tag{4.10}$$

For GNLFEA where static loads are applied, the external force is often applied during some set of pseudo time steps. In the case of sintering the only external force of interest is due to gravity. The force due to gravity is often small and is therefore held constant throughout the analysis. The internal force on element level can be described by Eq. (4.11).

$$\{\mathbf{f_{int}}\}^{(i,j)}_{(e,k)} = \left([\mathbf{B_{NL}}]^{(i,j)}_{(e,k)}\right)^T \{\mathbf{S}\}^{(i,j)}_{(e,k)} \tag{4.11}$$

In Eq. (4.11) and the remainder of the report $w_{(e,k)}$ will denote integration weights for numerical integration. In the GNLFEA formulation full Gauss integration is used. The $[\mathbf{B_{NL}}]^{(i,j)}_{(e,k)}$ matrix can be derived by variational analysis of the virtual work, cf. Crisfield [1991] for more information. $[\mathbf{B_{NL}}]^{(i,j)}_{(e,k)}$ is defined as Eq. (4.12).

$$[\mathbf{B_{NL}}]^{(i,j)}_{(e,k)} = [\mathbf{H}]\,[\mathbf{G}]_{(e,k)} + [\mathbf{A}(\boldsymbol{\theta})]^{(i,j)}_{(e,k)}\,[\mathbf{G}]_{(e,k)} \tag{4.12}$$

The solution to the GNLFEA problem is to find the displacements that ensure equilibrium between the internal and external forces. By variational analysis of the virtual work, the tangent stiffness matrix can be derived. For GNLFEA the tangent stiffness matrix has two contributions. The first contribution $[\mathbf{k_{ta}}]^{(i,j)}_{(e,k)}$ is purely due to displacement and material stiffness whereas, the second contribution $[\mathbf{k_{tS}}]^{(i,j)}_{(e,k)}$ is the stress stiffening contribution. The tangent stiffness matrix is defined in Eq. (4.13).

$$\begin{aligned}[\mathbf{k_t}]^{(i,j)}_{(e)} &= \sum_{e=1}^{n_{ipts}} \left([\mathbf{k_{ta}}]^{(i,j)}_{(e,k)} + [\mathbf{k_{tS}}]^{(i,j)}_{(e,k)}\right) w_{(e,k)} \\ &= \sum_{e=1}^{n_{ipts}} \left(\left([\mathbf{B_{NL}}]^{(i,j)}_{(e,k)}\right)^T [\mathbf{D}]^{(i,j)}_{(e,k)}\,[\mathbf{B_{NL}}]^{(i,j)}_{(e,k)} + \left([\mathbf{G}]_{(e,k)}\right)^T \left[\hat{\mathbf{S}}\right]^{(i,j)}_{(e,k)} [\mathbf{G}]_{(e,k)}\right) w_{(e,k)}\end{aligned} \tag{4.13}$$

The $\left[\hat{\mathbf{S}}\right]^{(i,j)}_{(e,k)}$ matrix is in 2D a 4 × 4 matrix containing the three unique stress components $\{\mathbf{S}\}^{(i,j)}_{(e,k)} = \{S_{xx}\ S_{yy}\ S_{xy}\}^T$ in a block diagonal manner. In 3D it is expanded to a 9 × 9 block diagonal matrix. $[\mathbf{D}]^{(i,j)}_{(e,k)}$ is the instantaneous material stiffness.

So far, the force residual and tangent stiffness matrix have been defined on element level only but must be assembled into global contributions before solving the system of equations. The $\mathcal{A}$ operator denotes consistent assembly of integration point level and element level values to global level. Thus, assembly of the global force residual vector and global tangent stiffness matrix is shown in Eq. (4.14).

$$\{\mathbf{R}\}^{(i,j)} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \{\mathbf{r}\}^{(i,j)}_{(e)} \qquad [\mathbf{K_t}]^{(i,j)} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} [\mathbf{k_t}]^{(i,j)}_{(e)} \tag{4.14}$$

Finally, using an iterative Newton-Raphson solver the linearized set of equations can be set-up as seen in Eq. (4.15). The advantage of the Newton-Raphson solver is that in each equilibrium iteration j only a linear set of equations has to be solved to find the unknown displacement increment and the NR-solver shows quadratic convergence close to the solution when using a stress-consistent tangent stiffness matrix [Simo and Taylor, 1985].

$$[\mathbf{K_t}]^{(i,j)}\,\{\boldsymbol{\Delta}\mathbf{U}\}^{(i,j)} = -\{\mathbf{R}\}^{(i,j)} \tag{4.15}$$

The full Newton-Raphson solver is used in this report. Thus, the global tangent stiffness matrix is evaluated in each equilibrium iteration. The total displacement can be updated during equilibrium iterations according to Eq. (4.16).

$$\{\mathbf{U}\}^{(i,j+1)} = \{\mathbf{U}\}^{(i,j)} + \{\boldsymbol{\Delta}\mathbf{U}\}^{(i,j)} \tag{4.16}$$

Based on the GNLFE theory presented in this chapter a GNLFEA program has been developed such that the large deformations induced by sintering of MBJ 3D printed specimens can be captured. To model the sintering-induced deformations, a material model based on the constitutive models introduced in Chap. 3 must be implemented in a material model in the GNLFEA MATLAB program.

5 Material model

The constitutive equations, as given in Chap. 3, have to be implemented as the material model in the FEA routine, as outlined in Chap. 4. This chapter will explain the approach taken to implement the material model in a finite-element context. In Sec. 5.1 the material model and integration of the rate equations is introduced. In Sec. 5.2 relaxation of the inherent discontinuities is introduced and in Sec. 5.3 the implementation is verified.

5.1 Material response residual

Departing from a known configuration at time t, the rate constitutive and evolution equations must be numerically integrated to get the material response at time $t + \Delta t$. Commonly, plasticity is calculated using a return-mapping algorithm that necessitates the iterative solution of internal residuals, as done in, e.g., Simo and Taylor [1985]. Iterative solution procedures have likewise been used for sintering simulations by Shi et al. [2023]. However, the current work seeks a direct evaluation to avoid computationally costly iterative solution procedures. This is achieved through a modified method of numerical integration of rate-constitutive equations for sintering, similar to one developed by McHugh and Riedel [1997], which has been used in numerical simulation of sintering [Braginsky et al., 2003] [Argüello et al., 2009].

In this section, the element and integration point subscripts have been omitted for readability. The material model is assumed to be defined in an arbitrary element and integration point. A superscript $[\ ]^{(i)}$ denotes that the value of $[\ ]$ is evaluated for converged variables at the i^{th} loadstep. The equilibrium iteration index (j) is omitted for the remainder of this section, as it is assumed that iteration is performed until convergence of the current time step (i). The material response depends on a set of internal variables: the RD d and grain-size G. The internal variables are also defined by rate equations and, therefore, need to be integrated over the entire time history to evaluate the material state. The internal variables are collected in a vector $\{\boldsymbol{\kappa}\}$ for ease of notation. An internal material residual, $\{\mathbf{H}\}$, can be defined that describes the material response and is dependent on the current stress state $\{\mathbf{S}\}$, the internal variables $\{\boldsymbol{\kappa}\}$, the strain $\{\boldsymbol{\varepsilon}\}$ and external factors inherent to the analysis, such as the time-step Δt and temperature T.

$$\{\mathbf{H}(\{\mathbf{S}\}, \{\boldsymbol{\kappa}\}, \{\boldsymbol{\varepsilon}\}, \mathbf{T}, \boldsymbol{\Delta}\mathbf{t})\}^{(i)} = \{\mathbf{0}\} \tag{5.1}$$

Enforcing this residual to be zero in the current iteration (i), the value of the corresponding internal variables, e.g., the stress $\{\mathbf{S}\}^{(i)}$ and $\{\boldsymbol{\kappa}\}^{(i)}$, can be calculated, for a given strain $\{\boldsymbol{\varepsilon}\}$. Thus, this residual needs to be defined.

Due to the strain decomposition, the total strain increment should equal the sum of the elastic and viscoplastic strain increments. Furthermore, the increment in the internal variables should be consistent with the rate equation.

$$\{\mathbf{H}\}^{(i)} = \begin{Bmatrix} \{\mathbf{h}_{\boldsymbol{\varepsilon}}\}^{(i)} \\ \{\mathbf{h}_{\boldsymbol{\kappa}}\}^{(i)} \end{Bmatrix} = \begin{Bmatrix} \{\boldsymbol{\Delta\varepsilon}^{\mathbf{e}}\}^{(i)} + \Delta t\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}^{(i)} - \{\boldsymbol{\Delta\varepsilon}\}^{(i)} \\ \{\boldsymbol{\Delta\kappa}\}^{(i)} - \Delta t\{\dot{\boldsymbol{\kappa}}\}^{(i)} \end{Bmatrix} = \{\mathbf{0}\} \tag{5.2}$$

The rates at (i) are unknown since the variables themselves are unknown. However, the rates are estimated using a first-order Taylor expansion. The method is, therefore, only semi-implicit.

$$\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}^{(i)} = \{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}^{(i-1)} + \left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}\{\boldsymbol{\Delta}\mathbf{S}\} + \left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}\{\boldsymbol{\Delta\kappa}\} + \left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\right)^{(i-1)}\Delta T \tag{5.3a}$$

$$\{\dot{\boldsymbol{\kappa}}\}^{(i)} = \{\dot{\boldsymbol{\kappa}}\}^{(i-1)} + \left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}\{\boldsymbol{\Delta}\mathbf{S}\} + \left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}\{\boldsymbol{\Delta\kappa}\} + \left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\right)^{(i-1)}\Delta T \tag{5.3b}$$

The increments in 2$^{\text{nd}}$ Piola-Kirchoff stress $\{\boldsymbol{\Delta}\mathbf{S}\}$ and in internal variables $\{\boldsymbol{\Delta\kappa}\}$ is defined as the change in value between subsequent iterations.

$$\{\boldsymbol{\Delta}\mathbf{S}\} = \{\mathbf{S}\}^{(i)} - \{\mathbf{S}\}^{(i-1)} \qquad\qquad \{\boldsymbol{\Delta\kappa}\} = \{\boldsymbol{\kappa}\}^{(i)} - \{\boldsymbol{\kappa}\}^{(i-1)}$$

Substitution of Eq. (5.3) into Eq. (5.2), and by using Hookes law to calculate the elastic strain, the linearised residuals are given in Eq. (5.4).

$$\begin{bmatrix} [\mathbf{D}]^{-1} + \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)} & \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)} \\ -\Delta t\left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)} & [\mathbf{I}] - \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)} \end{bmatrix} \begin{Bmatrix} \{\boldsymbol{\Delta}\mathbf{S}\} \\ \{\boldsymbol{\Delta\kappa}\} \end{Bmatrix} = \begin{Bmatrix} \{\boldsymbol{\Delta\varepsilon}\} - \Delta t\left(\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}^{(i-1)} - \left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\right)^{(i-1)}\Delta T\right) \\ \Delta t\left(\{\dot{\boldsymbol{\kappa}}\}^{(i-1)} + \left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\right)^{(i-1)}\Delta T\right) \end{Bmatrix} \tag{5.4}$$

Thus yielding a system of 8 linear equations to be solved for the increment in stress $\{\boldsymbol{\Delta}\mathbf{S}\}$ and internal variables $\{\boldsymbol{\Delta\kappa}\}$, for a known increment in time Δt and temperature ΔT. This is consistent with the derivations in McHugh and Riedel [1997], Braginsky et al. [2003] and Argüello et al. [2009], if these used a fully implicit trapezoidal integration rule. Solution of Eq. (5.4) yields the increment in stress and internal variables. Due to the matrix operations involved in the solution of this system of equations and the many times it would have to be evaluated, it is an inefficient implementation for the finite element scheme. Furthermore, the numerical inversion of the complete matrix may be numerically ill-conditioned, due to the large numerical difference in the values involved. Therefore, a direct computation is warranted. Eq. (5.4) can be rewritten by temporarily grouping terms, as in Eq. (5.5).

$$\begin{bmatrix} [\mathbf{A_{11}}] & [\mathbf{A_{12}}] \\ [\mathbf{A_{21}}] & [\mathbf{A_{22}}] \end{bmatrix} \begin{Bmatrix} \{\boldsymbol{\Delta}\mathbf{S}\} \\ \{\boldsymbol{\Delta\kappa}\} \end{Bmatrix} = \begin{Bmatrix} \{\mathbf{b_1}\} \\ \{\mathbf{b_2}\} \end{Bmatrix} \tag{5.5}$$

The grouped representation in Eq. (5.5) can then be isolated for the increment in stress and internal variables, as in Eqs. (5.6) to (5.9).

$$\begin{matrix}[\mathbf{A_{11}}]\{\boldsymbol{\Delta}\mathbf{S}\}+[\mathbf{A_{12}}]\{\boldsymbol{\Delta\kappa}\}=\{\mathbf{b_1}\}\\ [\mathbf{A_{21}}]\{\boldsymbol{\Delta}\mathbf{S}\}+[\mathbf{A_{22}}]\{\boldsymbol{\Delta\kappa}\}=\{\mathbf{b_2}\}\end{matrix}\longleftrightarrow \tag{5.6}$$

$$\begin{matrix}[\mathbf{A_{11}}]\{\boldsymbol{\Delta}\mathbf{S}\}+[\mathbf{A_{12}}][\mathbf{A_{22}}]^{-1}\big(\{\mathbf{b_2}\}-[\mathbf{A_{21}}]\{\boldsymbol{\Delta}\mathbf{S}\}\big)=\{\mathbf{b_1}\}\\ [\mathbf{A_{22}}]\{\boldsymbol{\Delta\kappa}\}+[\mathbf{A_{21}}][\mathbf{A_{11}}]^{-1}\big(\{\mathbf{b_1}\}-[\mathbf{A_{12}}]\{\boldsymbol{\Delta\kappa}\}\big)=\{\mathbf{b_2}\}\end{matrix}\longleftrightarrow \tag{5.7}$$

$$\begin{matrix}\big([\mathbf{A_{11}}]-[\mathbf{A_{12}}][\mathbf{A_{22}}]^{-1}[\mathbf{A_{21}}]\big)\{\boldsymbol{\Delta}\mathbf{S}\}=\{\mathbf{b_1}\}-[\mathbf{A_{12}}][\mathbf{A_{22}}]^{-1}\{\mathbf{b_2}\}\\ \big([\mathbf{A_{22}}]-[\mathbf{A_{21}}][\mathbf{A_{11}}]^{-1}[\mathbf{A_{12}}]\big)\{\boldsymbol{\Delta\kappa}\}=\{\mathbf{b_2}\}-[\mathbf{A_{21}}][\mathbf{A_{11}}]^{-1}\{\mathbf{b_1}\}\end{matrix}\longleftrightarrow \tag{5.8}$$

$$\begin{matrix}\{\boldsymbol{\Delta}\mathbf{S}\}=\big([\mathbf{A_{11}}]-[\mathbf{A_{12}}][\mathbf{A_{22}}]^{-1}[\mathbf{A_{21}}]\big)^{-1}\big(\{\mathbf{b_1}\}-[\mathbf{A_{12}}][\mathbf{A_{22}}]^{-1}\{\mathbf{b_2}\}\big)\\ \{\boldsymbol{\Delta\kappa}\}=\big([\mathbf{A_{22}}]-[\mathbf{A_{21}}][\mathbf{A_{11}}]^{-1}[\mathbf{A_{12}}]\big)^{-1}\big(\{\mathbf{b_2}\}-[\mathbf{A_{21}}][\mathbf{A_{11}}]^{-1}\{\mathbf{b_1}\}\big)\end{matrix} \tag{5.9}$$

Therefore, the stress- and internal variable increment can be partitioned into two separate calculations, cf. Eq. (5.9). The stress increment is stated explicitly in Eq. (5.10). Herein, the iteration notation is omitted for readability, but the partial derivatives are evaluated for values at the previously converged time-step $(i-1)$, as defined in Eq. (5.4).

$$\begin{aligned}\{\boldsymbol{\Delta}\mathbf{S}\}=\Big[\mathbf{D^{mod}}\Big]\Bigg(\{\boldsymbol{\Delta\varepsilon}\}-\Delta t\bigg(\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}-\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\Delta T\bigg)\\ -\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg([\mathbf{I}]-\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg)^{-1}\bigg(\Delta t\{\dot{\boldsymbol{\kappa}}\}+\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\Delta T\bigg)\Bigg)\end{aligned} \tag{5.10}$$

$$\Big[\mathbf{D^{mod}}\Big]=\bigg([\mathbf{D}]^{-1}+\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}+\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg([\mathbf{I}]-\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg)^{-1}\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\bigg)^{-1} \tag{5.11}$$

$\Big[\mathbf{D^{mod}}\Big]$ given in Eq. (5.11), is the consistent tangent stiffness [McHugh and Riedel, 1997], which relates the incremental strain to the incremental stress. It can also be seen that this takes the form of general rate-dependent material models as described in de Borst et al. [2012], given in Eq. (5.12).

$$\{\boldsymbol{\Delta}\mathbf{S}\}=\Big[\mathbf{D^{mod}}\Big]\{\boldsymbol{\Delta\varepsilon}\}-\{\mathbf{q}\} \tag{5.12}$$

The pseudo-load vector $\{\mathbf{q}\}$, defining non-mechanical quantities, is given in Eq. (5.13).

$$\begin{aligned}\{\mathbf{q}\}=\Big[\mathbf{D^{mod}}\Big]\Bigg(\Delta t\bigg(\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}-\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\Delta T\bigg)\\ -\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg([\mathbf{I}]-\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg)^{-1}\bigg(\Delta t\{\dot{\boldsymbol{\kappa}}\}+\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\Delta T\bigg)\Bigg)\end{aligned} \tag{5.13}$$

The increment in internal variables can be calculated as in Eq. (5.14).

$$\begin{aligned}\{\boldsymbol{\Delta\kappa}\}=[\mathbf{Q}]\Bigg(\Delta t\{\dot{\boldsymbol{\kappa}}\}+\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\Delta T+\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\bigg([\mathbf{D}]^{-1}+\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\bigg)^{-1}\\ \bigg(\{\boldsymbol{\Delta\varepsilon}\}-\Delta t\bigg(\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}-\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\Delta T\bigg)\bigg)\end{aligned} \tag{5.14}$$

$$[\mathbf{Q}]=\bigg([\mathbf{I}]-\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}+\Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\bigg([\mathbf{D}]^{-1}+\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\bigg)^{-1}\Delta t\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\bigg)^{-1} \tag{5.15}$$

The formulations as derived in Eqs. (5.10) to (5.15) are computed analytically and implemented in the NLFEA MATLAB program. The explicit form of all terms and how they are derived can be seen in Appendix A.

5.2 Smoothing

The Grain-growth rate $\dot{G}$ and sintering stress P_L are discontinuous at their transition temperatures, cf. Eq. (3.7) and Eq. (3.8). Therefore, they are non-differentiable and would result in very inaccurate Taylor approximations, cf. Eq. (5.3), if a transition temperature is encountered during the time-step Δt. Furthermore, continuous functions of the material model are needed, to evaluate sensitivities for the topology optimization, cf. Chap. 7. To alleviate the material parameter discontinuities, a smooth approximation to the step function is used, based on a hyperbolic tangent function, to turn them into continuous functions of the temperature, as in Eq. (5.16).

$$\varphi = \frac{1}{2} + \frac{1}{2}\tanh\big(\omega(T - T^{\star})\big) \tag{5.16}$$

T is the temperature, $T^{\star}$ is the transition temperature and ω is a positive parameter that determines the relaxation. As $\omega \to \infty$, the original step function is recovered. Therefore, Eq. (3.7) and Eq. (3.8) can be approximated as in Eq. (5.17).

$$Q_G = (1-\varphi)Q_{G1} + \varphi Q_{G2} = Q_{G1} + \varphi(Q_{G2} - Q_{G1}) \tag{5.17a}$$

$$P_L = \varphi\frac{6\alpha}{G}d^2 \tag{5.17b}$$

ω needs to be chosen, such that the relaxed step function approximates the original step function while ensuring numerical stability in the transition. For this work, it is heuristicly chosen to use $\omega = 1/\Delta T$, where ΔT is the initial temperature increment. This is based on the qualitative observation that it results in approximately the same number of load steps subject to the intermediate values, independent of the step size, as seen in Fig. 5.1. The motivation is, therefore, that it will result in an analysis that is less sensitive to the choice of step size. Additionally, it offers the appealing interpretation that the units cancel and for $\Delta T \to 0^+$, meaning the time discretization vanishes, the original step function is retrieved.

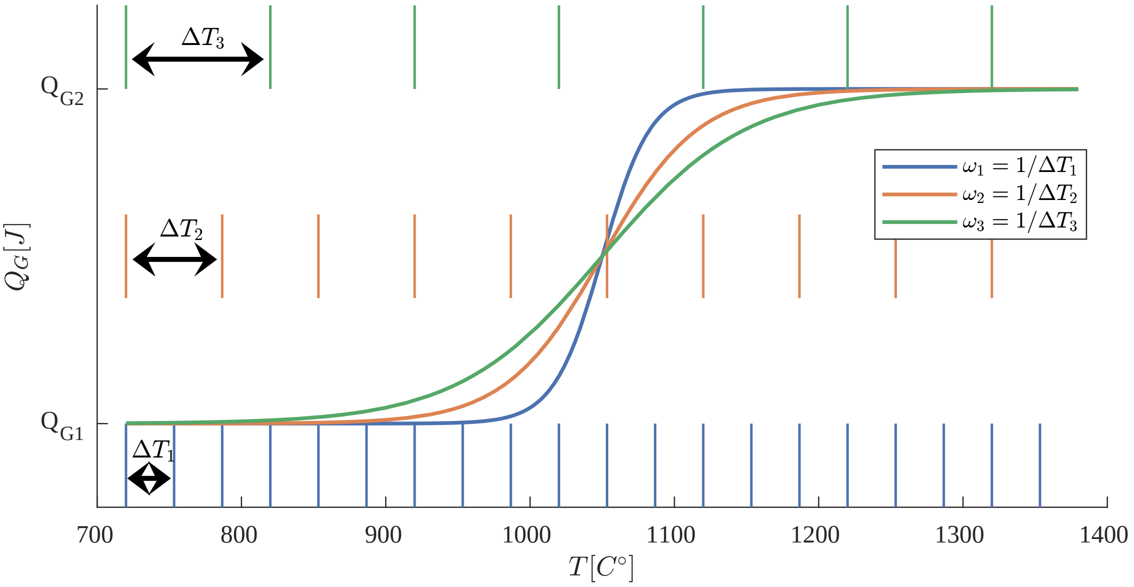


Figure 5.1. Relaxed step-function.

5.3 Verification

To verify the finite-element and material model implementation, two comparisons will be performed; An analytical benchmark, which is adopted in the literature and a commercial benchmark to verify that the response is similar to current software predictions.
Firstly, a short note on convergence is warranted. The same convergence criteria as used in e.g. de Borst et al. [2023] is used in the NLFEA program, whereby convergence of the i^{th} load-step is assumed when the relative residual of the j^{th} equilibrium iteration is below a defined tolerance times a reference value, given by:

$$\left\| \{\mathbf{R}\}^{(i,j)} \right\|_2 \leq R_{\text{tol}} \cdot \|\{\mathbf{F_{ext}}\}\|_2 \tag{5.18}$$

The reference is calculated as the norm of the applied load, and R_{tol} is a predefined convergence tolerance. The norm is the regular 2-norm, and R_{tol} may be different depending on the application. For verification purposes, a strict tolerance of 1e-6 may be used, however, for optimization purposes and general simulation, a more loose tolerance of 1e-3 may be used. The latter is the default value used in de Borst et al. [2023]. For free-sintering, e.g. no gravity is applied, a reference load of 1e-12 is used.

5.3.1 Analytical comparison

The MATLAB implementation is compared to an analytical solution, as done in Braginsky et al. [2003], Argüello et al. [2009] and Shi et al. [2023], based on the results in Olevsky et al. [2000]. The benchmark is two connected elements, subject to symmetry boundary conditions and a constant pressure of 5 MPa in the free end, as in Fig. 5.2.

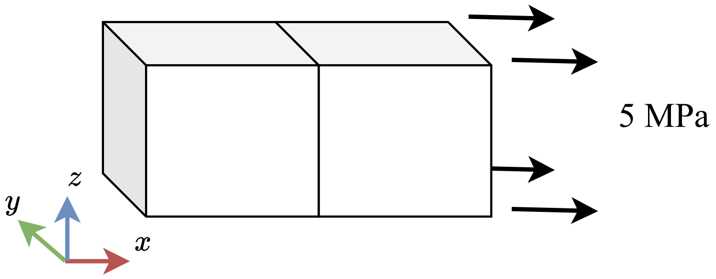


Figure 5.2. Verification benchmark geometry.

The material properties, Tab. 5.1, and sinter schedule will be as given in the former papers. The variables used are directly from these, and are only used for this analytical verification. A constant ramp of 5 °C/min is used from an initial ambient temperature of $T_0 = 750\,°\text{C}$ to the final temperature of $T_1 = 1000\,°\text{C}$. The viscosity is calculated according to a polynomial function.

$$\eta = a\left(\frac{T}{T_0}\right)^2 + b\frac{T}{T_0} + c \tag{5.19}$$

No grain size correction is used and sintering remains active, therefore, the smoothing introduced in Sec. 5.2 is not used and sintering stress is calculated for a constant grain-size G_0. Furthermore, gravity is not applied.

E	ν	d_0	a	b	c	G_0	α	P
123.7e9	0.356	0.47	51.7e10	$-$106.6e10	56.4e10	0.2e-6	1.27/2	5e6

Table 5.1. Material parameters for test case. All are given in SI units. Material values from Olevsky et al. [2000].

For the free sintering case, a closed-form analytical solution exists as given in Eq. (5.20).

$$
\begin{aligned}
d(t) &= 1 - \frac{1-d_0}{F} \exp\left(-\frac{3}{4} \frac{6\alpha d_0^2}{G_0} \frac{2}{\sqrt{Q}} \arctan\left(\frac{2Ct+B}{\sqrt{Q}} \right) \right) \\
F &= \exp\left(-\frac{3}{4} \frac{6\alpha d_0^2}{G_0} \frac{2}{\sqrt{Q}} \arctan\left(\frac{B}{\sqrt{Q}} \right) \right) \\
A = a+b+c\,, \quad B &= a\frac{10}{60T_0} + b\frac{5}{60T_0}\,, \quad C = a\frac{25}{3600T_0^2}\,, \quad Q = 4AC - B
\end{aligned}
\tag{5.20}
$$

For the sintering with an applied load on the free end, the densification rate has the form given in Eq. (5.21).

$$
\dot{d} = \frac{P}{4\eta} \frac{1-d}{d^2} - \frac{18\alpha d_0^2 (1-d)}{4\eta} \tag{5.21}
$$

This has been integrated using a fourth-order Runge-Kutta integration. The results from free sintering and pressured sintering are given in Figs. 5.3 to 5.6.

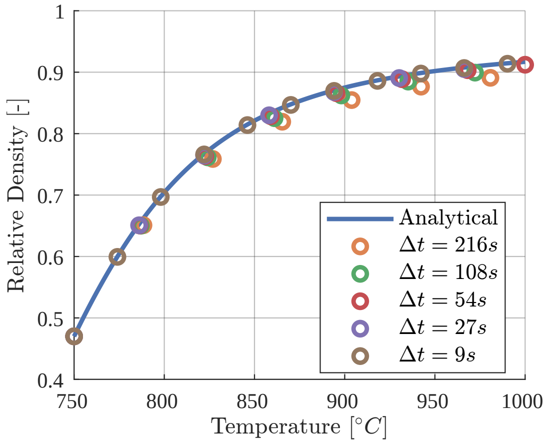


Figure 5.3. Free sintering evolution of RD during analysis.

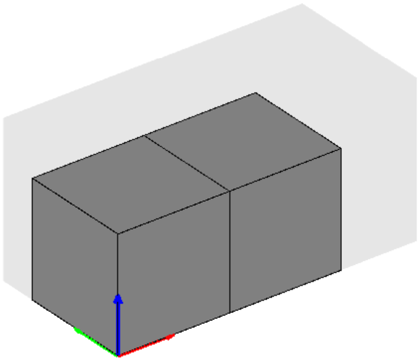


Figure 5.4. Free sintering geometry at T_1. Grey shaded area indicates initial dimensions.

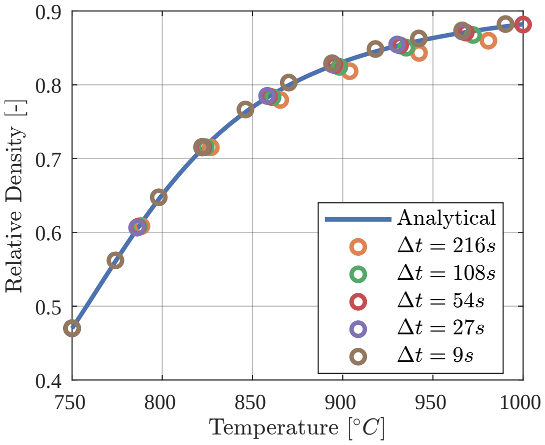


Figure 5.5. Loaded sintering evolution of RD during analysis.

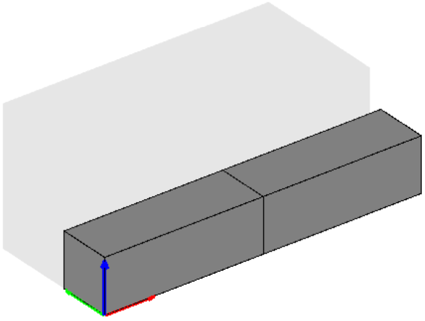


Figure 5.6. Loaded sintering geometry at T_1. Grey shaded area indicates initial dimensions.

As seen, the implementation accurately predicts the RD evolution during the analysis. Due to the semi-implicit material model, there are errors for large time-steps. However, in e.g. Shi et al. [2023], a step-size of 6 s is used as a baseline and the results using a step-size of 30 s likewise showed a small error. The difference between the analytical and numerical results is negligible for a time-step of at least 54 s. Therefore, the results are deemed satisfactory.

5.3.2 Numerical investigation

A preliminary numerical investigation on the influence of step size is warranted, as well as the expected stability of the analysis. This is the purpose of this section.

The NLFEA MATLAB implementation uses grain size correction of the sinter stress and smoothing of an internal variable step function. Thus material parameters and sinter schedule are as given in Sec. 3.2. The benchmark geometry used is the Messerschmitt-Bölkow-Blohm (MBB) beam, as seen in Fig. 5.7, which will be the standard benchmark geometry for the remainder of the work. The full geometry is a hexahedron of dimensions 360 × 60 × 60 mm^3. However, due to symmetry, only a quarter is simulated. The z-component of the displacement is constrained at the free edge, as indicated in the figure.

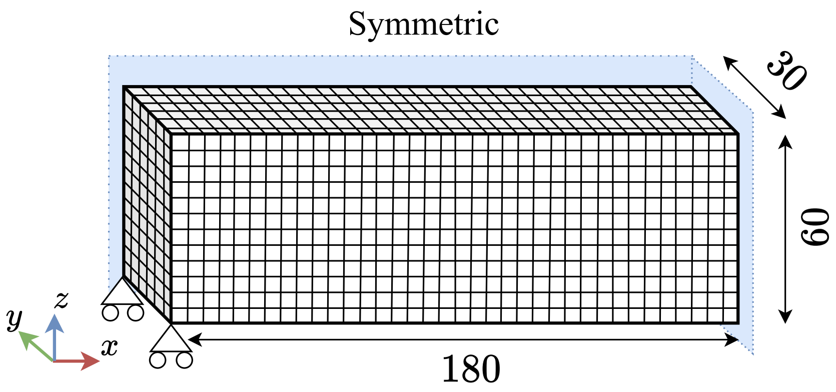


Figure 5.7. Benchmark MBB geometry.

It is adopted as a benchmark, due to the relative dimensions of the beam. Since it is 6 times longer than its cross-sectional dimensions, it is dominated by bending and large deflections are expected due to viscoplastic effects. It should be noted that the trilinear elements would experience locking and, therefore, would predict a lower-than-expected deformation.

To compare the implementations, the deflection and the transversal shrinkage of the top of the beam at the symmetry plane are measured. The MATLAB implementation uses a constant step-size. Due to the semi-implicit material model and the step-size dependent smoothing, a convergence study of the deformation is conducted. This can be seen in Figs. 5.8 to 5.9.

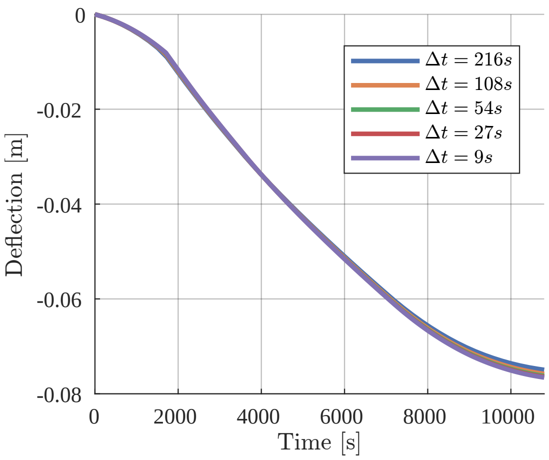


Figure 5.8. Comparison of deflection for different time-steps.

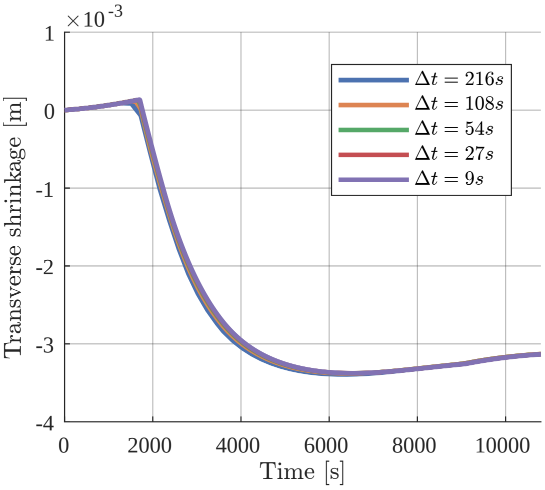


Figure 5.9. Comparison of transverse deformation for different time-steps.

As can be seen from Fig. 5.8 and Fig. 5.9, the response is almost independent of the size of the time steps. Small differences can be seen, which may be attributed to the smoothing introduced in Sec. 5.2. While not shown, similar negligible differences can be seen in RD, as in, e.g., Fig. 5.3 and Fig. 5.5. This means that large time steps may be used for topology optimization, without a great loss of accuracy, for computational efficiency. However, larger time steps may require more equilibrium iterations and may be more prone to divergence. If the correct tangent stiffness is used, quadratic convergence of the residual is expected [Simo and Taylor, 1985][Duretz et al., 2018]. The force residual during the analysis is plotted in Figs. 5.10 to 5.11, for a timestep of $\Delta t = 9\,\text{s}$ and $\Delta t = 216\,\text{s}$, respectively. Each marker represents the residual at an equilibrium iteration. Markers that are connected by lines are within the same load step.

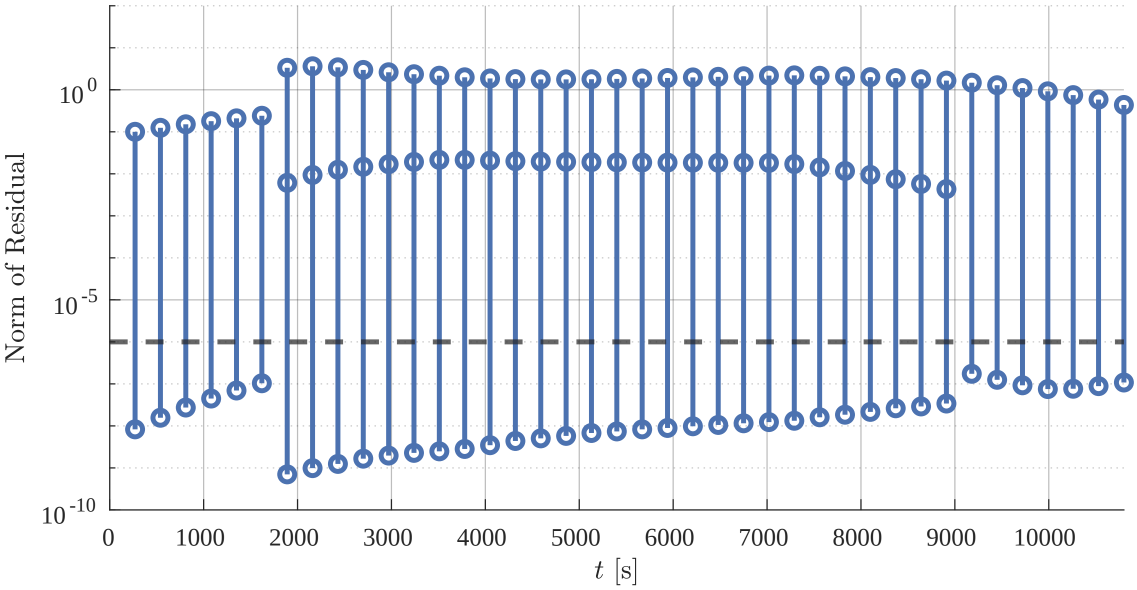


Figure 5.10. Residual during analysis with $\Delta t = 9\,\mathrm{s}$. Only every 30th load step is plotted. Dashed line shows convergence tolerance.

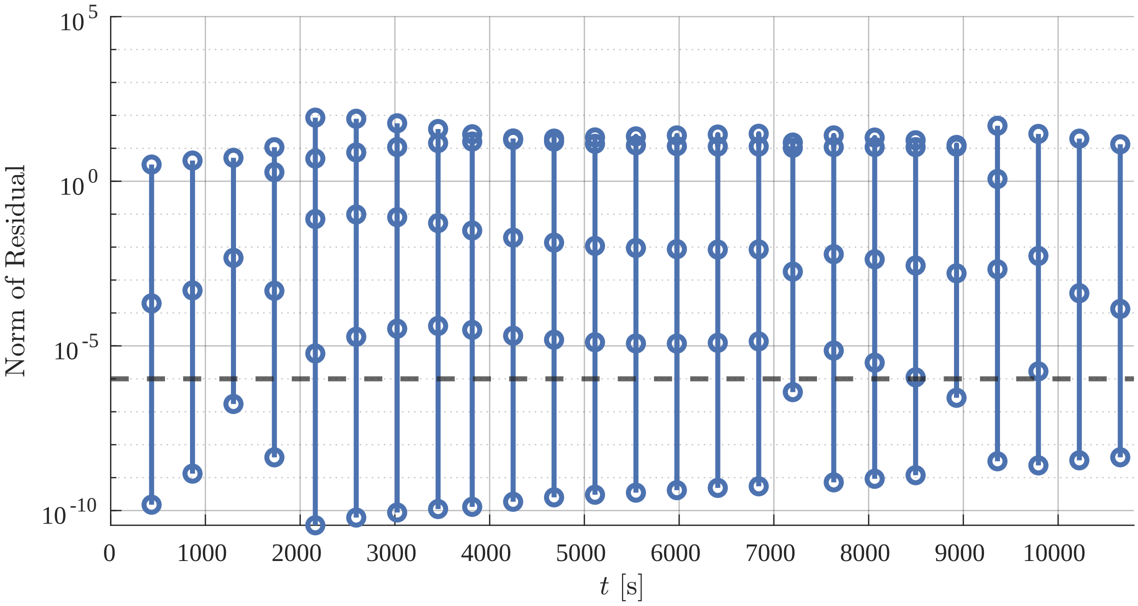


Figure 5.11. Residual during analysis with $\Delta t = 216\,\mathrm{s}$. Only every 2th load step is plotted. Dashed line shows convergence tolerance.

As seen, there is a quadratic convergence of the residual. However, as expected, more equilibrium iterations were needed within a load step, when a larger timestep was used. Also, the convergence appears more irregular. Furthermore, it was observed that larger time steps resulted in a more ill-conditioned stiffness matrix, which may attribute to slower convergence close to the convergence tolerance. The condition number of the stiffness matrix during the sintering analysis is seen in Fig. 5.12 for different time steps.

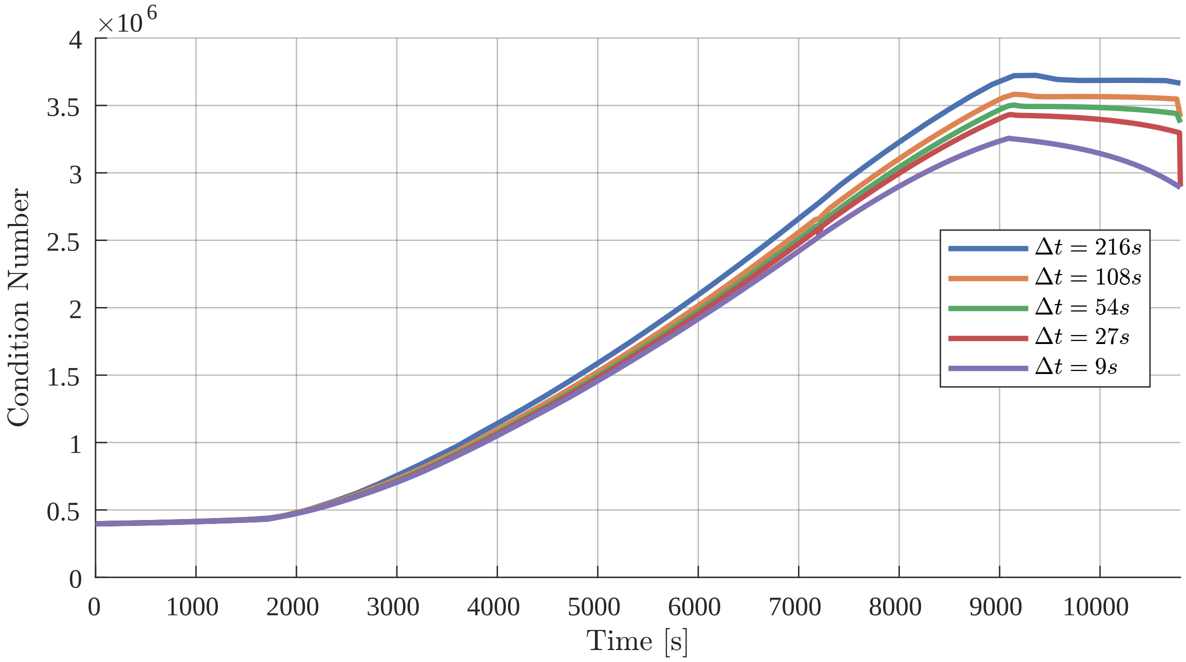


Figure 5.12. Condition number of stiffness matrix during analysis.

The condition number of the stiffness matrix increases roughly one order of magnitude during the analysis, which in general would result in less accurate convergence. However, the difference in condition number between $\Delta t = 216\,\mathrm{s}$ and $\Delta t = 9\,\mathrm{s}$ is relatively insignificant. Thus, the more irregular convergence more likely stems from the relatively inaccurate starting guess for the Newton-Raphson iterations, when large time-steps are used. $\Delta t = 216\,\mathrm{s}$ will be used for topology optimization. Fewer timesteps results in a shorter design sensitivity analysis due to having to solve fewer adjoint system solutions. This is discussed further in Sec. 7.1.

5.3.3 ANSYS comparison

To further validate the deformation prediction of the model, it is benchmarked against a widely used commercial FEA software. Thus, the MATLAB implementation is compared to ANSYS, using a constant step-size of $\Delta t = 9\,\mathrm{s}$, equal material parameters and equal discretization in both programs. The results can be seen in Figs. 5.13 to 5.16.

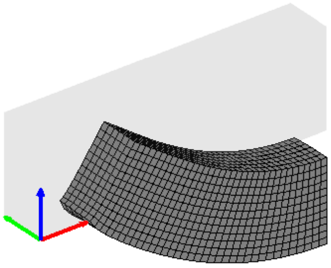


Figure 5.13. Final sintered geometry of MBB beam in MATLAB. Grey shaded area indicates initial dimensions.

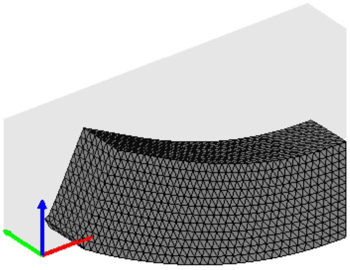


Figure 5.14. Final sintered geometry of MBB beam in ANSYS. Grey shaded area indicates initial dimensions. Note that it is an STL-file export. Thus, it is not a triangular mesh in the analysis.

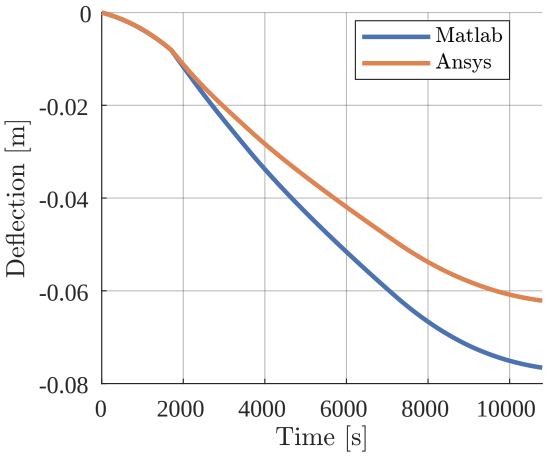


Figure 5.15. Comparison of bending deformation.

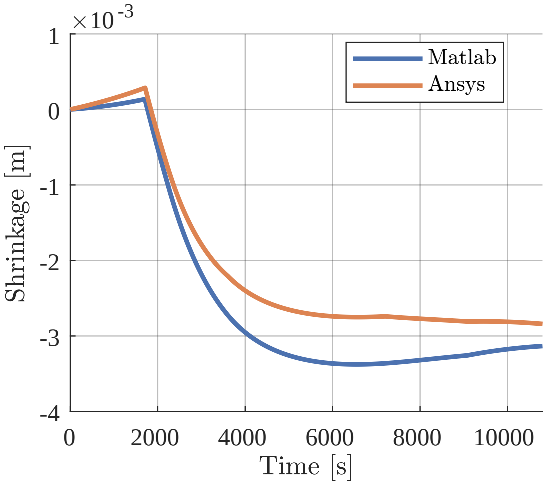


Figure 5.16. Comparison of transverse deformation.

The deformation is initialized to 0 for comparison since ANSYS calculates an initial deformation, likely while applying the gravity over a pseudo-timestep of 1 s. This time step has been excluded from these comparisons. As seen in Figs. 5.15 to 5.16, the deflection is larger in the MATLAB implementation, as compared to ANSYS. This is unexpected since ANSYS uses fully integrated elements with $\bar{B}$-method, alleviating locking. Thus, it would generally be less stiff in bending than the trilinear isoparametric elements used in this work. However, as can be seen in Figs. 5.13 to 5.14, the proposed implementation shows a visually larger deflection than ANSYS. Similarly, in Troelsgaard and Elmstrøm [2023], larger deformations were found as compared to ANSYS. It may be due to difference in implementation between ANSYS and the NLFEA MATLAB program. To evaluate the possible difference between the numerical integration of the internal variables, they are compared in Figs. 5.17 to 5.20, as well as the effective sintering stress and uniaxial viscosity.

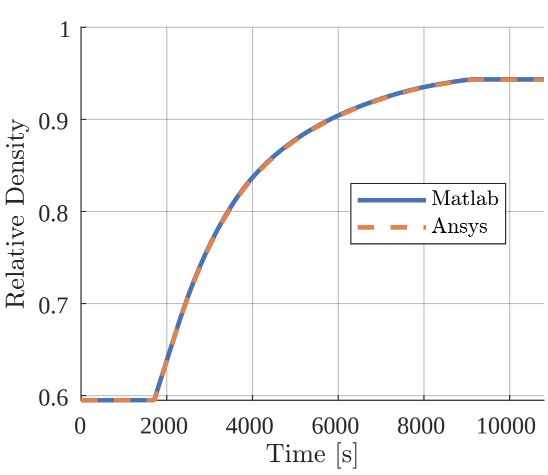


Figure 5.17. Evolution of RD during analysis.

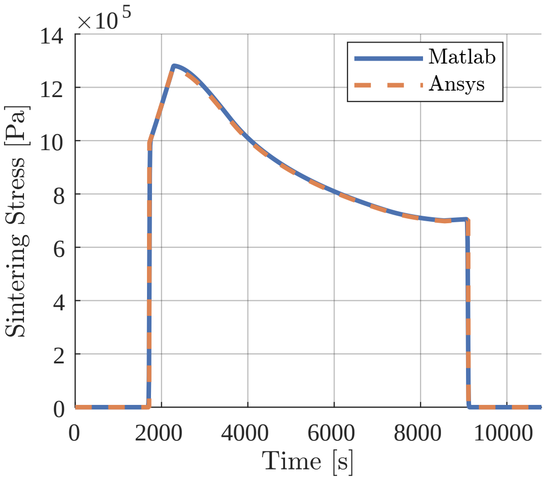


Figure 5.18. Effective sintering stress during analysis.

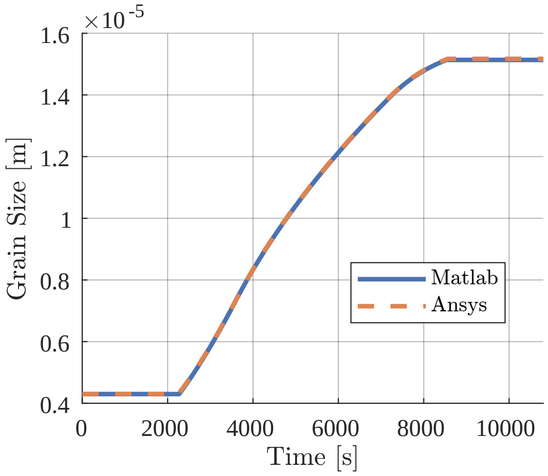


Figure 5.19. Evolution of grain-size during analysis.

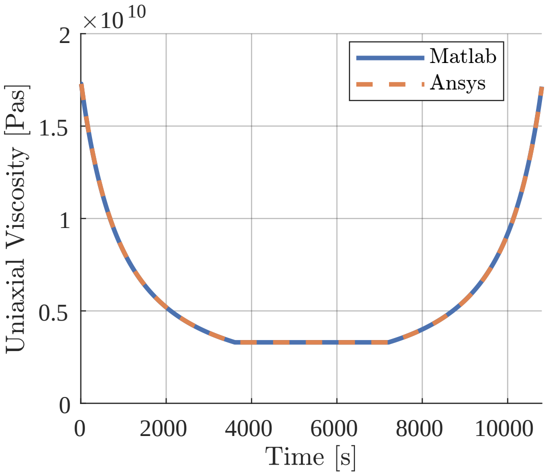


Figure 5.20. Uniaxial viscosity during analysis.

As seen from Figs. 5.17 to 5.20, the difference between the evolution of the sintering variables is negligible between the proposed implementation and ANSYS. Thus, while the exact reason is currently unknown, the differences may likely stem from the element formulation, rather than the material model. It is, however, outside the scope of this project to implement an enhanced element formulation. It is discussed further in Chap. 9.

Since the material model and GNLFEA have now been compared against a 1D analytical solution and ANSYS, and showed reasonable agreement, the scope can therefore be extended to include topology optimization in the following chapter.

6 Topology optimization

Topology Optimization (TO) is the algorithmic redistribution of material within a design space to maximize predefined structural performance objectives and satisfy constraints [Sigmund and Maute, 2013]. TO was initially proposed in Bendsøe and Kikuchi [1988], where material distribution was optimized to enhance structural performance by considering material microstructure. This was later reformulated as a density-based approach in Bendsøe [1989]. Nowadays, much of topology optimization is considered in this way in various forms. The design domain is discretized into a finite element mesh, which means the formulation conveniently reuses the finite element formulation used in the structural analysis. The fundamental concept then involves interpolating the material properties of an element using a continuous artificial density design variable x, which, in this work, is referred to as the Pseudo-Density (PD). Since binary density distribution is warranted, intermediate PD values are penalized, such that their contribution to the performance of the structure is relatively small, compared to the volume occupied. [Bendsøe, 1989][Sigmund and Petersson, 1998][Sigmund and Maute, 2013]

In Secs. 6.1 to 6.5, the theory of TO, as well as general problems entailing compliance optimization of TO, will be explored. In Sec. 6.6 the problem of design-dependent loading is explained. In Sec. 6.7 the continuation strategy is explained and exemplified on a simple example. In Secs. 6.8 to 6.9 the choice of penalization and objective formulation is explained. Lastly, a summary of these sections is given in Sec. 6.10.

6.1 Optimization formulation

The TO problem can be defined as in Eq. (6.1). The objective is to minimize a scalar performance function F with respect to the design variables $\{\mathbf{x}\}$, which for density-based TO are the PDs. F has to be minimized while complying with the underlying FEA analysis, as described in Chap. 4. This is the equality constraints, Eq. (6.1b) and Eq. (6.1c). Here $\{\mathbf{R}\}^{(i)}$ is the global residual defined at nodes, whereas $\{\mathbf{H}\}^{(i)}$ is the local constraints defined at integration points, at every time-step i. This is implicitly enforced, as the FEA is performed in an inner loop, which is the so-called Nested Analysis And Design approach to optimization. Further, it has to comply with a set of constraints $\{\mathbf{g}\}$ on the design variations, such as constraints on the maximum permissible occupied volume or stress.

Furthermore, Eq. (6.1e) is a constraint on the feasible range of the PD.

$$\underset{\{\mathbf{x}\}}{\text{Minimize:}} \quad F(\{\mathbf{x}\}) \tag{6.1a}$$

$$\text{Subject to} \quad \{\mathbf{R}\}^{(i)} = \{\mathbf{0}\} \tag{6.1b}$$

$$\{\mathbf{H}\}^{(i)} = \{\mathbf{0}\} \tag{6.1c}$$

$$\{\mathbf{g}\}^{(i)} \leq \{\mathbf{0}\} \tag{6.1d}$$

$$\{\mathbf{0}\} \leq \{\mathbf{x}\} \leq \{\mathbf{1}\} \tag{6.1e}$$

Although Eq. (6.1a) can represent any arbitrary performance measure, one of the simplest and most commonly utilized measures is compliance for linear elasticity. Therefore, much of the theory presented in the following sections will be related, either directly or indirectly, to linear compliance optimization or methods related to such. A further discussion on the choice of performance measure for the current work is given later in Sec. 6.9. For the geometrical notation used to describe TO throughout the chapter, refer to Fig. 6.1.

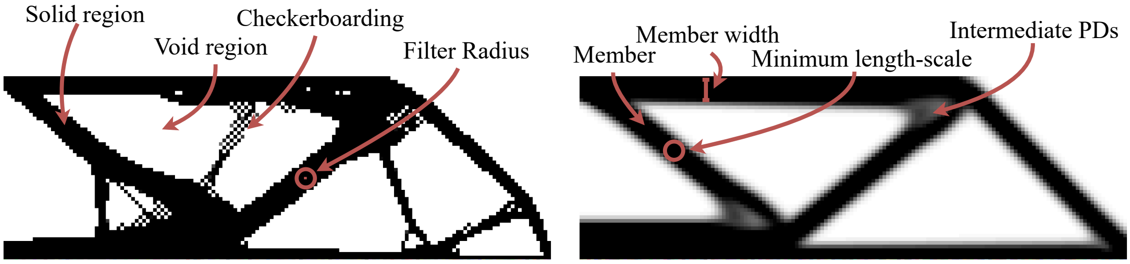


Figure 6.1. Illustrative and explanatory representation of TO. Black elements are solid, white elements are void and gray elements are intermediate

6.2 Material interpolation

As previously described, in TO a binary distribution of the PD, $\{\mathbf{x}\}$, is sought. Using clever interpolation of material parameters, the optimization problem can be guided toward a binary distribution by making it undesirable for the optimizer to include intermediate PD values. During sintering of MBJ 3D printed parts, the only force that acts on the printed parts is due to gravity. The topology-optimized design thus determines where the force due to gravity acts. When including design-dependent body forces, which will be further discussed in Sec. 6.6, the choice of interpolation function can affect the stability of the FEA analysis. Thus multiple interpolation approaches are examined, including Solid Isotropic Material with Penalization (SIMP) [Bendsøe, 1989], modified SIMP [Sigmund, 2007], Rational Approximation of Material Parameters (RAMP) [Stolpe and Svanberg, 2001a], SINH [Bruns, 2005] and Non-Linear Penalization or Interpolation (NLPI) [Pedersen and Pedersen, 2012]. Interpolation of PD using linear, SIMP, RAMP, SINH and NLPI methods are shown in Fig. 6.2. Modified SIMP has been omitted since it shows an almost identical response to the original SIMP.

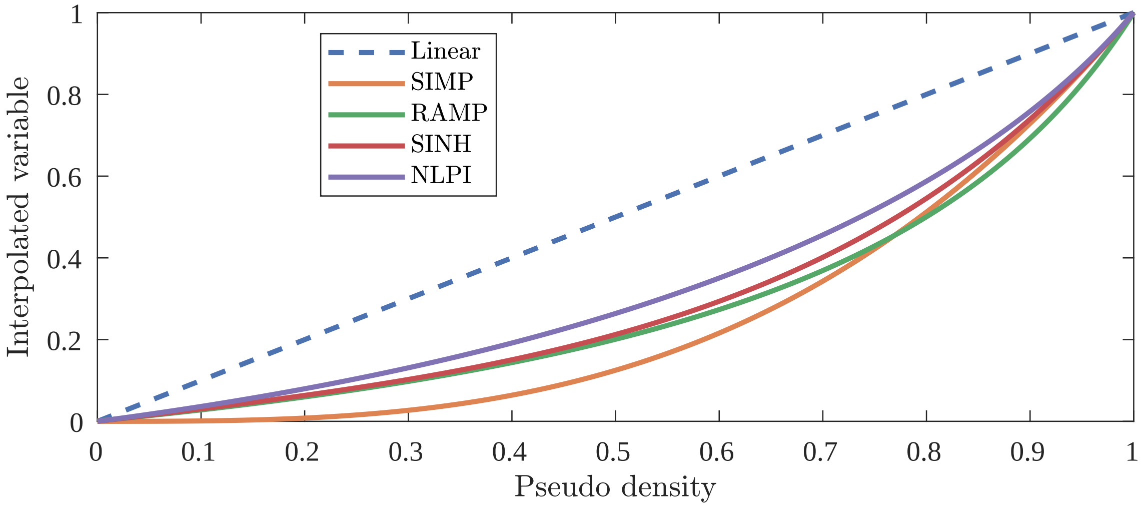


Figure 6.2. The response of linear, SIMP, RAMP, SINH and NLPI interpolation is shown. Typically, the y-axis variable is the stiffness.

SIMP and modified SIMP

The SIMP method is the original idea behind PD-based topology optimization as it is known today. The original SIMP material penalization can be seen in Eq. (6.2) with $E_{(e)}$ the interpolated element material stiffness, $x_{(e)}$ the element PD, p the penalization exponent and E_0 the bulk material stiffness [Bendsøe, 1989].

$$E_{(e)} = (x_{(e)})^p E_0 \tag{6.2}$$

In the original SIMP formulation, the element material density is interpolated linearly see Eq. (6.3) with $\rho_{(e)}$ the interpolated element density and ρ_0 the bulk material density [Bendsøe, 1989].

$$\rho_{(e)} = x_{(e)}\rho_0 \tag{6.3}$$

From Eq. (6.2) it can be seen that for no elemental PD, $x_{(e)} = 0$, the stiffness contribution is also zero which leads to singularities in the stiffness matrix when performing FEA with fixed loads. A method to resolve the singularities is to define a minimum PD or minimum stiffness for each element. This idea can also be incorporated in the SIMP method by use of Modified SIMP, see Eq. (6.4) [Sigmund, 2007].

$$E_{(e)} = E_{min} + (x_{(e)})^p(E_0 - E_{min}) \Leftrightarrow E_{(e)} = (x_{min} + (x_{(e)})^p(1 - x_{min}))E_0 \tag{6.4}$$

E_{min} is the minimum specified stiffness and x_{min} is the corresponding specified minimum PD. By including a minimum stiffness or minimum PD, the void phase, i.e. elements with zero PD ($x_{(e)} = 0$), is attributed with a non-physical stiffness. Both SIMP approaches have zero gradients when $x_{(e)} = 0$. Thus, once an element reaches $x_{(e)} = 0$, the PD in that element will remain zero for the remainder of the optimization. In Fig. 6.2 the SIMP function is plotted with $p = 3$.

RAMP and SINH

Both the RAMP and SINH interpolation function solves the problem of zero gradients when $x_{(e)} = 0$. The RAMP interpolation function is defined as Eq. (6.5a) [Stolpe and Svanberg, 2001a] with the derivative w.r.t. the elemental PD given in Eq. (6.5b) along with its value evaluated at $x_{(e)} = 0$. The RAMP approach also alleviates problems with singular entries in the stiffness matrix by including a minimum PD.

$$E_{(e)} = \Big(x_{min} + \frac{x_{(e)}}{1+p(1-x_{(e)})}(1-x_{min})\Big)E_0 \tag{6.5a}$$

$$\frac{\partial E_{(e)}}{\partial x_{(e)}}\bigg|_{x_{(e)}=0} = E_0\bigg(\frac{(1-x_{min})}{1+p(1-x_{(e)})} + \frac{x_{(e)}(1-x_{min})p}{(1+p(1-x_{(e)}))^2}\bigg)\bigg|_{x_{(e)}=0} = \frac{E_0 - E_0 x_{min}}{1+p} \tag{6.5b}$$

The RAMP interpolation function is plotted with $p = 3$ in Fig. 6.2.

The SINH interpolation function is simpler when taking the derivative w.r.t. the elemental PD, compared to RAMP. The SINH approach proposed by Bruns [2005] does not originally include a minimum PD value, but it can easily be included. The SINH interpolation function can be seen in 6.6a [Bruns, 2005] along with the derivative and the derivative evaluated at $x_{(e)} = 0$ in Eq. (6.6b)

$$E^{(i)} = \bigg(\frac{\sinh(px_{(e)})}{\sinh(p)}\bigg)E_0 \tag{6.6a}$$

$$\frac{\partial E_{(e)}}{\partial x_{(e)}}\bigg|_{x_{(e)}=0} = \bigg(\frac{p\cosh(px_{(e)})}{\sinh(\mathrm{p})}\bigg)E_0\bigg|_{x_{(e)}=0} = \frac{E_0 p}{\sinh(p)} \tag{6.6b}$$

In Fig. 6.2, the SINH interpolation function is plotted with $p = 3$.

NLPI

The NLPI function differs from the previously discussed interpolation functions, as the gradients can be specified at $x_{(e)} = 0$ and $x_{(e)} = 1$ through the two variables κ_0 and κ_1 respectively. Pedersen and Pedersen [2012] summarizes three constraints for choosing κ_0 and κ_1 as seen in Eq. (6.7)

$$0 \le \kappa_0 < 1 \qquad \kappa_1 > 1 \qquad \kappa_0 + \kappa_1 > 2 \tag{6.7}$$

With values chosen for κ_0 and κ_1, the four following parameters, Eq. (6.8), which are used to construct the interpolation function can be determined.

$$r_1 = \frac{\kappa_1 - 1}{\kappa_0 + \kappa_1 - 2} \qquad r_2 = r_1^2 \tag{6.8a}$$

$$r_3 = \frac{\kappa_0 - \kappa_1}{\kappa_0 + \kappa_1 - 2} \qquad r_4 = \frac{2\kappa_0(1-\kappa_1)}{\kappa_0 - \kappa_1} \tag{6.8b}$$

Finally, t can be determined from Eq. (6.9) and the interpolation function $f(x_e)$ can be determined from Eq. (6.10) [Pedersen and Pedersen, 2012].

$$t = r_1 - \sqrt{r_2 + (r_3 x_e)} \tag{6.9}$$

$$f(x_e) = r_4(1-t)t + t^2 \tag{6.10}$$

In Fig. 6.2, the NLPI interpolation has been plotted with $\kappa_0 = \frac{1}{3}$ and $\kappa_1 = 3$. Thus all the interpolation functions in Fig. 6.2 use 3 for the specified values to compare their behavior.

6.3 Psuedo-Density filter

The optimization problem must be well-defined to converge to practical applicable solutions. However, mesh-refinement may result in qualitative different structures with finer details and smaller members, a phenomenon known as mesh dependence. Additionally, checkerboarding is a common problem with compliance optimization using linear element formulations, where alternating solid and void linear elements display artificially high stiffness, cf. Fig. 6.1. Filtering techniques can alleviate both of these challenges. [Sigmund and Petersson, 1998]

Filtering, a technique inspired by image processing, involves smoothing element variables based on neighboring element values. Referencing Fig. 6.1, a mesh-independent filter radius can be introduced, hereby, both mesh-dependence and checkerboarding can be alleviated. Furthermore, a minimum length scale can be ensured and used as an implicit manufacturing constraint dependent on the type of filter used. [Sigmund, 2007]

Various types of smoothing can be used, e.g. Gaussian [Bruns, 2005] and linear [Sigmund, 2007]. Filters can be applied to various parameters, including densities [Bruns and Tortorelli, 2001], composite material fiber orientations [Jiang et al., 2019][Schmidt et al., 2020][Troelsgaard et al., 2023], and sensitivities [Sigmund, 1994]. The choice of filter depends on the specific application. The sensitivity filter augments the sensitivities, thus making them inconsistent with the PD. This can cause convergence issues when line-search-based optimization methods are used, as the sensitivities may not reflect the actual descent direction [Sigmund, 2007]. As a result, this study only considers density-based filtering techniques.

The density filter was introduced by Bourdin [2001], Bruns and Tortorelli [2001]. It alters the PD of an element by calculating it as a weighted average of the neighboring elements $N_{(e)}$ PDs within the filter radius r of the e^{th} element, including the extended domain (cf. Sec. 6.5) if applied, as shown in Eq. (6.11).

$$\tilde{x}_{(e)} = \frac{\sum_{l \in N_{(e)}} G^{(e,l)} x_{(l)}}{\sum_{l \in N_{(e)}} G^{(e,l)}} \tag{6.11}$$

$\tilde{x}_{(e)}$ is the density filtered PD variable of the e^{th} element. It is important to note that the set of neighboring elements $N_{(e)}$ includes the e^{th} element itself. $G^{(i,j)}$ is the weighting function. For a linear density filter, it can be expressed as in Eq. (6.12).

$$G^{(e,l)} = V^{(l)} \max\Big(0, r - \text{dist}(e,l)\Big) \tag{6.12}$$

$V^{(l)}$ is the volume of the l^{th} element and $\text{dist}(e,l)$ is the scalar distance between the centroids of the two elements. In general, the minimum length scale of the optimized design can be expected to be equal to the filter diameter [Zhou et al., 2015], see Fig. 6.1. The result of filtering different member widths can be seen in Fig. 6.3. As seen, for a completely discrete PD distribution, the filtered distribution will contain intermediate PD in the edges. If the member width is smaller than the filter diameter, it results in areas consisting purely of intermediate PD.

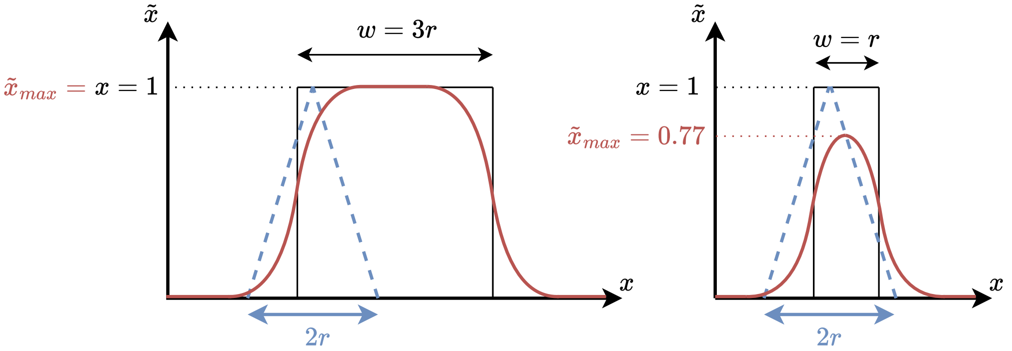


Figure 6.3. Linear density filter. Member width larger than filter diameter (left). Member width smaller than filter diameter (right). Inspired by Olesen and Hermansen [2020].

6.4 Threshold projection

To reduce the amount of intermediate PD resulting from the linear density filter, Wang et al. [2010] proposes a threshold projection filter, as given in Eq. (6.13).

$$\tilde{\tilde{x}}_{(e)} = \frac{\tanh(\beta\eta) + \tanh(\beta(\tilde{x}_{(e)} - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1-\eta))} \tag{6.13}$$

η is a threshold parameter, determining the cutoff value between projections. PDs above η are projected towards 1, while values below are projected towards 0. β determines how aggressively it is projected. As $\beta \to \infty$, the step-function is approached [Wang et al., 2010]. Eq. (6.13) is a generalization of the filters introduced in Guest et al. [2004] and Sigmund [2007]. For $\eta = 0$, the minimum length scale is enforced on the void regions [Sigmund, 2007], while for $\eta = 1$ it is enforced on the solid region [Guest et al., 2004]. This can also be seen from Fig. 6.3. If all density filtered PDs, $\tilde{x}$, below 1 are projected to 0, then the left one would violate the minimum length-scale of $2r$. If all above 0 is projected to 1, then the right one would comply with the minimum length scale. However, in both cases, the total volume of the structure would change, as the area is not the same as for the unfiltered variables x. In order to achieve the same volume of the structure after the filter is applied, the threshold parameter η should be equal to 0.5. [Olesen and Hermansen, 2020]

For $\eta = 0.5$, the influence of varying the projection parameter β can be seen in Fig. 6.4.

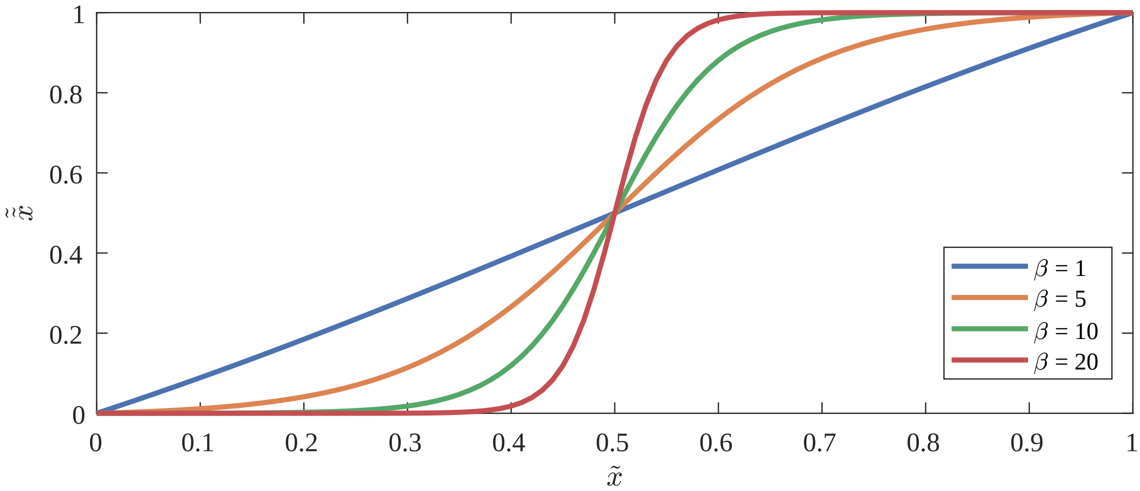


Figure 6.4. Influence of varying values of β on the steepness of the threshold projection filter.

When using threshold projection filtering, the length scale previously achieved through linear filtering is lost [Wang et al., 2010]. This can be alleviated through the introduction of robust TO. This places an implicit length scale and ensures robustness relative to manufacturing variations, at the cost of running three concurrent FE analyses [Wang et al., 2010]. However, the implicit length scale may not be ensured for complex optimization problems, and it warrants a numerical study on the influence of the parameters since they may be problem-specific [Li et al., 2023]. Therefore, robust TO will not be used in this project. An alternative approach is to ensure the minimum length scale through geometric constraints. In Zhou et al. [2015], the length scale is defined by requiring that the topology does not change between an eroded and dilated design, in a way reminiscent of robust TO. Another geometric approach to ensure minimum length scale is proposed by Li et al. [2023], where the mean PD within the filter radius is used as a length scale constraint. More information regarding the length scale constraint by Li et al. [2023] can be found in Appendix B.

6.5 Domain extension

Another problem entailing the minimum length scale is the implicit introduction of Neumann boundary conditions on the PD field, resulting from the density filtering. The problem manifests itself in three ways; Minimum length-scale is not satisfied at design domain boundaries, structural members are forced to be perpendicular to the aforementioned boundaries and will tend to 'stick' to them. This is due to element PDs close to boundaries not being correctly filtered with void elements. The relative weighting of these boundary elements PD's are increased to compensate for the reduced number of elements within the filter radius. This can be alleviated by increasing the size of the structural domain in all directions, except at boundary conditions, by a distance equal to the filter radius without increasing the design space. [Clausen and Andreassen, 2017]
In Fig. 6.5 a sketch of a 2D MBB beam with extended domains can be seen. The extended domains should only extend to a distance of r, the filter radius from Sec. 6.3, away from any non-symmetry deformation boundary conditions.

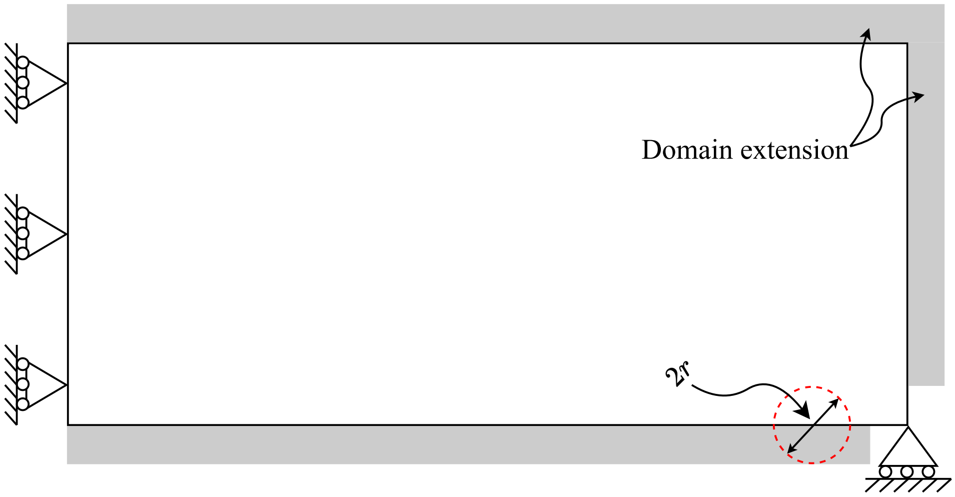


Figure 6.5. A sketch of an MBB beam along with the extended domain dependent on the filter size

Another approach to alleviate the implicit Neumann boundary conditions on the PD field along design domain boundaries is to apply explicit Dirichlet boundary conditions along with a Partial Differential Equation (PDE) filter [Lazarov and Sigmund, 2011]. However, the PDE filter approach has the drawback that symmetry, e.g. LHS of Fig. 6.5, can not be guaranteed after applying the filter [Clausen and Andreassen, 2017]. This effect occurs for design boundaries close to symmetry conditions. Additionally, Clausen and Andreassen [2017] concludes that minimum member size might not be enforced at design domain boundaries. Since symmetry conditions can not be ensured and minimum member size might not be enforced at design domain boundaries when using the PDE filter with explicit Dirichlet boundary conditions, the PDE filter will not be used.

Due to the FE domain being imported from an external program, as described in Chap. 4, it is non-trivial to correctly extend the domain in a consistent manner for general geometries and different filtering radii. Further, the increased computational time imposed by adding more elements to the mesh is not advantageous. Domain extension is therefore not included in the formulation and is discussed further in Chap. 9.

6.6 Design dependent loading

During sintering of an MBJ 3D printed specimen, the only force acting on the specimen is due to gravity. For the case of topology optimization, the total volume can change due to PD being redistributed by the optimizer. This effect should be reflected in the magnitude of the force due to gravity during optimization, see Eq. (6.14) where linear interpolation of the bulk density is included.

$$\{\mathbf{f_{ext}}\}_{(e)} = \tilde{\tilde{x}}_{(e)} \sum_{k=1}^{n_{ipts}} d_0 \rho_0 \, [\mathbf{N}]^T_{(e,k)} \, \{\mathbf{g}\} w_{(e,k)} \tag{6.14}$$

$\{\mathbf{f_{ext}}\}_{(e)}$ is the external load for the e$^{\text{th}}$ element, $[\mathbf{N}]_{(e,k)}$ is a matrix containing the element shape functions and $\{\mathbf{g}\}$ is a matrix of acceleration components e.g. gravitational acceleration. d_0 is the initial RD which is only used for sintering and thus not part of the formulation in the following literature.

Bruyneel and Duysinx [2005] and Munro [2024] introduced a lower bound on the volume constraint to eliminate the trivial solution of $\{\tilde{\tilde{\mathbf{x}}}\} = \{\mathbf{0}\}$. If the element does not have PD, no gravity acts on it Eq. (6.14), and therefore, it does not contribute to compliance. Further, Bruyneel and Duysinx [2005] uses the traditional SIMP approach Eq. (6.2). The SIMP approach uses a penalization factor of $p > 1$ for the stiffness interpolation, and often $p = 3$ is used. This leads to a difference in interpolation of the element stiffness and element density. Due to the difference in interpolation, at small values of $x_{(e)}$, the element density to element stiffness ratio can become large and lead to unbounded displacements, increased compliance and intermediate densities [Bruyneel and Duysinx, 2005]. Bruyneel and Duysinx [2005] also introduces a modified SIMP approach where linear interpolation of stiffness is performed for small values of $x_{(e)}$ only. The drawback of introducing a piece-wise interpolation is non-differentiability. Félix et al. [2020] uses the modified SIMP approach instead, cf. Eq. (6.4). The introduction of the minimum stiffness E_{min} alleviates the problem of unbounded displacements and singularities in the stiffness matrix for low PD values [Félix et al., 2020]. It might be possible to penalize intermediate densities further through the element density interpolation function such that intermediate densities are attributed with a larger than linear density, at the cost of possible unbounded displacements. To show the problem of unbounded displacements see Fig. 6.6, where the ratio μ_g/μ_E is the interpolation function for the density divided by the interpolation function for the stiffness. For traditional SIMP, $\mu_g = x_{(e)}$ and $\mu_E = (x_{(e)})^3$. The ratio is plotted for all material interpolation functions from Sec. 6.2, except for the linear interpolation.

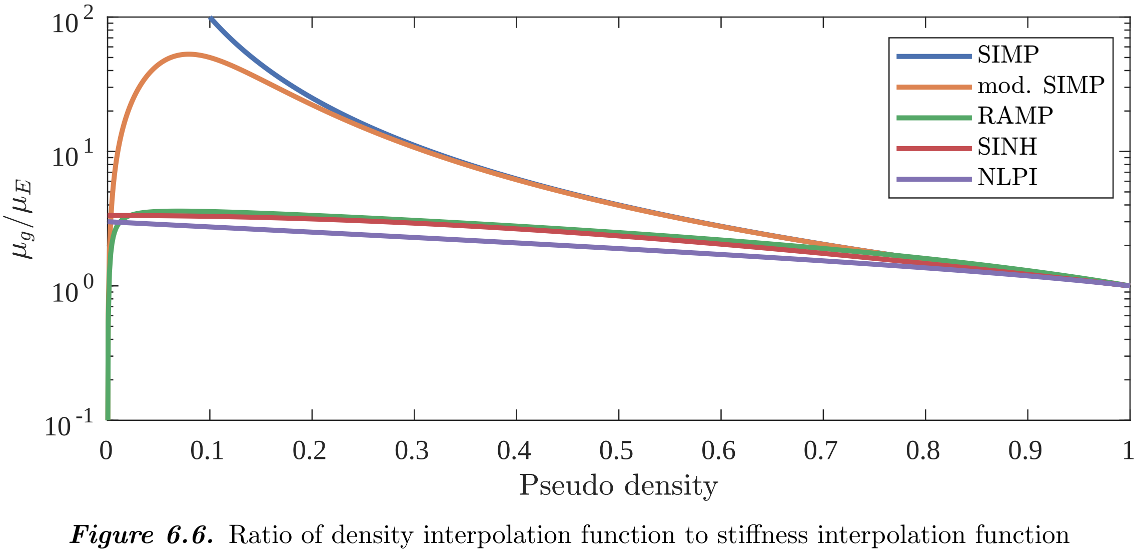


Figure 6.6. Ratio of density interpolation function to stiffness interpolation function

As seen in Fig. 6.6, the ratio of density to stiffness tends to infinity only for the SIMP method, for small values of PD. The modified SIMP response shows that for small values of PD, the ratio between gravity and stiffness interpolation still becomes large but does not go toward infinity. The other material interpolation functions do not behave in a manner that leads to unbounded displacements for low PD values.

Bruyneel and Duysinx [2005] also found that the optimum might be unconstrained w.r.t. the volume constraints. If a lower volume constraint and an upper volume constraint are defined for the topology optimization, the optimizer may incur a local minimum where neither constraint is active. The final design is thus independent of the volume constraints. As an example of designs that are independent of the given volume constraint, see Fig. 6.7-6.9, where a 2D linear FEA compliance TO program from [Sigmund, 2001][Troelsgaard et al., 2023] has been modified to use linear elastic material and gravitational design dependent loads on an 2D MBB beam like Fig. 5.7 but mirrored. Further, the modified SIMP model Eq. (6.4) is used with a linear density filter Eq. (6.11), a threshold projection filter Eq. (6.13) and the Method of Moving Asymptotes (MMA) optimizer is used [Svanberg, 1987]. The initial volume is set to 70% which is equal to the volume constraint.

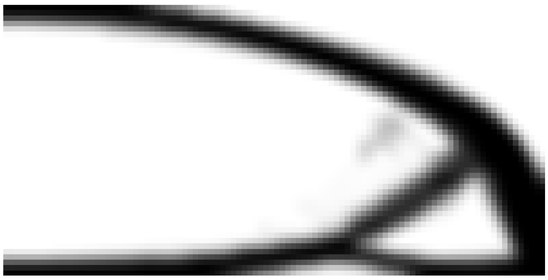


Figure 6.7. Design after 20 iterations, remaining volume is 25%.

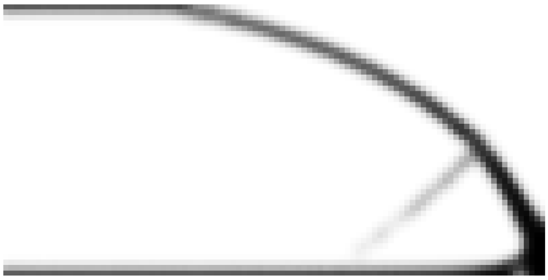


Figure 6.8. Design after 200 iterations, remaining volume is 7.4%.

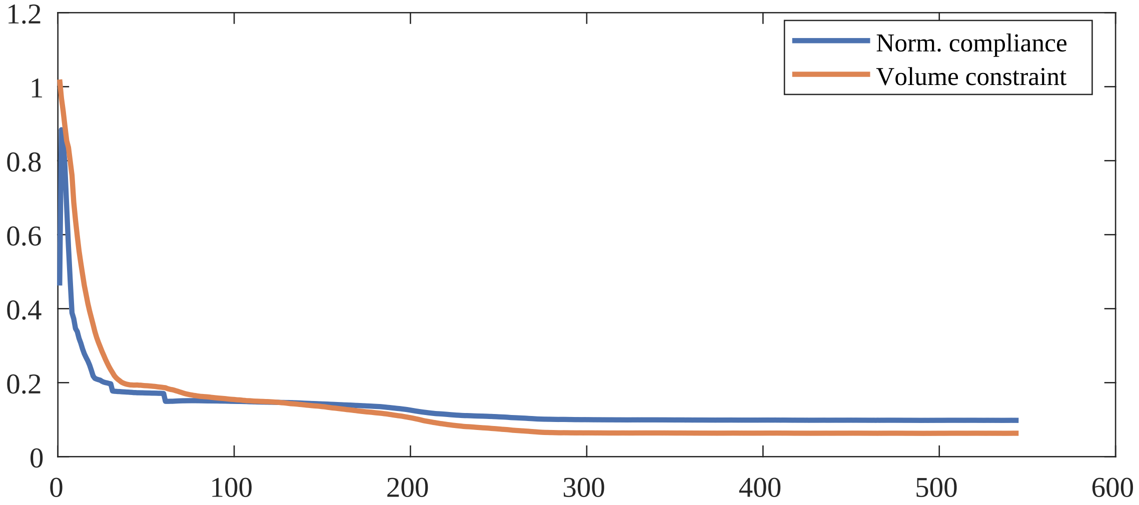


Figure 6.9. Plot of normalized compliance value and volume constraint. The two jumps in compliance at 30 and 60 iterations are due to continuation, see Sec. 6.7 for further information on continuation

Fig. 6.9 shows that the volume constraint is exactly active in the first iteration where the design domain has been initialized such that all finite elements have a PD of 0.7. From the first iteration onwards, the optimizer removes material. The optimization converged after 545 iterations to a non-physical design with many intermediate PDs.

To alleviate the problem of unconstrained volume, a minimum volume constraint can be included in the optimization formulation. Munro [2024] uses the two constraints cf. Eq. (6.15), where Eq. (6.15a) is the traditional maximum volume constraint used in compliance optimization with fixed loads and Eq. (6.15b) is the minimum volume constraint

$$g_1 = \frac{\sum_{e=1}^{n_{elem}} V^{(e)} x^{(e)}}{V_U \sum_{e=1}^{n_{elem}} V^{(e)}} - 1 \leq 0 \tag{6.15a}$$

$$g_2 = 1 - \frac{\sum_{e=1}^{n_{elem}} V^{(e)} x^{(e)}}{V_L \sum_{e=1}^{n_{elem}} V^{(e)}} \leq 0 \tag{6.15b}$$

V_U is the upper volume fraction and V_L is is the lower volume fraction.

For the same scenario as shown in Fig. 6.7 and Fig. 6.8 but with the lower constraint Eq. (6.15b) included and the lower volume fraction set at 0.4 can be seen in Fig. 6.10. All elements are initialized with a PD equal to the upper volume constraint of 0.7.

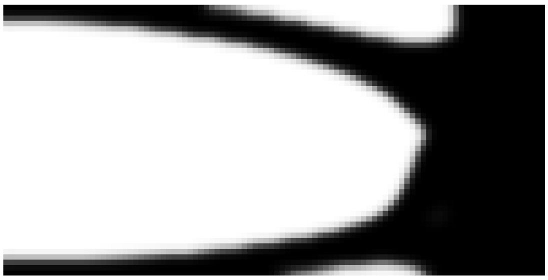


Figure 6.10. Converged optimizer design for $V_L = 0.4$

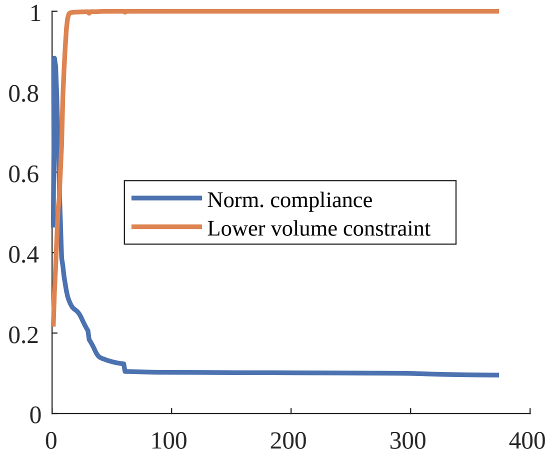


Figure 6.11. The optimizer converged after 374 iterations. The lower volume constraint was active at convergence iteration

The design shown in Fig. 6.10 intuitively makes sense, since most of the material is lumped nearest to the vertical support where it will contribute the least to increased compliance. Only the lower volume constraint is shown in Fig. 6.11 since it is the only active constraint. The constraint is active when it has a value of one.

A large task of the optimization procedure is contained in deriving the design sensitivities. This step further reveals some complications when dealing with PD-dependent volume forces, like gravity or rotational accelerations. The derivative of the linear elastic compliance on element level is given in Eq. (6.16) for geometrically linear FEA [Bruyneel and Duysinx, 2005].

$$\frac{\partial C}{\partial x_{(e)}} = 2(\{\mathbf{a}\}_{(e)})^T \frac{\partial \{\mathbf{f_{ext}}\}_{(e)}}{\partial x_{(e)}} - (\{\mathbf{a}\}_{(e)})^T \frac{\partial \, [\mathbf{k}]_{(e)}}{\partial x_{(e)}} \{\mathbf{a}\}_{(e)} \tag{6.16}$$

For fixed loads, the first part of the RHS of Eq. (6.16) vanishes since the external force is independent of the PD. When this term is neglected, the derivative of the compliance is monotonic with a constant negative sign. When design-dependent body forces are acting, the first part of the RHS does not vanish and the derivative can not be ensured to be monotonic, which can result in oscillatory compliance behavior during optimization. The choice of optimizer can alleviate the oscillatory compliance behavior by choosing an optimizer that does not make monotonous approximations to the original problem. [Bruyneel and Duysinx, 2005]

However as seen in Figs. 6.7 to 6.11, initial testing showed no oscillatory compliance response when using the MMA solver.

6.7 Continuation

By employing the filtering technique from Secs. 6.3 to 6.4, the problem may display improved behavior. However, it may still have multiple local minima, due to the inherent non-convexity of the penalized problem. As a result, it may converge prematurely to an inferior local minimum. Employing a continuation approach is advantageous to address this issue by gradually introducing the non-linearity, as seen in, e.g., Stolpe and Svanberg [2001b]. [Sigmund and Petersson, 1998]

This, in turn, may increase the computational time, due to having to find multiple local minima throughout the optimization. Continuation is a method that can be applied on several parameters, e.g., PD penalization [Félix et al., 2020], allowable constraint violation [Li et al., 2023] or stress penalization [Olesen et al., 2021]. Most commonly, continuation is used on the threshold projection steepness β, cf. Sec. 6.4, to gradually increase how aggressively PDs are projected from intermediate to binary [Olesen et al., 2021][Le et al., 2010][Wu et al., 2017]. While the effect of employing different continuation methods may differ, the motivation of these strategies is generally the same: to gradually restrict the design space, thereby promoting a strong minimum and a stable convergence. Stolpe and Svanberg [2001b] investigated the continuation of the PD penalization and showed that for problems with a unique binary solution, the trajectories of the design variables may be discontinuous. Furthermore, for certain problems, a binary solution may be infeasible, no matter how large the penalization on intermediate values is. Continuation methods provide good results in practice [Félix et al., 2020][Olesen et al., 2021][Oest and Lund, 2017]. This can be exemplified in Figs. 6.12 to 6.13. In Fig. 6.12 a constant $\beta = 30$ is used, whereas in Fig. 6.13 continuation is used, such that β is doubled each fifth iteration, starting with $\beta = 1$ but is limited to a maximum $\beta_{max} = 30$. It can be seen that the resulting geometries are qualitatively different, while the one with continuation has converged to a lower compliance, however using more iterations.

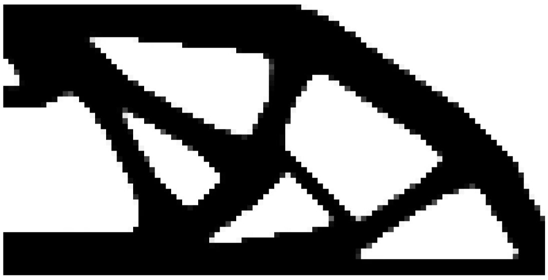


Figure 6.12. Compliance minimized MBB without continuation. Compliance is 1.598 Nm. Converged in 50 iterations.

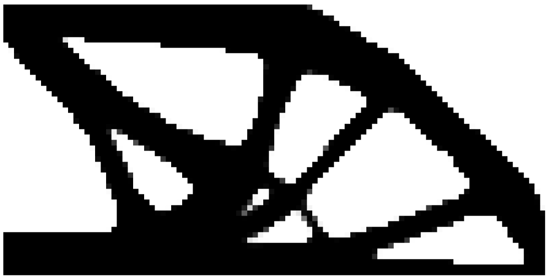


Figure 6.13. Compliance minimized MBB with continuation. Compliance is 1.583 Nm. Converged in 104 iterations.

6.8 Penalization

As explained in Sec. 6.1, topology optimization seeks a binary material or PD distribution. However, due to the ill-posedness of the discrete problem, it is relaxed to allow for intermediate values. Subsequently, elements of intermediate PDs are penalized, as illustrated in Sec. 6.2, to drive the optimizer to an integer solution. This method of relaxation and penalization was initially introduced by Bendsøe [1989], where the elasticity tensor was scaled by the penalized interpolation function. This naturally begs the question of how to include penalization of the sinter material model, cf. Chap. 5.

Penalization of tangent stiffness

As given in Eq. (5.12) and explicitly shown in Eq. (5.10), the stress is linear in the strain increments and constant in the pseudo-load. Eq. (5.12) is restated here for clarity.

$$\{\mathbf{\Delta S}\}^{(i,j)}_{(e,k)} = \left[\mathbf{D^{mod}}\right]^{(i)}_{(e,k)} \{\mathbf{\Delta}\boldsymbol{\varepsilon}\}^{(i,j)}_{(e,k)} - \{\mathbf{q}\}^{(i)}_{(e,k)}$$

Thus, for the stress to be consistent with the strains, which is necessary for the GNLFEA, $\left[\mathbf{D^{mod}}\right]^{(i)}_{(e,k)}$ has to be interpolated. This would mean that for $\tilde{\tilde{x}}_{(e)} \to 0$, then $\{\mathbf{\Delta S}\}^{(i,j)}_{(e,k)} \to -\{\mathbf{q}\}^{(i)}_{(e,k)}$. Therefore, the stress would not vanish alongside the density. Non-zero stress in void elements is unphysical. Thus, it would be appealing if the penalization resulted in $\{\mathbf{\Delta S}\}^{(i,j)}_{(e,k)} \to \{\mathbf{0}\}$ for $\tilde{\tilde{x}}_{(e)} \to 0$. Observing Eq. (5.10), it can be seen that $\{\mathbf{\Delta S}\}^{(i,j)}_{(e,k)}$ is a linear function in $\left[\mathbf{D^{mod}}\right]^{(i)}_{(e,k)}$, both in terms of the pseudo-load and strain increment. Thus, applying the interpolation directly to $\left[\mathbf{D^{mod}}\right]^{(i)}_{(e,k)}$ would result in a vanishing stress for a vanishing PD.

For the optimization to favor solid over void elements, intermediate elements should provide inefficient stiffness relative to the volume occupied for compliance-based optimization objectives [Bendsøe, 1989]. To investigate whether this holds for penalization on $\left[\mathbf{D^{mod}}\right]^{(i)}_{(e,k)}$, the benchmark test used in Sec. 5.3.3 is reused. A traction force is applied to the top of the beam, analogous to the usual boundary conditions for an MBB beam. The material interpolation scheme used is a modified SIMP approach with $p = 3$ and x_{min} = 1e-3 , as described in Sec. 6.2. The analysis is executed with a constant and uniform PD field throughout the structure of each analysis. Three different analyses are compared, with a PD of 1.0, 0.5 and 0.25, respectively. The results are given in Figs. 6.14 to 6.15.

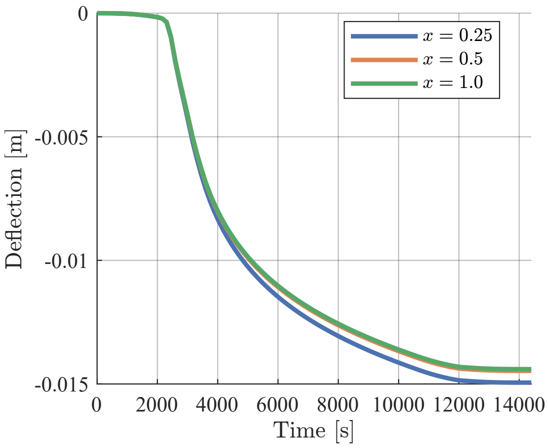


Figure 6.14. Comparison of bending deformation.

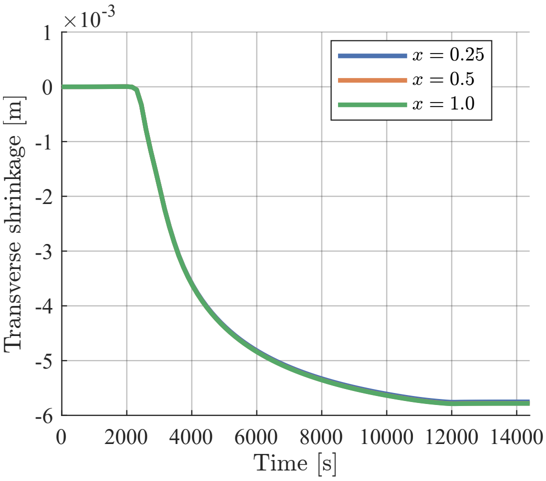


Figure 6.15. Comparison of transverse deformation.

As is apparent from the result, very little difference can be seen between the full- and intermediate densities. Thus, penalization of $\left[\mathbf{D}^{\mathbf{mod}}\right]_{(e,k)}^{(i)}$ does not provide the expected and sought-after outcome, that intermediate densities result in larger deformations. This could be due to the fact the lower stress contribution from the mechanical strain is equally balanced by higher stress from the thermally induced strain, since penalization of $\left[\mathbf{D}^{\mathbf{mod}}\right]_{(e,k)}^{(i)}$ scales the two contributions equally.

Penalization of internal force

In the works by Lindgaard and Dahl [2013], modified SIMP is used to penalize the whole element tangent stiffness and residual contribution equally. This was done to circumvent artificial buckling modes in low-density regions and ensure consistency between the global residual and tangent stiffness matrix. Thus, this method of penalization is tested. A different benchmark example is used, compared to Figs. 6.14 to 6.15, whereby the linear gravity interpolation is introduced, as given in Sec. 6.6. Furthermore, the internal force and tangent stiffness are penalized by RAMP, cf. Sec. 6.2. The results can be seen in Figs. 6.16 to 6.17.

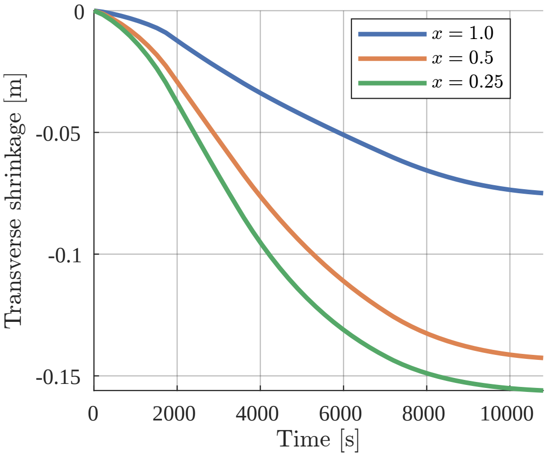


Figure 6.16. Comparison of bending deformation.

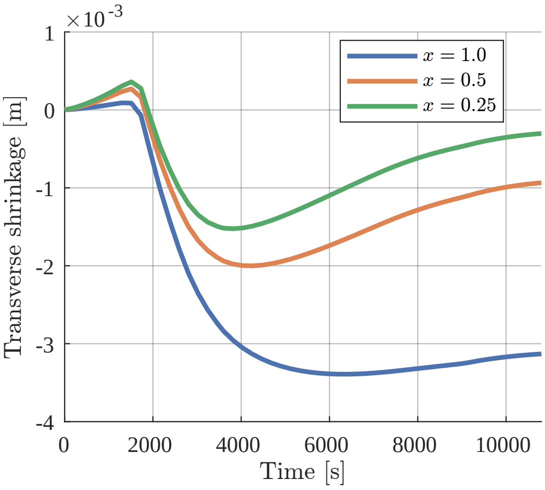


Figure 6.17. Comparison of transverse deformation.

Since a lower PD results in a larger deflection, it is therefore concluded that penalization of $\{\mathbf{f_{int}}\}_{(e)}^{(i,j)}$ provides the sought-after behavior. The final deformation of the structures in Figs. 6.16 to 6.17 can be seen in Fig. 6.18.

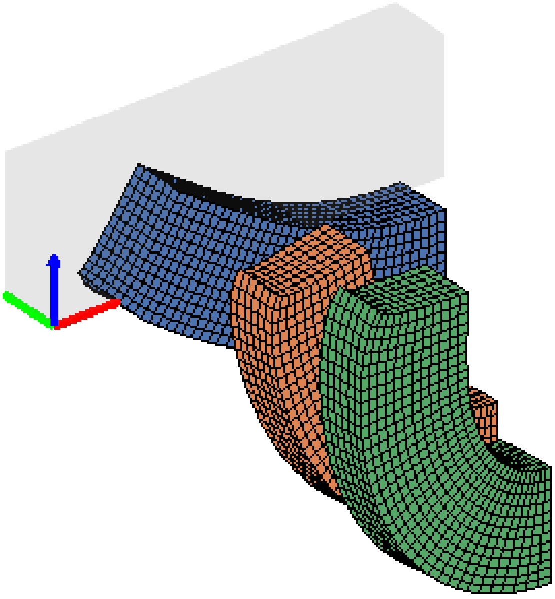


Figure 6.18. Deformation of structure for different PDs; 1.0 (blue), 0.5 (orange) and 0.25 (green).

6.9 Objective formulation

Correctly formulating the objective is paramount to achieving the desired outcome.
To the author's knowledge, no previous attempts have been pursued to use topology optimization in the context of sintering. Consequently, it is also not obvious which objective function to minimize, to solve a practical problem, that produces the desired outcome. Therefore, the effort may be to develop a framework for implementing and testing several different objectives. In the present section, several different formulations of objectives that may or may not be useful will be presented. These will be either based on the author's heuristical knowledge of the underlying problem or on the literature of similar problems. These are all implemented, tested and compared to gauge which formulations show sensible behavior. A further discussion on choice of objective function is given in Chap. 9. Returning to the problem statement, it is formulated as:

How can density-based topology optimization be used to reduce the deformation of MBJ 3D-printed components due to sintering?

Thus, the objective is to *reduce deformation.* This may be implemented in a similar fashion as in, e.g., Troelsgaard and Elmstrøm [2023], given in Eq. (6.17).

$$f_1 = \frac{1}{2}\{\mathbf{U}\}^T\{\mathbf{U}\} \tag{6.17}$$

Which is the form of the usual least-squares formulation for a reference deformation of zero. This would, in practice, result in a structure with the lowest deformation. It may, however, prove impractical since this would prefer elements with constrained DOFs and likely result in full elements on symmetry planes.
In the absence of friction, the cause of the undesired deformation are due to the effect of gravity. Thus, a similar approach may be employed by maximizing the apparent stiffness of the structure to the force of gravity. As mentioned earlier, a common objective encountered in topology optimization is minimizing compliance. This has also been employed in the context of self-weight by, e.g., Bruyneel and Duysinx [2005], Félix et al. [2020] and Munro [2024]. The formulation of such an objective is given in Eq. (6.18).

$$f_2 = \{\mathbf{F_{ext}}\}^T\{\mathbf{U}\} \tag{6.18}$$

This may however not necessarily reduce deformation in the directions orthogonal to the gravity load. A more general approach, that captures deformation in all directions may be sought by instead minimizing the strain energy of the structure. For linear elasticity, minimizing the compliance of a structure is equal to minimizing the strain energy of that structure. This is given in the usual continuous tensor format in Eq. (6.19)

$$W = \int_V S_{ij}\varepsilon_{ij}dV \tag{6.19}$$

The work can also be stated in integral form, as used in e.g. Alberdi et al. [2018], as given in Eq. (6.20).

$$f_3 = W = \int_t \int_V S_{ij}\dot{\varepsilon}_{ij}dV dt \tag{6.20}$$

The main contribution to Eq. (6.20) is expected to be due to viscoplastic straining. Thus, it bears a resemblance to optimizing for plastic work due to the non-reversible viscoplastic straining. Alberdi et al. [2018] maximized total plastic work for small-strain pressure-dependent Drucker-Prager plasticity and similarly maximized total plastic work for small-strain visco-plasticity. The optimization resulted in different qualitative structures, highlighting the difference between constitutive models with the same objective functions.

However, as is apparent from Chap. 3, the viscoplastic strain rate consists of two components: dilation and distortion. Dilation is the shrinkage primarily caused by the geometry-independent sintering stress, which is a necessary component of the thermal sintering process. The second component is distortion, which causes an undesirable change in shape during sintering. Thus, the objective can more accurately be reformulated to *reduce distortion.*

Distortion is caused by deviatoric work, which is given in tensor notation in Eq. (6.21).

$$f_4 = W_{dev} = \int_t \int_V S'_{ij} \dot{\varepsilon}'_{ij} dV dt \tag{6.21}$$

Liu et al. [2019] minimized distortional strain energy density, by aggregating the deviatoric strain energy of each element using a P-norm approach. Deng et al. [2019] similarly minimized distortional strain energy using a P-norm approach, however, by relaxing the deviatoric strain energy in each element, similar to a q-p approach [Bruggi, 2008], to avoid excessive distortional energy contribution of elements whose PD approach zero.

Eqs. (6.17) to (6.21) constitutes the trial objective functions that are going to be used for comparison. It should be noted that the goal is to develop a formulation where an arbitrary objective can easily be implemented.

6.10 Summary

In the remainder of this report, RAMP will be used to penalize the element level internal force $\{\mathbf{f_{int}}\}_{(e)}^{(i,j)}$ and element level tangent stiffness $[\mathbf{k_t}]_{(e)}^{(i,j)}$ directly. Four different trial objective functions Eqs. (6.17) to (6.21) will be tested. These being the squared deformation, compliance, strain energy and deviatoric work.

A linear density filter and threshold projection with continuation on the projection parameter β will be used. Linear gravity interpolation along with an upper and lower bound on the permissible volume will be used. The solution will be found by sequential convex programming utilizing MMA as briefly discussed in Sec. 6.6 with further information about the optimizer given in Appendix C. The design sensitivities will be calculated in the following chapter.

7 Design sensitivity analysis

When using gradient-based optimizers such as MMA, it is necessary to supply the optimizer with gradient information about the topology optimization problem. Multiple approaches can be applied to find the gradients either analytically, by Finite Difference approximation or by Automatic Differentiation. In this work, the Design Sensitivity Analysis (DSA) will be analytically based due to its favorable computational speed compared to, e.g. Finite Difference approximations. As described in Chap. 1 and Chap. 5, the deformation of MBJ 3D printed parts as a response to sintering can be described by a viscoplastic constitutive model. Due to the viscoplastic response, the material model is path-dependent which means that the sensitivity analysis likewise becomes path-dependent.

When the problem of deriving analytical sensitivities for topology optimization arises, two methods can be used. The two methods are, the Adjoint Method and the Direct Differentiation Method. Both methods yield the same sensitivity expressions; however, depending on the optimization formulation, one method is computationally faster than the other. In topology optimization, the number of design variables (Pseudo-Densities) often far outweighs the number of objective functions and constraints, making the adjoint method more efficient [Alberdi et al., 2018]. The path-dependent adjoint method is not trivial and will be introduced in Sec. 7.1. In Sec. 7.2, the terms used in the path-dependent adjoint method are specified for the sintering analysis. In Secs. 7.3 to 7.5, the necessary sensitivity terms will be derived and finally, the sensitivities will be verified against Forward Difference (FD) approximated sensitivities to check the implementation and derived sensitivity terms are correct.

7.1 Path dependent adjoint formulation

This section is based on the adjoint formulation of path-dependent topology optimization problems from [Michaleris et al., 1994] [Alberdi et al., 2018]. The notation will generally follow from [Alberdi et al., 2018] but with small modifications.

For a GNLFEA topology optimization problem with linear elastic materials, only the structural residual $\{\mathbf{R}\}^{(i)}$ from the GNLFEA formulation is an implicit constraint. Since the 3D-printed MBJ material exhibits a viscoplastic response to sintering, the material model becomes path-dependent. Therefore, both the global residual $\{\mathbf{R}\}^{(i)}$ and the local $\{\mathbf{H}\}^{(i)}$ residual from Chap. 5 are implicit constraints of the topology optimization problem. Due to the incremental description of both the GNLFE analysis and the material model, the implicit constraints can be described as functions of the design variables, the current global variables $\{\mathbf{U}\}^{(i)}$, the previous global variables $\{\mathbf{U}\}^{(i-1)}$, the current value of the local variables $\{\mathbf{c}\}^{(i)}$ and the previous values of the local variables $\{\mathbf{c}\}^{(i-1)}$, see Eq. (7.1). The structural analysis is performed using the projection filtered PDs, $\{\tilde{\tilde{\mathbf{x}}}\}$, as defined in Secs. 6.3 to 6.4. Thus, the global residual $\{\mathbf{R}\}^{(i)}$ and local residual $\{\mathbf{H}\}^{(i)}$ are functions of these filtered PDs.

$$\{\mathbf{R}(\{\tilde{\tilde{\mathbf{x}}}\}, \{\mathbf{U}\}^{\mathbf{(i)}}, \{\mathbf{U}\}^{\mathbf{(i-1)}}, \{\mathbf{c}\}^{\mathbf{(i)}}, \{\mathbf{c}\}^{\mathbf{(i-1)}})\}^{(i)} = \{\mathbf{0}\} \tag{7.1a}$$

$$\{\mathbf{H}(\{\tilde{\tilde{\mathbf{x}}}\}, \{\mathbf{U}\}^{\mathbf{(i)}}, \{\mathbf{U}\}^{\mathbf{(i-1)}}, \{\mathbf{c}\}^{\mathbf{(i)}}, \{\mathbf{c}\}^{\mathbf{(i-1)}})\}^{(i)} = \{\mathbf{0}\} \tag{7.1b}$$

For each constraint in $\{\mathbf{R}\}^{(i)}$ and $\{\mathbf{H}\}^{(i)}$ there is a corresponding global $\{\mathbf{U}\}^{(i)}$ and local $\{\mathbf{c}\}^{(i)}$ variable. The local constraints enforce that the material model is satisfied in each time step analogous to how the global constraint $\{\mathbf{R}\}^{(i)}$ from the GNLFEA analysis enforces equilibrium between the internal and external forces. Note that in Eq. (7.1) the superscript (j) from Sec. 4.1 that describes equilibrium iterations are not necessary since only values from converged load or time steps are used in the adjoint formulation.

The global residual $\{\mathbf{R}\}^{(i)}$ is equal to the force residual from the FEA, and the global variables are therefore the nodal displacements from the FEA as defined in Eq. (7.2).

$$\{\mathbf{U}\}^{(i)} = \left\{ a^{(i)}_{(1)} \quad ... \quad a^{(i)}_{(n_{dofs})} \right\}^T \tag{7.2}$$

The vector $\{\mathbf{H}\}^{(i)}$ contains the constraints for the internal variables. The internal variables are defined at each integration point of each element. Therefore, the constraints on the local variables are also formulated at the integration point level. The size of $\{\mathbf{H}\}^{(i)}$ will generally be larger than the size of $\{\mathbf{R}\}^{(i)}$. The structure of the local constraint vector $\{\mathbf{H}\}^{(i)}$ can be seen in Eqs. (7.3) to (7.4).

$$\{\mathbf{H}\}^{(i)} = \left\{ \{\mathbf{h}\}^{(i)}_{(1)} \quad ... \quad \{\mathbf{h}\}^{(i)}_{(n_{elm})} \right\}^T \quad \textit{where} \tag{7.3}$$

$$\{\mathbf{h}\}^{(i)}_{(e)} = \left\{ \{\mathbf{h}\}^{(i)}_{(e,1)} \quad ... \quad \{\mathbf{h}\}^{(i)}_{(e,n_{ipts})} \right\}^T \tag{7.4}$$

The elementwise local constraint vector $\{\mathbf{h}\}^{(i)}_{(e)}$ has an entry for each integration point in the element. In this case, full integration of hexahedral isoparametric elements is used; therefore there are 8 integration points for each element. The components of the elementwise local constraint vector $\{\mathbf{h}\}^{(i)}_{(e,k)}$ are themselves vectors that can contain multiple local constraints for each integration point. The integration point level constraints are formulated identically for all integration points in the FE problem i.e., the formulation of $\{\mathbf{h}\}^{(i)}_{(e,1)}$ is equal to $\{\mathbf{h}\}^{(i)}_{(e,2)}$ and also equal to $\{\mathbf{h}\}^{(i)}_{(e,n_{ipts})}$.

For each integration point level local constraint that is included in $\{\mathbf{H}\}^{(i)}$ a corresponding local variable collected in $\{\mathbf{c}\}^{(i)}$ must be included as well. These are similar to $\{\mathbf{h}\}^{(i)}_{(e)}$ a vector in each integration point in each element but are formulated equally for all integration points and all elements. The structure of $\{\mathbf{c}\}^{(i)}$ follows from $\{\mathbf{H}\}^{(i)}$ and are shown in Eqs. (7.5) to (7.6).

$$\{\mathbf{c}\}^{(i)} = \left\{ \{\mathbf{c}\}^{(i)}_{(1)} \quad ... \quad \{\mathbf{c}\}^{(i)}_{(n_{elm})} \right\}^T \quad \textit{where} \tag{7.5}$$

$$\{\mathbf{c}\}^{(i)}_{(e)} = \left\{ \{\mathbf{c}\}^{(i)}_{(e,1)} \quad ... \quad \{\mathbf{c}\}^{(i)}_{(e,n_{ipts})} \right\}^T \tag{7.6}$$

The formulation of the integration point local constraints ensures that the constraint $\{\mathbf{h}\}^{(i)}_{(e,y)}$ is only a function of the local variable $\{\mathbf{c}\}^{(i)}_{(e,j)}$ in the same integration point, i.e. when $y = j$. However, $\{\mathbf{h}\}^{(i)}_{(e,y)}$ can be a function of the local variables from previous as well as current time steps i.e. both $\{\mathbf{c}\}^{(i)}_{(e,y)}$ and $\{\mathbf{c}\}^{(i-1)}_{(e,y)}$.

In path-dependent topology optimization, the objective function may also be dependent on values from all load or time steps, e.g., the time integral in Eqs. (6.20) to (6.21). Consider a general objective function that is dependent on global variables $\{\mathbf{U}\}^{(i)}$, in this case, nodal deformation, and local variables $\{\mathbf{c}\}^{(i)}$, in this case, material model variables:

$$f(\{\tilde{\tilde{\mathbf{x}}}\}) = F(\{\tilde{\tilde{\mathbf{x}}}\}\{\mathbf{U}\}^{(1)},..,\{\mathbf{U}\}^{(n)},\{\mathbf{c}\}^{(1)},..,\{\mathbf{c}\}^{(n)})) \tag{7.7}$$

n denotes the final load or time step. For GNLFE topology optimization problems with path-dependent nonlinear material response, a Lagrangian function $\hat{f}$ can be formulated as Eq. (7.8).

$$\hat{f} = f(\{\tilde{\tilde{\mathbf{x}}}\}) + \sum_{i=1}^{n} \{\boldsymbol{\lambda}\}^{(i)^T}\{\mathbf{R}\}^{(i)} + \sum_{i=1}^{n} \{\boldsymbol{\mu}\}^{(i)^T}\{\mathbf{H}\}^{(i)} \tag{7.8}$$

$\{\mathbf{R}\}^{(i)}$ is the global constraint at loadstep i, $\{\mathbf{H}\}^{(i)}$ is the internal material residual constraint at loadstep i and $\{\boldsymbol{\lambda}\}^{(i)^T}$ and $\{\boldsymbol{\mu}\}^{(i)^T}$ are both Lagrange multipliers associated with the global structural constraint and material equilibrium constraint respectively.

Next, the full derivative of the Lagrangian function is taken w.r.t. the design variables, see Eq. (7.9).

$$\begin{aligned}
\frac{d\hat{f}}{d\{\tilde{\tilde{\mathbf{x}}}\}} &= \frac{\partial F}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \sum_{i=1}^{n}\left(\frac{\partial\{\mathbf{F}\}}{\partial\{\mathbf{U}\}^{(i)}}\frac{d\{\mathbf{U}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{F}\}}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}\right) \\
&+ \sum_{i=1}^{n}\left(\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}}\frac{d\{\mathbf{U}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i-1)}}\frac{d\{\mathbf{U}\}^{(i-1)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}\right. \\
&\qquad \left. + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}}\frac{d\{\mathbf{c}\}^{(i-1)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}\right)\lambda^{(i)T} \\
&+ \sum_{i=1}^{n}\left(\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}}\frac{d\{\mathbf{U}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i-1)}}\frac{d\{\mathbf{U}\}^{(i-1)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}\right. \\
&\qquad \left. + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}}\frac{d\{\mathbf{c}\}^{(i-1)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}\right)\mu^{(i)T}
\end{aligned} \tag{7.9}$$

By shifting the summation indices, the derivatives of $\{\mathbf{U}\}^{(0)}$ and $\{\mathbf{c}\}^{(0)}$ can be avoided, since they are initial conditions and thus do not contribute to the sensitivity. Then, the end state of $i = n$ must be considered separately. Further, Eq. (7.9) can be rewritten such that the implicit derivatives $\frac{d\{\mathbf{U}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}$ and $\frac{d\{\mathbf{c}\}^{(i)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}$ are factorized, see Eq. (7.10).

$$
\begin{aligned}
\frac{d\hat{f}}{d\{\tilde{\tilde{\mathbf{x}}}\}} = \frac{\partial F}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} &+ \sum_{i=1}^{n}\left(\{\boldsymbol{\lambda}\}^{(i)T}\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \{\boldsymbol{\mu}\}^{(i)T}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}}\right) \\
&+ \left(\frac{\partial F}{\partial\{\mathbf{U}\}^{(n)}} + \{\boldsymbol{\lambda}\}^{(n)T}\frac{\partial\{\mathbf{R}\}^{(n)}}{\{\mathbf{U}\}^{(n)}} + \{\boldsymbol{\mu}\}^{(n)T}\frac{\partial\{\mathbf{H}\}^{(n)}}{\{\mathbf{U}\}^{(n)}}\right)\frac{d\{\mathbf{U}\}^{(n)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} \\
+\sum_{q=1}^{n-1}&\left[\frac{\partial F}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\lambda}\}^{(q+1)T}\frac{\partial\{\mathbf{R}\}^{(q+1)}}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q+1)T}\frac{\partial\{\mathbf{H}\}^{(q+1)}}{\partial\{\mathbf{U}\}^{(q)}}\right. \\
&\left. + \{\boldsymbol{\lambda}\}^{(q)T}\frac{\partial\{\mathbf{R}\}^{(q)}}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q)T}\frac{\partial\{\mathbf{H}\}^{(q)}}{\partial\{\mathbf{U}\}^{(q)}}\right]\frac{d\{\mathbf{U}\}^{(q)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} \\
&+ \left(\frac{\partial F}{\partial\{\mathbf{c}\}^{(n)}} + \{\boldsymbol{\lambda}\}^{(n)T}\frac{\partial\{\mathbf{R}\}^{(n)}}{\{\mathbf{c}\}^{(n)}} + \{\boldsymbol{\mu}\}^{(n)T}\frac{\partial\{\mathbf{H}\}^{(n)}}{\{\mathbf{c}\}^{(n)}}\right)\frac{d\{\mathbf{c}\}^{(n)}}{d\{\tilde{\tilde{\mathbf{x}}}\}} \\
+\sum_{q=1}^{n-1}&\left[\frac{\partial F}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\lambda}\}^{(q+1)T}\frac{\partial\{\mathbf{R}\}^{(q+1)}}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q+1)T}\frac{\partial\{\mathbf{H}\}^{(q+1)}}{\partial\{\mathbf{c}\}^{(q)}}\right. \\
&\left. + \{\boldsymbol{\lambda}\}^{(q)T}\frac{\partial\{\mathbf{R}\}^{(q)}}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q)T}\frac{\partial\{\mathbf{H}\}^{(q)}}{\partial\{\mathbf{c}\}^{(q)}}\right]\frac{d\{\mathbf{c}\}^{(q)}}{d\{\tilde{\tilde{\mathbf{x}}}\}}
\end{aligned}
\tag{7.10}
$$

From Eq. (7.10) it is possible to identify four equations which must at all times be zero, by appropriate choice of the Lagrangian multipliers λ^T and μ^T, such that the implicit derivatives are eliminated. These four adjoint equations can be seen in Eq. (7.11).

$$
\frac{\partial F}{\partial\{\mathbf{U}\}^{(n)}} + \{\boldsymbol{\lambda}\}^{(n)T}\frac{\partial\{\mathbf{R}\}^{(n)}}{\partial\{\mathbf{U}\}^{(n)}} + \{\boldsymbol{\mu}\}^{(n)T}\frac{\partial\{\mathbf{H}\}^{(n)}}{\partial\{\mathbf{U}\}^{(n)}} = \{\mathbf{0}\} \tag{7.11a}
$$

$$
\frac{\partial F}{\partial\{\mathbf{c}\}^{(n)}} + \{\boldsymbol{\lambda}\}^{(n)T}\frac{\partial\{\mathbf{R}\}^{(n)}}{\partial\{\mathbf{c}\}^{(n)}} + \{\boldsymbol{\mu}\}^{(n)T}\frac{\partial\{\mathbf{H}\}^{(n)}}{\partial\{\mathbf{c}\}^{(n)}} = \{\mathbf{0}\} \tag{7.11b}
$$

$$
\begin{aligned}
\frac{\partial F}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\lambda}\}^{(q+1)T}\frac{\partial\{\mathbf{R}\}^{(q+1)}}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q+1)T}\frac{\partial\{\mathbf{H}\}^{(q+1)}}{\partial\{\mathbf{U}\}^{(q)}}& \\
+\{\boldsymbol{\lambda}\}^{(q)T}\frac{\partial\{\mathbf{R}\}^{(q)}}{\partial\{\mathbf{U}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q)T}\frac{\partial\{\mathbf{H}\}^{(q)}}{\partial\{\mathbf{U}\}^{(q)}} &= \{\mathbf{0}\}
\end{aligned}
\tag{7.11c}
$$

$$
\begin{aligned}
\frac{\partial F}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\lambda}\}^{(q+1)T}\frac{\partial\{\mathbf{R}\}^{(q+1)}}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(l+1)T}\frac{\partial\{\mathbf{H}\}^{(q+1)}}{\partial\{\mathbf{c}\}^{(q)}}& \\
+\{\boldsymbol{\lambda}\}^{(q)T}\frac{\partial\{\mathbf{R}\}^{(q)}}{\partial\{\mathbf{c}\}^{(q)}} + \{\boldsymbol{\mu}\}^{(q)T}\frac{\partial\{\mathbf{H}\}^{(q)}}{\partial\{\mathbf{c}\}^{(q)}} &= \{\mathbf{0}\}
\end{aligned}
\tag{7.11d}
$$

Eq. (7.11a)-(7.11b) must only be solved for the final converged time step, while Eq. (7.11c)-(7.11d) must be solved for all other converged time steps. The index q is defined in the interval $q \in [n-1,\ldots,3,2,1]$, i.e. the adjoint equations must be solved in a backward manner compared to the sintering analysis. The Lagrange multipliers are first determined for the final analysis time-step n and solved backward in time until the first time-step, i.e. a terminal value problem. When the adjoint equations have been solved and appropriate values have been found for the Lagrange multipliers, the sensitivity of the problem simplifies to Eq. (7.12):

$$
\frac{d\hat{f}}{d\{\tilde{\tilde{\mathbf{x}}}\}} = \frac{\partial F}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \sum_{i=1}^{n}\left(\{\boldsymbol{\lambda}\}^{(i)T}\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} + \{\boldsymbol{\mu}\}^{(i)T}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}}\right) \tag{7.12}
$$

It can be seen from Eq. (7.11) and Eq. (7.12) that only a few terms are dependent on the choice of objective, and therefore, additional objectives are easily implemented. To ensure the formulation is implemented correctly in the analysis, it is verified with a simple surrogate objective function for both a linear elastic material model (no local residuals) and an incrementally linear material model (using local residual). The results are given in Appendix D.

7.2 Application to sintering

The contents of the global and local constraints along with the global and local variables are in this section extended to the sintering material model introduced in Chap. 3 and Chap. 5. The $(\frac{\partial f}{\partial x})^{i-1}$ operator denotes that the derivative is evaluated with values from the previous time step indicated by $i-1$. The $()^{i-1}$ notation is used to improve readability since some second-order derivatives will occur. Further, like in Sec. 4.1 the $\mathcal{A}$ assembly operator will be used commonly in the following sections. The assembly operator simply denotes a consistent assembly of matrices or vectors. Many types of assembly are performed but the same $\mathcal{A}$ symbol is used to denote these.

The global constraint follows from the force residual from the FEA formulation in Sec. 4.1. It is redefined in Eqs. (7.13) to (7.16) since the equilibrium iteration index j is omitted and penalization is now included.

$$\{\mathbf{R}\}^{(i)} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \{\mathbf{r}\}^{(i)}_{(e)} \tag{7.13}$$

The elemental residual $\{\mathbf{r}\}^{(i)}_{(e)}$ is defined as Eq. (7.14).

$$\{\mathbf{r}\}^{(i)}_{(e)} = \{\mathbf{f_{int}}\}^{(i)}_{(e)} - \{\mathbf{f_{ext}}\}_{(e)} \tag{7.14}$$

There is no load-step index (i) in $\{\mathbf{f_{ext}}\}_{(e)}$ since it is constant during the structural analysis but not during the topology optimization. The elemental internal and external force is redefined in Eq. (7.15) and Eq. (7.16) and both include penalization like discussed in Sec. 6.2 and Sec. 6.8.

$$\{\mathbf{f_{ext}}\}_{(e)} = \sum_{k=1}^{n_{ipts}} \tilde{\tilde{x}}_{(e)} d_0 \rho \, [\mathbf{N}]^T_{(e,k)} \, \{\mathbf{g}\} w_{(e,k)} \tag{7.15}$$

$$\{\mathbf{f_{int}}\}^{(i)}_{(e)} = \sum_{k=1}^{n_{ipts}} v_{(e)} \left([\mathbf{B_{NL}}]^{(i)}_{(e.k)} \right)^T \{\mathbf{S}\}^{(i)}_{(e,k)} w_{(e,k)} \tag{7.16}$$

In Eq. (7.15) d_0 is the initial relative density as previously discussed in Chap. 2, $\tilde{\tilde{x}}$ is the projection filtered PD and in Eq. (7.16) v is the RAMP penalization of the PD's like discussed in Sec. 6.6 and is repeated in Eq. (7.17).

$$v_{(e)} = x_{min} + \frac{\tilde{\tilde{x}}_{(e)}}{1 + p\Big(1 - \tilde{\tilde{x}}_{(e)}\Big)} \Big(1 - x_{min}\Big) \tag{7.17}$$

The local constraints on integration point level are defined as Eq. (5.4) in the material model section but are repeated and rewritten in Eqs. (7.18) to (7.20).

$$\{\mathbf{h}\}^{(i)}_{(e,k)} = \begin{Bmatrix} \{\mathbf{h}_{\boldsymbol{\varepsilon}}\}^{(i)}_{(e,k)} \\ \{\mathbf{h}_{\boldsymbol{\kappa}}\}^{(i)}_{(e,k)} \end{Bmatrix} = \{\mathbf{0}\} \tag{7.18}$$

$$\begin{aligned} \{\mathbf{h}_{\boldsymbol{\varepsilon}}\}^{(i)}_{(e,k)} =& \Big([\mathbf{D}]^{-1} + \Delta t\Big(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\Big)^{i-1}\Big)\{\mathbf{\Delta S}\} + \Big(\Delta t\Big(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\Big)^{i-1}\Big)\{\boldsymbol{\Delta\kappa}\} \\ &+ \Delta t\Big(\Big(\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}\Big)^{i-1} - \Big(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T}\Big)^{i-1}\Delta T\Big) - \{\boldsymbol{\Delta\varepsilon}\} = \{\mathbf{0}\} \end{aligned} \tag{7.19}$$

$$\begin{aligned} \{\mathbf{h}_{\boldsymbol{\kappa}}\}^{(i)}_{(e,k)} =& \Big(-\Delta t\Big(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\Big)^{i-1}\Big)\{\mathbf{\Delta S}\} + \Big([\mathbf{I}] - \Delta t\Big(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\Big)^{i-1}\Big)\{\boldsymbol{\Delta\kappa}\} \\ &- \Delta t\Big(\Big(\{\dot{\boldsymbol{\kappa}}\}\Big)^{i-1} + \Big(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial T}\Big)^{i-1}\Delta T\Big) = \{\mathbf{0}\} \end{aligned} \tag{7.20}$$

In Eqs. (7.19) to (7.20) the element and integration point indices have been omitted for readability. In each integration point, eight local constraints are defined. Six of the constraints are due to the strain residual and the remaining two are due to the residual on the evolution of the internal variables G and d. For each local constraint in Eq. (7.18), a corresponding local variable is defined as seen in Eq. (7.21).

$$\{\mathbf{c}\}^{(i)}_{(e,k)} = \Big\{\{\mathbf{S}\}^{(i)}_{(e,k)} \quad \{\boldsymbol{\kappa}\}^{(i)}_{(e,k)}\Big\}^T \tag{7.21}$$

Objective function

As outlined in Sec. 6.9, the objective is to develop a framework where several objective functions can easily be implemented. In the current work, four different objective functions will be tested and compared. This section outlines the numerical implementation of these objectives. The first objective, f_1, being the squared deformation, given in Eq. (6.17), is only calculated in the terminal time-step, n. This is given in

$$f_1 = \big(\{\mathbf{U}\}^{(n)}\big)^T\{\mathbf{U}\}^{(n)} \tag{7.22}$$

Similarly, the compliance, cf. Eq. (6.18), is only calculated for the terminal time step, as given in Eq. (7.23).

$$f_2 = \{\mathbf{F_{ext}}\}^T\{\mathbf{U}\}^{(n)} \tag{7.23}$$

The n has been omitted from the external force, as the external force is constant through the structural analysis.

The strain energy, cf. Eq. (6.20), is calculated by summation of each integration point contribution. Similarly to e.g. [Alberdi et al., 2018], the strain energy is calculated through numerical integration by a trapezoidal rule.

$$\begin{aligned} f_3 = \sum_{i=1}^{n} W^{(i)} = \sum_{i=1}^{n}\sum_{e=1}^{n_{elm}} W^{(i)}_{(e)} \qquad\qquad & W^{(i)}_{(e)} = \sum_{k=1}^{n_{ipts}} W^{(i)}_{(e,k)} \\ W^{(i)}_{(e,k)} = \frac{1}{2}\Big(\{\mathbf{S}\}^{(i)}_{(e,k)} + \{\mathbf{S}\}^{(i-1)}_{(e,k)}\Big)^T & \Big(\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i-1)}_{(e,k)}\Big) w_{(e,k)} \end{aligned} \tag{7.24}$$

The deviatoric work can be similarly formulated as seen in Eq. (7.25).

$$\begin{aligned} f_4 &= \sum_{i=1}^{n} W_{dev}^{(i)} = \sum_{i=1}^{n}\sum_{e=1}^{n_{elm}} W_{dev_{(e)}}^{(i)} \qquad\qquad W_{dev_{(e)}}^{(i)} = \sum_{k=1}^{n_{ipts}} W_{dev_{(e,k)}}^{(i)} \\ W_{dev_{(e,k)}}^{(i)} &= \frac{1}{2}\left(\{\mathbf{S}\}_{(e,k)}^{(i)} + \{\mathbf{S}\}_{(e,k)}^{(i-1)}\right)^T [\boldsymbol{\Theta}] \left(\{\boldsymbol{\varepsilon}\}_{(e,k)}^{(i)} - \{\boldsymbol{\varepsilon}\}_{(e,k)}^{(i-1)}\right) w_{(e,k)} \end{aligned} \tag{7.25}$$

$[\boldsymbol{\Theta}]$ is a deviatoric operator, defined as Eq. (7.26).

$$[\boldsymbol{\Theta}] = [\mathbf{I}] - \frac{1}{3}\{\mathbf{A}\}^T\{\mathbf{A}\} \tag{7.26}$$

$\{\mathbf{A}\} = \begin{Bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \end{Bmatrix}$ is the trace operator on vector form.
The contents of $\{\mathbf{R}\}^{(i)}$, $\{\mathbf{H}\}^{(i)}$, $\{\mathbf{U}\}^{(i)}$ and $\{\mathbf{c}\}^{(i)}$ have now been defined along with the objective functions, thus the work of determining the individual partial derivatives described in Sec. 7.1 can be performed.

7.3 Filter derivatives

As mentioned previously, the structural analysis is performed using the projection filtered variables, $\{\tilde{\tilde{\mathbf{x}}}\}$. However, the derivative needed for optimization is with respect to the unfiltered PDs $\{\mathbf{x}\}$. Thus, when evaluating $\frac{d\hat{f}}{d\{\mathbf{x}\}}$, a chain rule has to be employed, which requires summation over the elements within the filter radius. The sensitivity for the e^{th} element is given in Eq. (7.27).

$$\frac{d\hat{f}}{dx_{(e)}} = \sum_{l\in N_{(e)}} \frac{d\hat{f}}{d\tilde{\tilde{x}}_{(l)}} \frac{\partial \tilde{\tilde{x}}_{(l)}}{\partial \tilde{x}_{(l)}} \frac{\partial \tilde{x}_{(l)}}{\partial x_{(e)}} \tag{7.27}$$

$N_{(e)}$ is the neighboring elements of the e^{th} element, including the element itself, as explained in Sec. 6.3. This chain rule is likewise needed for the evaluation of the sensitivities of the constraint equations. The filtering equations are given in Eq. (6.11) and Eq. (6.13), but are restated here for clarity.

$$\tilde{x}_{(j)} = \frac{\sum_{p\in N_{(j)}} G^{(j,p)} x_{(p)}}{\sum_{p\in N_{(j)}} G^{(j,p)}} \qquad\qquad \tilde{\tilde{x}}_{(j)} = \frac{\tanh(\beta\eta) + \tanh(\beta(\tilde{x}_{(j)} - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1-\eta))}$$

The derivative of the threshold-projection filter is given in Eq. (7.28).

$$\frac{\partial \tilde{\tilde{x}}_{(j)}}{\partial \tilde{x}_{(j)}} = \frac{\beta \text{sech}(\beta\tilde{x}_{(j)} - \eta)^2}{\tanh(\beta\eta) + \tanh\left(\beta - \beta\eta\right)} \tag{7.28}$$

The derivative of the linear density filter is given in Eq. (7.29).

$$\frac{\partial \tilde{x}_{(j)}}{\partial x_{(e)}} = \frac{G^{(j,e)}}{\sum_{p\in N_{(j)}} G^{(j,p)}} \tag{7.29}$$

Thus, the derivative needed for the chain rules of differentiation is derived. The following section will outline how to evaluate $\frac{d\hat{f}}{d\tilde{\tilde{x}}_{(j)}}$, thus, only the derivatives w.r.t. filtered PDs are going to be derived in following sections and it is implied that the above chain-rule has to be applied subsequently to obtain $\frac{d\hat{f}}{dx_{(e)}}$.

7.4 Residual derivatives

The purpose of this section is to state the partial derivatives of the structural residual vectors w.r.t. the structural state variables. Firstly, the derivatives of the global residual $\{\mathbf{R}\}$ w.r.t. the global- and local variables, $\{\mathbf{U}\}$ and $\{\mathbf{c}\}$, are derived. Hereafter, the derivatives of the local residual $\{\mathbf{H}\}$ are derived.
The global and local residuals in Eq. (7.1) are for the e^{th} element, $\{\mathbf{r}\}^{(i)}_{(e)}$ and $\{\mathbf{h}\}^{(i)}_{(e)}$, only dependent on their corresponding element global and local variables $\{\mathbf{a}\}_{(e)}$ and $\{\mathbf{c}\}_{(e,k)}$, from Eqs. (7.2) to (7.4), such that $\{\mathbf{r}\}^{(i)}_{(1)}$ and $\{\mathbf{h}\}^{(i)}_{(1)}$ are only dependent on elemental values from $\{\mathbf{a}\}_{(1)}$ and $\{\mathbf{c}\}_{(1,k)}$. Thus when taking the derivatives of $\{\mathbf{R}\}^{(i)}$ and $\{\mathbf{H}\}^{(i)}$ w.r.t. the global and local variables, there is only a contribution from the global and local variables from the same e^{th} element.

Starting with the derivative of the global residual $\{\mathbf{R}\}^{(i)}$ from Eq. (7.13), w.r.t. the global variables $\{\mathbf{U}\}$ from Eq. (7.2). Since there is no explicit dependency on the previous element nodal displacements, $\{\mathbf{a}\}^{(i-1)}_{(e)}$, this derivative is zero. Since the stress is a local variable, it is kept constant during differentiation w.r.t. $\{\mathbf{a}\}_{(e)}$. The derivative w.r.t. the current element nodal displacements, $\{\mathbf{a}\}^{(i)}_{(e)}$, can therefore be shown to be the stress stiffening contribution to the element tangent stiffness, as given in Eq. (7.30). For a derivation of Eq. (7.30), see Appendix E.2.

$$\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i-1)}} = [\mathbf{0}] \qquad \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}} = \overset{n_{elm}}{\underset{e=1}{\mathcal{A}}} \frac{\partial\{\mathbf{r}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}$$
$$\frac{\partial\{\mathbf{r}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} = \sum_{k=1}^{n_{ipts}} v_{(e)} \left([\mathbf{G}]^{(i)}_{(e,k)}\right)^T \left[\hat{\mathbf{S}}\right]^{(i)}_{(e,k)} [\mathbf{G}]^{(i)}_{(e,k)}\, w_{(e,k)} \tag{7.30}$$

Next, taking the derivatives of the global residual $\{\mathbf{R}\}^{(i)}$ from Eq. (7.13), w.r.t. the local variables $\{\mathbf{c}\}$ from Eq. (7.21). Since there is no explicit dependency on the previous element local variables, $\{\mathbf{c}\}^{(i-1)}_{(e)}$, this derivative is zero. The residual is only a function of the local stress variable. Due to the summation over the integration point, as defined in Eq. (7.16), the derivative is a block-diagonal matrix, as seen in Eq. (7.31).

$$\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}} = [\mathbf{0}] \qquad \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}} = \overset{n_{elm}}{\underset{e=1}{\mathcal{A}}} \frac{\partial\{\mathbf{r}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}}$$
$$\frac{\partial\{\mathbf{r}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} = \overset{n_{ipts}}{\underset{k=1}{\mathcal{A}}} \left[v_{(e)}\, [\mathbf{B_{NL}}]^{(i)}_{(e,k)}\, w_{(e,k)} \quad [\mathbf{0}]\right] \tag{7.31}$$

Next, taking the derivative of $\{\mathbf{h}\}^{(i)}_{(e)}$ from Eqs. (7.18) to (7.20) w.r.t. the global variables $\{\mathbf{U}\}$ from Eq. (7.2). Only the local residuals relating to the strain decomposition, $\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}$ from Eq. (7.19), in each integration point of the e^{th} element are explicit functions of these displacements, through the strain increment, $\{\boldsymbol{\Delta\varepsilon}\}^{(i-1)}_{(e,k)}$. Thus, the derivative w.r.t. the current nodal displacement, $\{\mathbf{U}\}^{(i)}$ is given as in Eq. (7.32). For a derivation of Eq. (7.32), see Appendix E.1.

$$\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} \qquad \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} = \mathop{\mathcal{A}}_{k=1}^{n_{ipts}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}$$
$$\frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} = \begin{bmatrix} -\left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)} \\ [\mathbf{0}] \end{bmatrix} \tag{7.32}$$

The derivative w.r.t. the previous nodal displacements $\{\mathbf{U}\}^{(i-1)}$ is similarly defined, as in Eq. (7.33) but with opposite sign and evaluated for the previous strain, $\{\boldsymbol{\varepsilon}\}^{(i-1)}_{(e,k)}$.

$$\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i-1)}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i-1)}_{(e)}} \qquad \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i-1)}_{(e)}} = \mathop{\mathcal{A}}_{k=1}^{n_{ipts}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i-1)}_{(e)}}$$
$$\frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i-1)}_{(e)}} = \begin{bmatrix} \left[\mathbf{B_{NL}}\right]^{(i-1)}_{(e,k)} \\ [\mathbf{0}] \end{bmatrix} \tag{7.33}$$

The derivative of the local residual for the e^{th} element, $\{\mathbf{h}\}^{(i)}_{(e)}$ from Eqs. (7.18) to (7.20), w.r.t. the local variables $\{\mathbf{c}\}^{(i)}$ is only non-zero for the corresponding element local variables, $\{\mathbf{c}\}^{(i)}_{(e)}$. This simplifies the derivative since all the integration points are uncoupled. Thus, the derivative of $\{\mathbf{h}\}^{(i)}_{(e)}$ w.r.t. the current local variables, $\{\mathbf{c}\}^{(i)}$ from Eq. (7.21), is given as in Eq. (7.34),

$$\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} \qquad \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} = \mathop{\mathcal{A}}_{k=1}^{n_{ipts}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}}$$
$$\frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} = \begin{bmatrix} [\mathbf{D}]^{-1} + \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}_{(e,k)} & \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}_{(e,k)} \\ -\Delta t\left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}_{(e,k)} & [\mathbf{I}] - \Delta t\left(\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}_{(e,k)} \end{bmatrix} \tag{7.34}$$

which is derived in Sec. 5.1, since only the increments are functions of the current values. The structure of the derivative w.r.t. the previous local variables, $\{\mathbf{c}\}^{(i-1)}$, has an identical structure, as given in Eq. (7.35).

$$\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i-1)}_{(e)}} \qquad \frac{\partial\{\mathbf{h}\}^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i-1)}_{(e)}} = \mathop{\mathcal{A}}_{k=1}^{n_{ipts}} \frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i-1)}_{(e,k)}}$$
$$\frac{\partial\{\mathbf{h}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i-1)}_{(e,k)}} = \begin{bmatrix} \left(\frac{\partial\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}_{(e,k)} & \left(\frac{\partial\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}_{(e,k)} \\ \left(\frac{\partial\{\mathbf{h}_{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}}\right)^{(i-1)}_{(e,k)} & \left(\frac{\partial\{\mathbf{h}_{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}}\right)^{(i-1)}_{(e,k)} \end{bmatrix} \tag{7.35}$$

Due to the partial derivatives being evaluated at the previously converged time-step, each term in Eq. (7.35) necessitates the evaluation of the second-order derivatives as given in

Eq. (7.36).

$$\begin{aligned}
\frac{\partial\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}}{\partial\{\mathbf{S}\}} &= \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}^2}\{\boldsymbol{\Delta}\mathbf{S}\} + \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}\partial\{\mathbf{S}\}}\{\boldsymbol{\Delta\kappa}\} + \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T\partial\{\mathbf{S}\}}\Delta T - [\mathbf{D}]^{-1}\\
\frac{\partial\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}}{\partial\{\boldsymbol{\kappa}\}} &= \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\mathbf{S}\}\partial\{\boldsymbol{\kappa}\}}\{\boldsymbol{\Delta}\mathbf{S}\} + \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial\{\boldsymbol{\kappa}\}^2}\{\boldsymbol{\Delta\kappa}\} + \Delta t\frac{\partial^2\{\dot{\boldsymbol{\varepsilon}}^{\mathbf{vp}}\}}{\partial T\partial\{\boldsymbol{\kappa}\}}\Delta T\\
\frac{\partial\{\mathbf{h}_{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}} &= -\Delta t\frac{\partial^2\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}^2}\{\boldsymbol{\Delta}\mathbf{S}\} - \Delta t\frac{\partial\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}\partial\{\mathbf{S}\}}\{\boldsymbol{\Delta\kappa}\} - \Delta t\frac{\partial^2\{\dot{\boldsymbol{\kappa}}\}}{\partial T\partial\{\mathbf{S}\}}\Delta T\\
\frac{\partial\{\mathbf{h}_{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}} &= -\Delta t\frac{\partial^2\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\mathbf{S}\}\partial\{\boldsymbol{\kappa}\}}\{\boldsymbol{\Delta}\mathbf{S}\} - \Delta t\frac{\partial^2\{\dot{\boldsymbol{\kappa}}\}}{\partial\{\boldsymbol{\kappa}\}^2}\{\boldsymbol{\Delta\kappa}\} - \Delta t\frac{\partial^2\{\dot{\boldsymbol{\kappa}}\}}{\partial T\partial\{\boldsymbol{\kappa}\}}\Delta T - [\mathbf{I}]
\end{aligned} \tag{7.36}$$

The indices relating to the element, integration point and time-step have been omitted for readability. The shorthand derivative notation used in Eq. (7.36) is simply for brevity and the explicit form of each of the terms in Eq. (7.36) are given in Appendix A. Further, It can be seen that the DSA of the implicit constraints also results in the same form of the consistent tangent stiffness as derived in, e.g., Vidal et al. [1991], see Eq. (7.37). This means that the tangent stiffness can equally be described by:

$$[\mathbf{K_t}]^{(i)} = \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}} - \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\left(\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\right)^{-1}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i)}} \tag{7.37}$$

By inserting Eq. (7.30), Eq. (7.31), Eq. (7.32) and Eq. (7.34) in Eq. (7.37) it can be shown to be equivalent to the tangent stiffness defined in Chaps. 4 to 5, thereby substantiating the derived terms.

The derivative of the current global residual $\{\mathbf{R}\}^{(i)}$ with respect to the filtered PDs $\{\tilde{\tilde{\mathbf{x}}}\}$ is given in Eq. (7.38). The derivative of the global residual for the e^{th} element, $\{\mathbf{r}\}_{(e)}^{(i)}$, is only nonzero w.r.t. the corresponding filtered element PD, $\tilde{\tilde{x}}_{(e)}$. Due to the influence of gravity, both the external and current internal forces, $\{\mathbf{f_{ext}}\}_{(e,k)}$ and $\{\mathbf{f_{int}}\}_{(e,k)}^{(i)}$ respectively, are explicit functions of the filtered element PD $\tilde{\tilde{x}}_{(e)}$.

$$\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{r}\}_{(e)}^{(i)}}{\partial\tilde{\tilde{x}}_{(e)}} \tag{7.38}$$

$$\frac{\partial\{\mathbf{r}\}_{(e)}^{(i)}}{\partial\tilde{\tilde{x}}_{(e)}} = \sum_{k=1}^{n_{ipts}}\left(\frac{\partial\{\mathbf{f_{int}}\}_{(e,k)}^{(i)}}{\partial\tilde{\tilde{x}}_{(e)}} - \frac{\partial\{\mathbf{f_{ext}}\}_{(e,k)}}{\partial\tilde{\tilde{x}}_{(e)}}\right) \tag{7.39}$$

$$\frac{\partial\{\mathbf{f_{int}}\}_{(e,k)}^{(i)}}{\partial\tilde{\tilde{x}}_{(e)}} = \frac{\partial v_{(e)}}{\partial\tilde{\tilde{x}}_{(e)}}\left([\mathbf{B_{NL}}]_{(e,k)}^{(i)}\right)^T\{\mathbf{S}\}_{(e,k)}^{(i)}w_{(e,k)} \tag{7.40}$$

$$\frac{\partial\{\mathbf{f_{ext}}\}_{(e,k)}}{\partial\tilde{\tilde{x}}_{(e)}} = d_0\rho\,[\mathbf{N}]_{(e,k)}^T\,\{\mathbf{g}\}w_{(e,k)} \tag{7.41}$$

Since the local residuals $\{\mathbf{H}\}$ are not penalized by the filtered PDs, the derivative with respect to these is zero.

$$\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} = [\mathbf{0}] \tag{7.42}$$

7.4.1 Objective derivatives

The derivatives of the objective functions described in Sec. 6.9 are derived in the following section. When deriving the sensitivities of the objective functions, the same approach is utilized independent of the chosen objective. As was shown for the implicit constraints, the derivatives w.r.t. the global variables, local variables and PDs have to be derived for each of the proposed objective functions. Thus it is relatively straightforward to change or include new objective functions.

Squared displacements

The derivative of the objective function, as described in Eq. (7.25), with respect to the global variables, is given in Eq. (7.43).

$$\frac{\partial f_1}{\partial\{\mathbf{U}\}^{(i)}} = \begin{cases} \left(\{\mathbf{U}\}^{(n)}\right)^T, & \text{if } i = n \\ \{\mathbf{0}\}^T, & \text{if } i \neq n \end{cases} \tag{7.43}$$

The derivative of the objective function with respect to the local variables is given in Eq. (7.44).

$$\frac{\partial f_1}{\partial\{\mathbf{c}\}^{(i)}} = \{\mathbf{0}\}^T \tag{7.44}$$

There is no explicit dependency of the objective function on the design variables thus this derivative is zero.

Compliance

The derivative of the objective function, as described in Eq. (7.25), with respect to the global variables, is given in Eq. (7.45).

$$\frac{\partial f_2}{\partial\{\mathbf{U}\}^{(i)}} = \begin{cases} \{\mathbf{F_{ext}}\}^T, & \text{if } i = n \\ \{\mathbf{0}\}^T, & \text{if } i \neq n \end{cases} \tag{7.45}$$

The derivative of the objective function with respect to the local variables is given in Eq. (7.46).

$$\frac{\partial f_2}{\partial\{\mathbf{c}\}^{(i)}} = \{\mathbf{0}\}^T \tag{7.46}$$

Due to the external force $\{\mathbf{F_{ext}}\}$ being an explicit function of the filtered PDs $\{\tilde{\tilde{\mathbf{x}}}\}$, cf Eq. (7.15), there is an explicit term of the sensitivities, given in Eq. (7.47).

$$\begin{aligned} \frac{\partial f_2}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} &= \left(\{\mathbf{U}\}^{(n)}\right)^T \frac{\partial\{\mathbf{F_{ext}}\}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} \qquad \frac{\partial\{\mathbf{F_{ext}}\}}{\partial\{\tilde{\tilde{\mathbf{x}}}\}} = \mathop{\mathcal{A}}_{e=1}^{n_{elm}} \frac{\partial\{\mathbf{f_{ext}}\}_{(e)}}{\partial\tilde{\tilde{x}}_{(e)}} \\ \frac{\partial\{\mathbf{f_{ext}}\}_{(e)}}{\partial\tilde{\tilde{x}}_{(e)}} &= \sum_{k=1}^{n_{ipts}} d_0\rho\,[\mathbf{N}]_{(e,k)}^T\,\{\mathbf{g}\}w_{(e,k)} \end{aligned} \tag{7.47}$$

Strain energy

The derivative of the objective function, as described in Eq. (7.25), with respect to the global variables, is slightly more involved than the former objective functions, due to the summation of the incremental strain energy over the time history. The incremental strain energy in the current time-step (i) and the subsequent time-step $(i+1)$ is a function of the current displacement, as given in Eq. (7.48).

$$\frac{\partial f_3}{\partial\{\mathbf{U}\}^{(i)}} = \begin{cases} \frac{\partial W^{(i)}}{\partial\{\mathbf{U}\}^{(i)}} + \frac{\partial W^{(i+1)}}{\partial\{\mathbf{U}\}^{(i)}} = \underset{e=1}{\overset{n_{elm}}{\mathcal{A}}} \frac{\partial W^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} + \frac{\partial W^{(i+1)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} & \text{, if } i \neq n \\ \frac{\partial W^{(n)}}{\partial\{\mathbf{U}\}^{(n)}} = \underset{e=1}{\overset{n_{elm}}{\mathcal{A}}} \frac{\partial W^{(n)}_{(e)}}{\partial\{\mathbf{a}\}^{(n)}_{(e)}} & \text{, if } i = n \end{cases} \tag{7.48}$$

The individual terms are given by Eq. (7.49).

$$\begin{aligned} \frac{\partial W^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} \frac{1}{2}\left(\{\mathbf{S}\}^{(i)}_{(e,k)} + \{\mathbf{S}\}^{(i-1)}_{(e,k)}\right)^T [\mathbf{B_{NL}}]^{(i)}_{(e,k)}\, w_{(e,k)} \\ \frac{\partial W^{(i+1)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} -\frac{1}{2}\left(\{\mathbf{S}\}^{(i+1)}_{(e,k)} + \{\mathbf{S}\}^{(i)}_{(e,k)}\right)^T [\mathbf{B_{NL}}]^{(i)}_{(e,k)}\, w_{(e,k)} \\ \frac{\partial W^{(n)}_{(e)}}{\partial\{\mathbf{a}\}^{(n)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} \frac{1}{2}\left(\{\mathbf{S}\}^{(n)}_{(e,k)} + \{\mathbf{S}\}^{(n-1)}_{(e,k)}\right)^T [\mathbf{B_{NL}}]^{(n)}_{(e,k)}\, w_{(e,k)} \end{aligned} \tag{7.49}$$

The derivative of the objective function with respect to the local variables is similarly given in Eq. (7.50).

$$\frac{\partial f_3}{\partial\{\mathbf{c}\}^{(i)}} = \begin{cases} \frac{\partial W^{(i)}}{\partial\{\mathbf{c}\}^{(i)}} + \frac{\partial W^{(i+1)}}{\partial\{\mathbf{c}\}^{(i)}} = \underset{e=1}{\overset{n_{elm}}{\mathcal{A}}} \frac{\partial W^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} + \frac{\partial W^{(i+1)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} & \text{, if } i \neq n \\ \frac{\partial W^{(n)}}{\partial\{\mathbf{c}\}^{(n)}} = \underset{e=1}{\overset{n_{elm}}{\mathcal{A}}} \frac{\partial W^{(n)}_{(e)}}{\partial\{\mathbf{c}\}^{(n)}_{(e)}} & \text{, if } i = n \end{cases} \tag{7.50}$$

$$\begin{aligned} \frac{\partial W^{(i)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} &= \underset{k=1}{\overset{n_{ipts}}{\mathcal{A}}} \frac{\partial W^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} & \frac{\partial W^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i-1)}_{(e,k)}\right)^T w_{(e,k)} \quad \{\mathbf{0}\}\right\} \\ \frac{\partial W^{(i+1)}_{(e)}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} &= \underset{k=1}{\overset{n_{ipts}}{\mathcal{A}}} \frac{\partial W^{(i+1)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} & \frac{\partial W^{(i+1)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(i+1)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)}\right)^T w_{(e,k)} \quad \{\mathbf{0}\}\right\} \\ \frac{\partial W^{(n)}_{(e)}}{\partial\{\mathbf{c}\}^{(n)}_{(e)}} &= \underset{k=1}{\overset{n_{ipts}}{\mathcal{A}}} \frac{\partial W^{(n)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(n)}_{(e,k)}} & \frac{\partial W^{(n)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(n)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(n)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(n-1)}_{(e,k)}\right)^T w_{(e,k)} \quad \{\mathbf{0}\}\right\} \end{aligned} \tag{7.51}$$

There is no explicit dependency of the objective function on the design variables thus this derivative is zero.

Deviatoric Work

The derivative of the deviatoric work is derived analogously to the strain energy, as given in Eq. (7.52).

$$\frac{\partial f_4}{\partial\{\mathbf{U}\}^{(i)}} = \begin{cases} \frac{\partial W^{(i)}_{dev}}{\partial\{\mathbf{U}\}^{(i)}} + \frac{\partial W^{(i+1)}_{dev}}{\partial\{\mathbf{U}\}^{(i)}} = \mathop{\mathcal{A}}\limits_{e=1}^{n_{elm}} \frac{\partial W^{(i)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} + \frac{\partial W^{(i+1)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} & \text{, if } i \neq n \\ \frac{\partial W^{(n)}_{dev}}{\partial\{\mathbf{U}\}^{(n)}} = \mathop{\mathcal{A}}\limits_{e=1}^{n_{elm}} \frac{\partial W^{(n)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(n)}_{(e)}} & \text{, if } i = n \end{cases} \tag{7.52}$$

$$\begin{aligned} \frac{\partial W^{(i)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} \frac{1}{2}\left(\{\mathbf{S}\}^{(i)}_{(e,k)} + \{\mathbf{S}\}^{(i-1)}_{(e,k)}\right)^T [\mathbf{\Theta}]\,[\mathbf{B_{NL}}]^{(i)}_{(e,k)}\, w_{(e,k)} \\ \frac{\partial W^{(i+1)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} -\frac{1}{2}\left(\{\mathbf{S}\}^{(i+1)}_{(e,k)} + \{\mathbf{S}\}^{(i)}_{(e,k)}\right)^T [\mathbf{\Theta}]\,[\mathbf{B_{NL}}]^{(i)}_{(e,k)}\, w_{(e,k)} \\ \frac{\partial W^{(n)}_{dev_{(e)}}}{\partial\{\mathbf{a}\}^{(n)}_{(e)}} &= \sum_{k=1}^{n_{ipts}} \frac{1}{2}\left(\{\mathbf{S}\}^{(n)}_{(e,k)} + \{\mathbf{S}\}^{(n-1)}_{(e,k)}\right)^T [\mathbf{\Theta}]\,[\mathbf{B_{NL}}]^{(n)}_{(e,k)}\, w_{(e,k)} \end{aligned} \tag{7.53}$$

The derivative of the objective function with respect to the local variables is given in Eq. (7.54).

$$\frac{\partial f_4}{\partial\{\mathbf{c}\}^{(i)}} = \begin{cases} \frac{\partial W^{(i)}_{dev}}{\partial\{\mathbf{c}\}^{(i)}} + \frac{\partial W^{(i+1)}_{dev}}{\partial\{\mathbf{c}\}^{(i)}} = \mathop{\mathcal{A}}\limits_{e=1}^{n_{elm}} \frac{\partial W^{(i)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} + \frac{\partial W^{(i+1)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} & \text{, if } i \neq n \\ \frac{\partial W^{(n)}_{dev}}{\partial\{\mathbf{c}\}^{(n)}} = \mathop{\mathcal{A}}\limits_{e=1}^{n_{elm}} \frac{\partial W^{(n)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(n)}_{(e)}} & \text{, if } i = n \end{cases} \tag{7.54}$$

$$\begin{aligned} \frac{\partial W^{(i)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} &= \mathop{\mathcal{A}}\limits_{k=1}^{n_{ipts}} \frac{\partial W^{(i)}_{(e,k)}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} & \frac{\partial W^{(i)}_{dev_{(e,k)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i-1)}_{(e,k)}\right)^T [\mathbf{\Theta}]\, w_{(e,k)} \quad \{\mathbf{0}\}\right\} \\ \frac{\partial W^{(i+1)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e)}} &= \mathop{\mathcal{A}}\limits_{k=1}^{n_{ipts}} \frac{\partial W^{(i+1)}_{dev_{(e,k)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} & \frac{\partial W^{(i+1)}_{dev_{(e,k)}}}{\partial\{\mathbf{c}\}^{(i)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(i+1)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)}\right)^T [\mathbf{\Theta}]\, w_{(e,k)} \quad \{\mathbf{0}\}\right\} \\ \frac{\partial W^{(n)}_{dev_{(e)}}}{\partial\{\mathbf{c}\}^{(n)}_{(e)}} &= \mathop{\mathcal{A}}\limits_{k=1}^{n_{ipts}} \frac{\partial W^{(n)}_{dev_{(e,k)}}}{\partial\{\mathbf{c}\}^{(n)}_{(e,k)}} & \frac{\partial W^{(n)}_{dev_{(e,k)}}}{\partial\{\mathbf{c}\}^{(n)}_{(e,k)}} &= \left\{\tfrac{1}{2}\left(\{\boldsymbol{\varepsilon}\}^{(n)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(n-1)}_{(e,k)}\right)^T [\mathbf{\Theta}]\, w_{(e,k)} \quad \{\mathbf{0}\}\right\} \end{aligned} \tag{7.55}$$

There is no explicit dependency of the objective function on the design variables thus this derivative is zero.

7.5 Constraint sensitivities

For all the constraints included in the optimization problem, their sensitivities wrt. the design variables must be determined. Only the derivatives w.r.t. the filtered design variables are derived since the derivatives of the filters have already been found in Sec. 7.3. The sensitivities of the volume constraints will be derived in the following. The upper and lower volume constraints are defined by Eq. (6.15a) and Eq. (6.15b), respectively.

Upper volume constraint

The sensitivity of the upper volume constraint wrt. the filtered design variables can be seen in Eq. (7.56)

$$\frac{\partial g_1}{\partial \tilde{\tilde{x}}_{(e)}} = \sum_{l \in N_{(e)}} \frac{\partial g_1}{\partial \tilde{\tilde{x}}_{(l)}} \tag{7.56}$$

The derivative of the upper volume constraint for the l^{th} neighboring element is given in Eq. (7.57).

$$\frac{\partial g_1}{\partial \tilde{\tilde{x}}_{(l)}} = \frac{V_{(l)}}{V_U \sum_{i=1}^{n_{elem}} V_{(i)}} \tag{7.57}$$

Lower volume constraint

The sensitivity of the upper volume constraint wrt. the filtered design variables can be seen in Eq. (7.58)

$$\frac{\partial g_2}{\partial \tilde{\tilde{x}}_{(e)}} = \sum_{l \in N_{(e)}} \frac{\partial g_2}{\partial \tilde{\tilde{x}}_{(l)}} \tag{7.58}$$

The derivative of the lower volume constraint for the l^{th} neighboring element is given in Eq. (7.59).

$$\frac{\partial g_2}{\partial \tilde{\tilde{x}}_{(l)}} = -\frac{V_{(l)}}{V_L \sum_{i=1}^{n_{elem}} V_{(i)}} \tag{7.59}$$

The next step is to verify the derived objective and constraint sensitivities before topology optimization can be performed.

7.6 Verification

The resulting sensitivity, $\frac{df}{d\{\mathbf{x}\}}$ is compared to the forward difference approximation, $\frac{df}{d\{\mathbf{x}\}}_{\text{fda}}$, which is calculated as;

$$\frac{df}{dx_{(e)}}_{\text{fda}} = \frac{f(x_{(e)} + \Delta x) - f(x_{(e)})}{\Delta x} + \mathcal{O}(\Delta x) \tag{7.60}$$

Here f is an arbitrary function of the PDs $\{\mathbf{x}\}$, e.g. the objective function or constraints. For each sensitivity of the variable $x_{(e)}$, all other entries in $\{\mathbf{x}\}$ are kept constant. It is then assembled into the vector of sensitivities $\frac{df}{d\{\mathbf{x}\}}_{\text{fda}}$. The error between the forward difference approximation and analytical sensitivity is calculated as:

$$\text{error} = \left\| \frac{\frac{df}{d\{\mathbf{x}\}}_{\text{fda}} - \frac{df}{d\{\mathbf{x}\}}}{\frac{df}{d\{\mathbf{x}\}}} \right\|_2 \tag{7.61}$$

The analytical and numerically approximated sensitivities are compared by plotting the error for increasingly small perturbations Δx. The error is linear in the perturbation size, i.e., the approximation is expected to become more correct as the perturbation size is reduced. At one point, however, truncation errors will dominate and the accuracy of the approximation will decrease.

The verification is performed on the same discretized model for all sensitivity verifications. An MBB-beam like Fig. 5.7 is used but with 12 × 4 × 2 elements in the x, y and z-directions to reduce computational time during gradient checks, due to the time-costly forward difference calculations.

7.6.1 Computational problems

During troubleshooting, while implementing the derived sensitivities in the MATLAB framework, a set of tests was performed on a half MBB beam, like the sketch in Fig. 5.7 but without symmetry in the x-y plane, discretized with 1800 elements. The sintering analysis was parallelized and solved with an eight-core Ryzen 7 5800X desktop CPU, the sintering analysis took around 2.5 minutes to solve. The sintering analysis solution time was constant with small variations. The solution time for the adjoint system in each optimizer iteration is shown in Fig. 7.1.

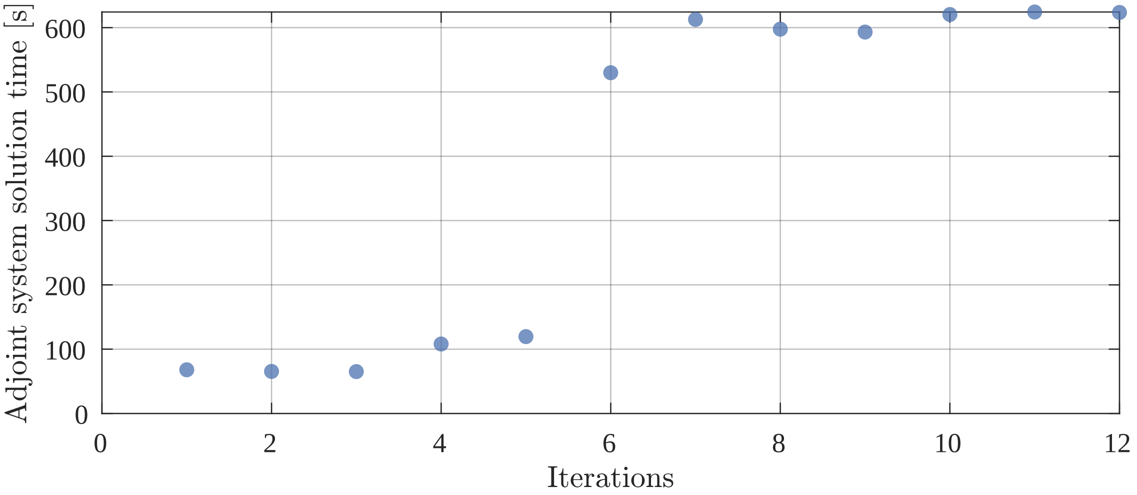


Figure 7.1. Solution time of the final time step (initial) adjoint system for the first 12 optimizer iterations, for an 8-core Ryzen 7 5800X CPU

It is evident that the solution time of the adjoint system increases dramatically even though a direct solver is used. This was hypothesized to be due to the adjoint Jacobian being ill-conditioned. In Fig. 7.2, the condition number of the adjoint Jacobian during a sintering analysis is shown.

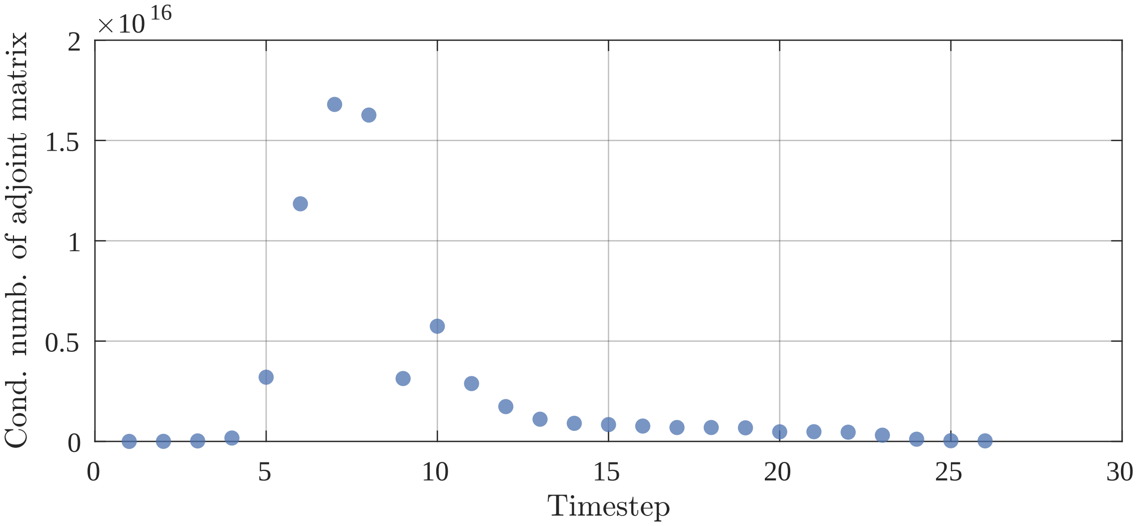


Figure 7.2. The condition number of the adjoint system matrix during solution of the adjoint system with SI units

The Jacobian is ill-conditioned and nearly singular. The peak in condition number coincides with the sintering stress activation temperature. Thus far, SI units have been used for material- and geometric properties. However, by a consistent change of units, using metric units (e.g. mm, N, ton, MPa) instead of SI, the condition number of the adjoint Jacobian was lowered significantly, see Fig. 7.3 compared to Fig. 7.2.

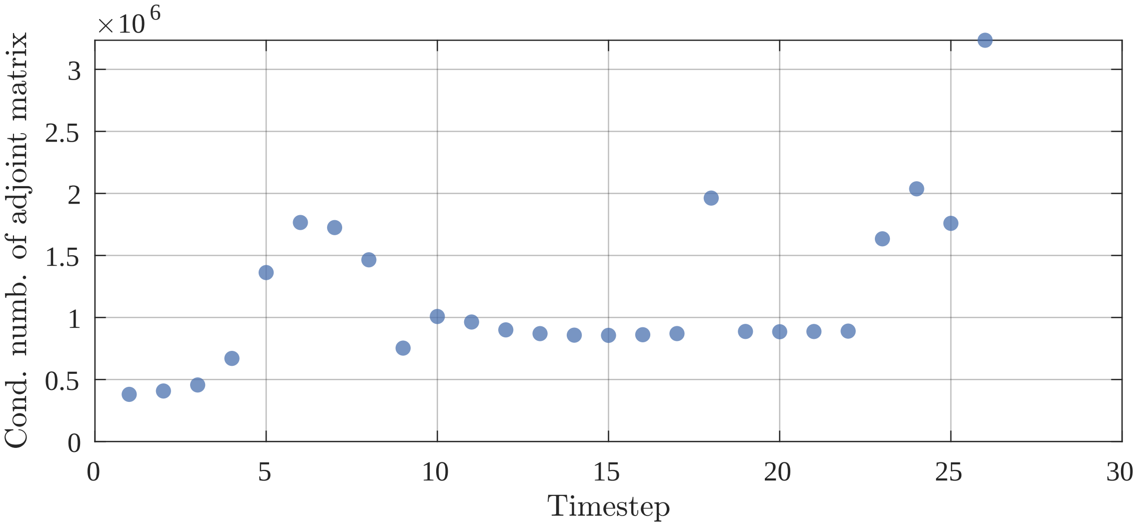


Figure 7.3. The condition number of the adjoint system matrix during solution of the adjoint system, using non-SI units

As a result, the increase in computational time ceased to exist and better numerical accuracy is to be expected. Therefore, these units are used for the remainder of the report for the algorithmic implementation only.

7.6.2 Objective

The design sensitivity formulation as derived in Sec. 7.4 is implemented and verified as described. The material properties are given previously in Tab. 3.1 and the sintering schedule in Fig. 3.2. A uniform distribution of PDs is used, which are initialized at 0.5. Since the sensitivities are derived based on the assumption that $\{\mathbf{R}\}^{(i)}$ is equal to zero, the convergence tolerance, R_{tol}, is reduced to 1e-8, for verification purposes. Two data points are calculated for each decade of perturbation size.

Squared displacements

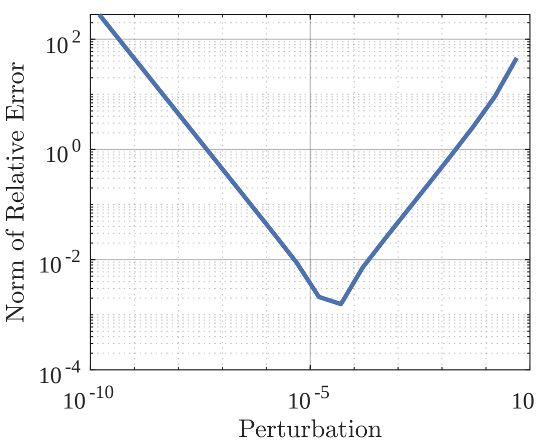


Figure 7.4. Norm of the relative difference between analytical and FDA approximated sensitivities.

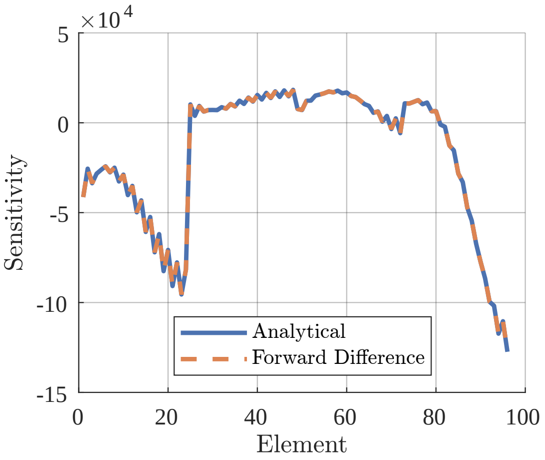


Figure 7.5. Comparison of analytical and FDA approximated sensitivities.

Compliance

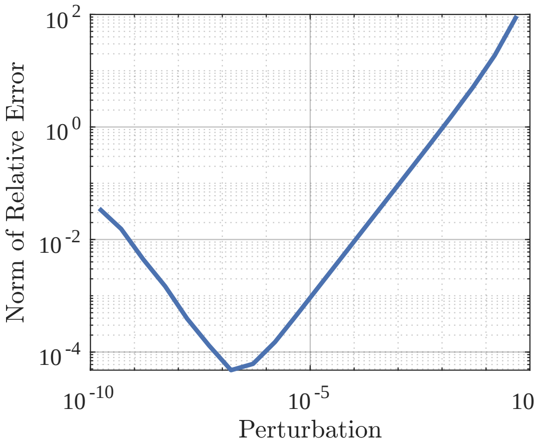


Figure 7.6. Norm of the relative difference between analytical and FDA approximated sensitivities.

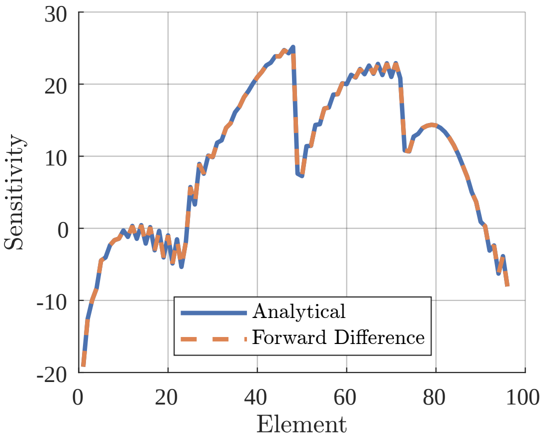


Figure 7.7. Comparison of analytical and FDA approximated sensitivities.

Strain energy

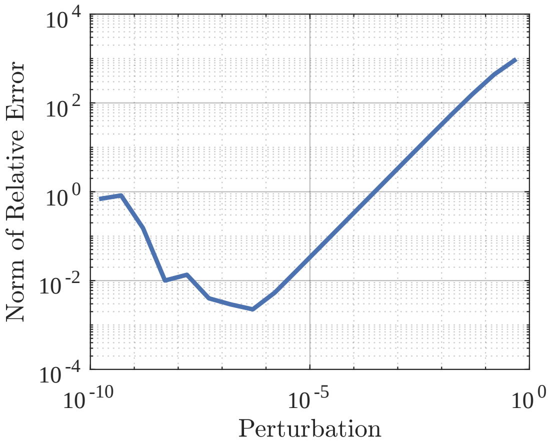


Figure 7.8. Norm of the relative difference between analytical and FDA approximated sensitivities.

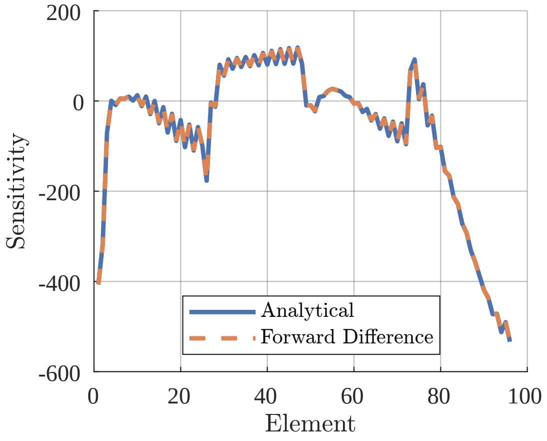


Figure 7.9. Comparison of analytical and FDA approximated sensitivities.

Deviatoric Work

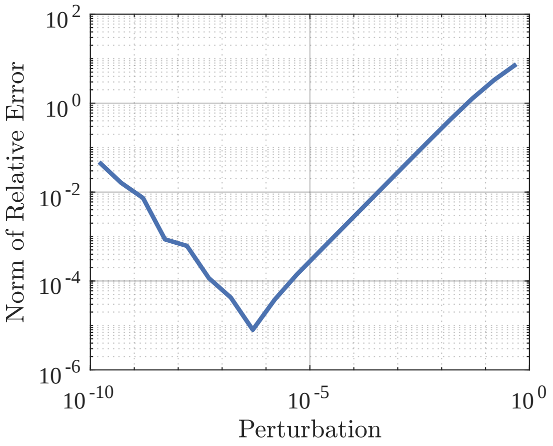


Figure 7.10. Norm of the relative difference between analytical and FDA approximated sensitivities.

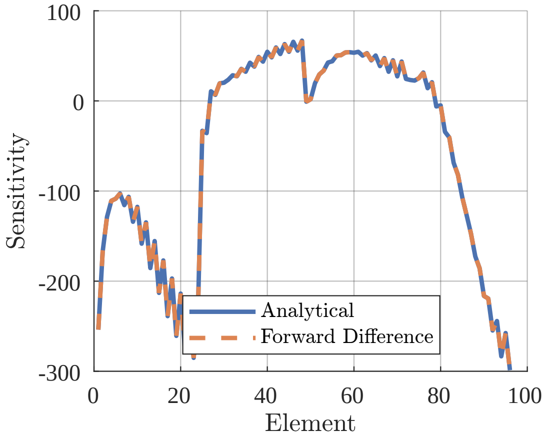


Figure 7.11. Comparison of analytical and FDA approximated sensitivities.

All step sizes show a linear reduction of the error with a reduction of the perturbation until a perturbation size in the order of 1e-6. However, the displacement objective function f_1 is an outlier, where the inflection point occurs at approximately 1e-4. In Alberdi et al. [2018] the inflection point was typically observed in a range between 1e-4 and 1e-6, however using a central difference method. Thus, the inflection points observed for these sensitivities are comparatively high. This may likely be explained by a loss of numerical accuracy, due to the solution of equation systems with poor condition numbers during the structural- and design sensitivity analysis, as has been previously shown in Fig. 5.12 and Fig. 7.3. Further, the comparison of the forward difference approximation, which results in the lowest relative error, and the analytical sensitivity shows no visual difference between them. Therefore, it is assumed that the analytical sensitivities has been implemented correctly. Next, a set of test cases is performed to show the effects of topology optimization with the different objective functions.

8 Test cases

In this section a set of test cases will be introduced and used to test the developed topology optimization procedure. Initially, the test case geometry and settings will be introduced. The topology-optimized designs will be presented, and a comparative discussion of all the results will be held in a separate section. Hereafter, an example of potential use cases will be demonstrated and discussed.

8.1 Introduction of test cases and settings

The initial test case will be the MBB beam already presented in Fig. 5.7 which is a quarter of a full MBB beam. The geometry is discretized in trilinear hexahedral elements of 5 mm × 5 mm × 5 mm. A total of 2520 elements constitutes the geometry. The material parameters from Tab. 3.1 and sinter schedule from Fig. 3.2 will be used in the topology optimization. The optimization parameters are given in Tab. 8.1.

r_{min}	β_0	$\Delta\beta$	β_{tol}	β_{max}	V_L	V_U	Δt	R_{tol}
15e-3	1	1	5e-5	40	0.5	0.6	216	1e-3

Table 8.1. Optimization parameters for test case. All are given in SI units.

All PDs are initialized to the upper volume constraint V_U. The four criteria functions to be tested are given in Eqs. (6.17) to (6.21). Additionally, no passive elements or extended domains are used due to the extensive increase in computational cost as discussed in Sec. 6.5. Continuation is performed on the projection parameter β such that when the relative change in objective value between two concurrent optimizer iterations is below a continuation tolerance, β_{tol}, then β is increased by a factor $\Delta\beta$, until a maximum value of β_{max}. The tolerance is slightly above the convergence tolerance of the optimizer. Furthermore, a different colormap than the traditional greyscale is used, such that there is a larger contrast between element faces and edges, for ease of interpretation of the 3D results. It is a subset of MATLAB's colormap *pink*. The color scale can be seen in Fig. 8.1.

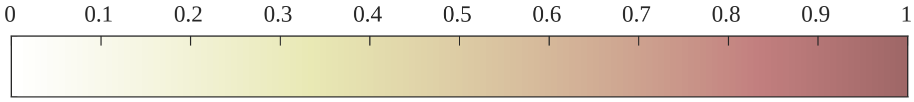


Figure 8.1. Colormap for PDs

All numerical tests are manually stopped after 400 iterations. The topology-optimized designs along with their iteration data are initially shown collectively and afterwards discussed and compared.

8.2 Numerical results

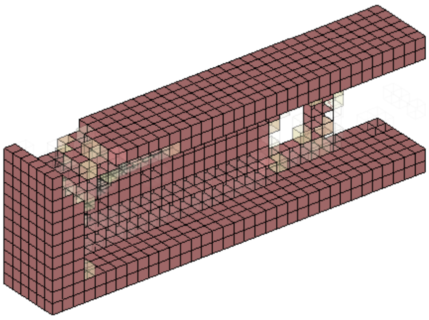


Figure 8.2. Isometric view of design after 400 iterations using f_1.

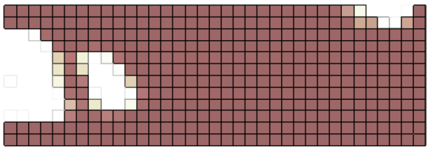


Figure 8.3. Side view of design after 400 iterations using f_1.

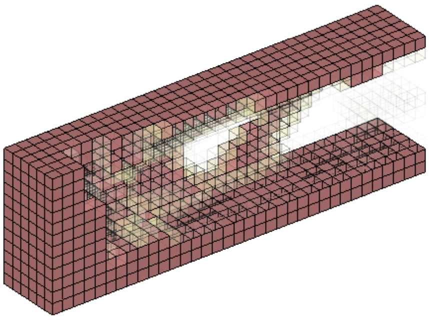


Figure 8.4. Isometric view of design after 400 iterations using f_2.

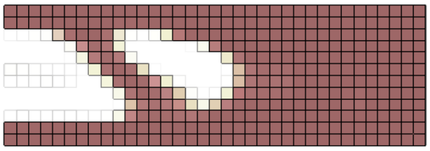


Figure 8.5. Side view of design after 400 iterations using f_2.

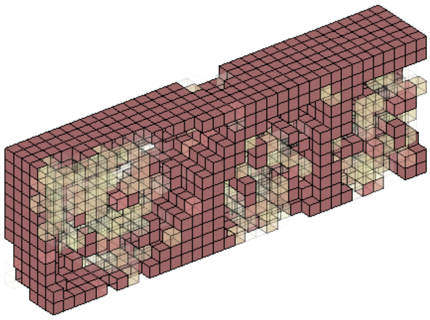


Figure 8.6. Isometric view of design after 400 iterations using f_3.

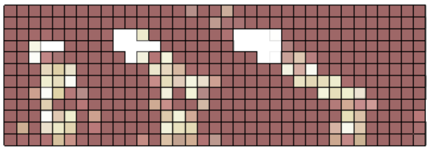


Figure 8.7. Side view of design after 400 iterations using f_3

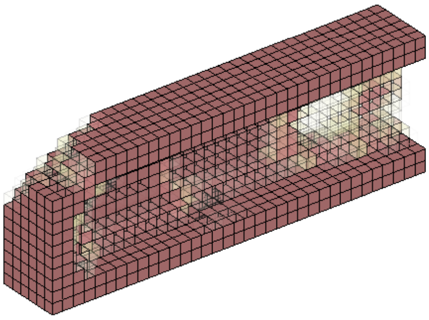


Figure 8.8. Isometric view of design after 400 iterations using f_4

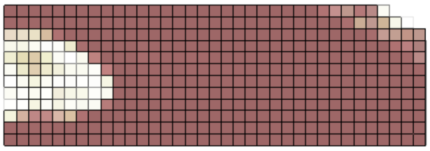


Figure 8.9. Side view of design after 400 iterations using f_4

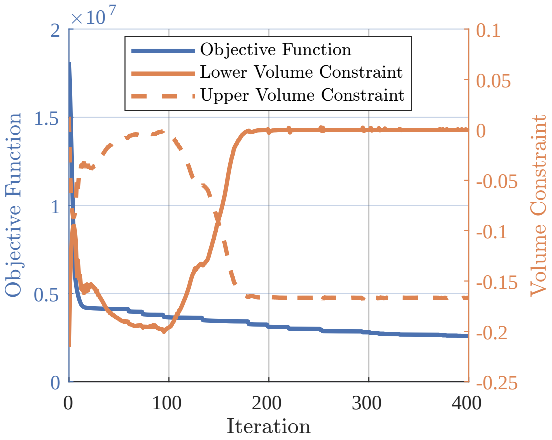


Figure 8.10. Iteration data for f_1.

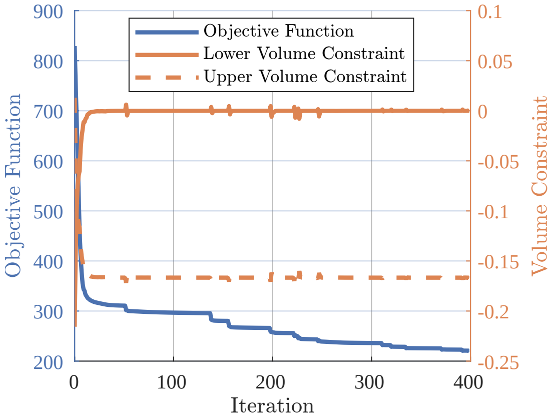


Figure 8.11. Iteration data for f_2.

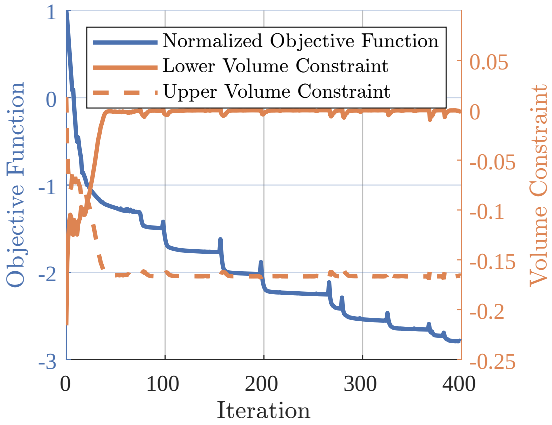


Figure 8.12. Iteration data for f_3.

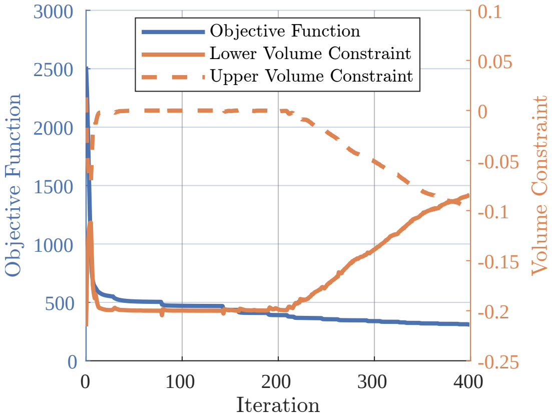


Figure 8.13. Iteration data for f_4.

8.3 Comparison

In this section, the data given in Sec. 8.2 will be discussed. Firstly, the iteration data will be discussed. A qualitative assessment of the resulting design follows hereafter. Due to the notable outlier in f_3, comparison will first be made between f_1, f_2 and f_4. Hereafter, the contrasting results of f_3 will be discussed.

In all cases, the optimization reduced the objective function by at least a factor of four. The fluctuations in the objective function and constraints are due to the continuation of the threshold projection.
For f_1 and f_2, the lower volume constraint is active after 400 iterations. f_2 quickly converges to a solution where the lower volume constraint remains active, likely meaning that further relaxation of the constraint makes a further reduction in objective function plausible. This is comparable to the results as seen for linear elastic compliance optimization, as in e.g. Bruyneel and Duysinx [2005], where the global optimum is the trivial solution of zero PDs. f_1 initially fluctuates between adding and removing material, before settling on the lower volume constraint after approximately 170 iterations.
Conversely, neither volume constraint is active for f_4 after 400 iterations. Furthermore, the upper volume constraint was active up until iteration 200. However, inspecting the change in constraint values indicates that the lower volume constraint would likely become active, given sufficient iterations.
The apparent outlier is f_3, whose objective function changes sign. This implies an unphysical negative strain energy in the structure. This is likely due to the non-associative flow of unrecoverable strain in the structure. A large part of the unrecoverable strain is in the direction of the geometry-independent sintering stress, and only a lesser part is from the mechanical stress stemming from the external load. This parasitic negative strain energy is, therefore, likely caused by design alterations that result in strains opposite of the sintering stress, thereby causing negative strain energy. This is further validated by the fact that this phenomenon is not seen for the deviatoric strain energy, f_4, where the volumetric contribution from the sintering stress is not included. Hereby, the remaining unrecoverable strain is always in the direction of mechanical stress.

The resulting design after 400 iterations using f_1 did not result in predominantly full elements on symmetry planes, as was expected. Contrarily, the full elements are concentrated on free edges, resulting in a hollow structure. Furthermore, a single reinforcing member is present, which is vertical, and a dimple is present atop the geometry towards the support. The result is somewhat reminiscent of the 2D self-weight example, as given in Fig. 6.10.
After 400 iterations using f_2, the resulting design is comparable to f_1. Both result in full elements concentrated on free edges and towards the support. A single reinforcing member is also present; however, it is inclined, and the dimple is absent.
After 400 iterations using f_4, the design is again reminiscent of the former result. Much of the mass is concentrated on free edges and towards the support. However, a slight fillet is present atop the edge. Furthermore, there is no reinforcing member.
Contrarily, using f_3 results in a structure that unfolds during deformation, causing large deformations. This further reinforces the invalidity of the objective function.

8.4 Example of application

In the former section, Sec. 8.3, the resulting design of a benchmark example was compared, using different objective formulations. Apart from the inferior objective f_3, all objective formulations may be relevant depending on the application. Therefore, in this section, an example of a practical application of the framework is presented and discussed.

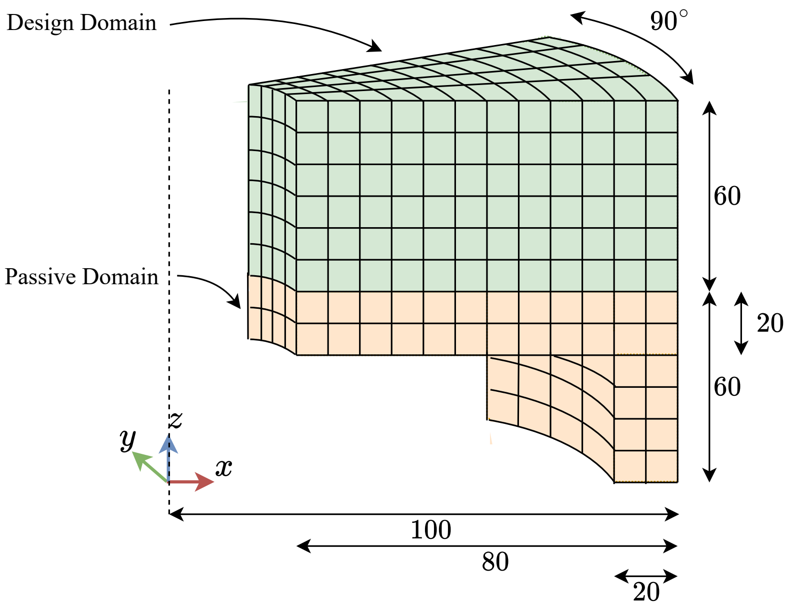


Figure 8.14. A sketch of the real-world application test geometry, dimensions in mm

The geometry for the test application is shown in Fig. 8.14. The application example involves modifying an existing geometry, defined by the passive domain, to improve manufacturability through MBJ and sintering, while ensuring it complies with its intended use. Therefore, a design domain is constructed that does not diminish or disrupt its application. Hereafter, an optimization objective may be defined to ensure that either the original or modified as-sintered geometry is optimized for its intended purpose. The resulting optimized geometry can then be compensated and printed.

8.4.1 Parameters

For this application example, the same material properties and sinter schedule used previously are given in Tab. 3.1 and Fig. 3.2, respectively.

The model is discretized using a conformal mesh with shared nodes between the passive and design domains. The passive domain is discretized into 504 elements and the design domain is discretized into 2840 elements. A high resolution in the design domain is warranted for optimization purposes. Therefore, a coarsely discretized passive domain is used.

In the passive domain, all element PDs are 1, and they are excluded from the optimization in terms of the design variables used for the optimization and the calculation of volume constraints. They are still included in the set of neighboring elements for filtering. The upper- and lower volume constraints are only calculated in the design domain. The convergence criterion is chosen to be when the relative change between two concurrent optimizer iterations is below a convergence tolerance Δf_{tol}. The optimization parameters are given in Tab. 8.2.

r_{min}	β_0	$\Delta\beta$	β_{tol}	β_{max}	V_L	V_U	Δt	R_{tol}	Δf_{tol}
10e-3	1	1	1e-3	40	0	0.7	216	1e-3	1e-5

Table 8.2. Optimization parameters for test case. All are given in SI units.

For some visualization, the segment will be replicated angularly around the axis of revolution to construct the full geometry.

8.4.2 Case 1

As shown in the introduction, a lower nominal deformation compensation results in a geometry for which a larger percentage of the tolerances were complied with. Therefore, it is essential to construct a geometry where the least amount of compensation is necessary, both in the design and passive domains. This may be achieved as shown previously, by using deviatoric work as an objective. The result is given in Figs. 8.15 to 8.22.

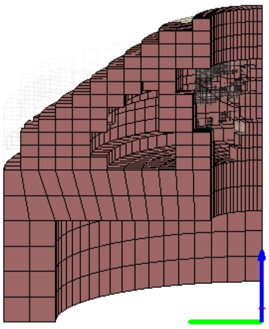


Figure 8.15. Section view along x-axis of design for case 1.

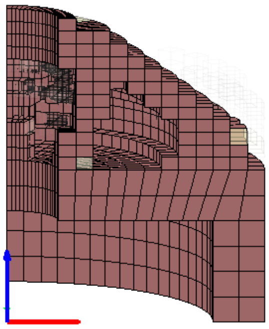


Figure 8.16. Section view along y-axis of design for case 1.

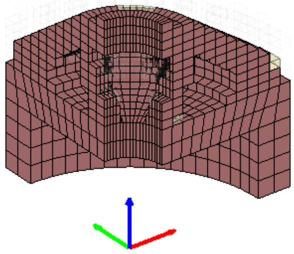


Figure 8.17. Frontal isometric view of design for case 1.

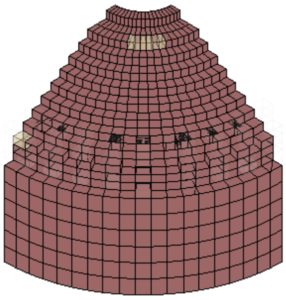


Figure 8.18. Rear isometric view of design for case 1.

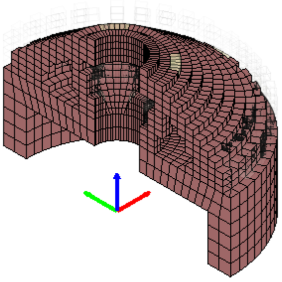


Figure 8.19. Isometric view of design flipped around XZ-symmetry plane for case 1.

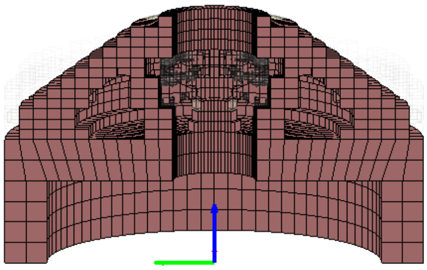


Figure 8.20. Section view of design flipped around XZ-symmetry plane for case 1.

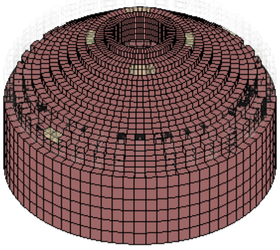


Figure 8.21. Isometric view of design flipped around both symmetry axis for case 1.

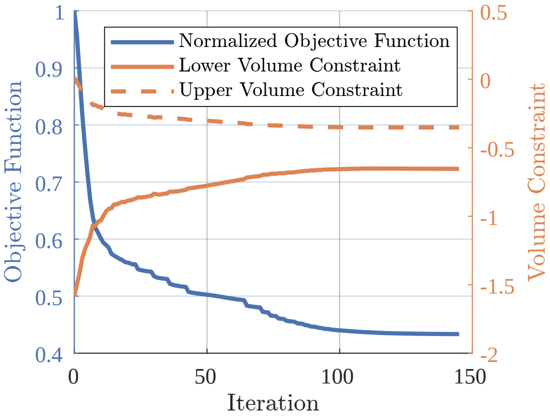


Figure 8.22. Iteration history for case 1.

Using the deviatoric work as an objective, a hollow staircase- or pyramid-like pattern emerges with four central inclined supports. A slight asymmetry can be seen. This may be due to small numerical inaccuracies, which cause an asymmetric PD distribution, which in turn causes an asymmetry in the primal analysis, such that the effect compounds over time. The iteration history can be seen in Fig. 8.22. Notably, neither volume constraint is active for the converged design. It should be investigated whether the resulting design results in a lower deformation than a design with no material in the design domain. Thus, a qualitative deformation comparison is made in Figs. 8.23 to 8.24, to a simulation where the design domain is removed entirely from the sintering analysis. The deformation is equally scaled by four in both cases to better visualize the difference. The deformation of the unmodified design is represented by an orange wireframe structure atop the modified design.

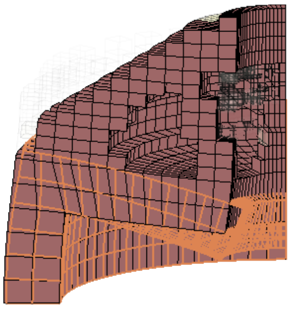


Figure 8.23. Comparison of deformation along x-axis

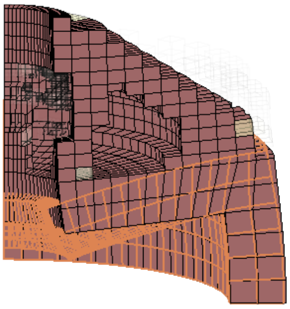


Figure 8.24. Comparison of deformation along y-axis

As can be seen, the deflection of the passive domain in the overhang is slightly reduced; however, the height decrease in the outer section is larger due to the increased gravity from the design domain.

8.4.3 Case 2

A desirable outcome during sintering would be to reach a target geometry without distortions and sagging. This would result from an idealized sintering process, where no applied loads or material or print variations are present. Therefore, it may be beneficial to simulate this idealized sintering process and use the nodal displacements as a reference deformation for f_1. Hereby, nonideal process variations such as gravity, material variations, and friction can be introduced. The optimization could then be a tool to modify the geometry to withstand these variations. In the present case, only gravity is included. The resulting design can be seen from several views, as well as the iteration history, in Figs. 8.25 to 8.32.

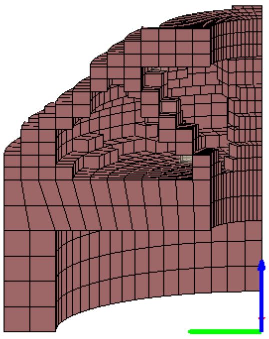


Figure 8.25. Section view along x-axis of design for case 2.

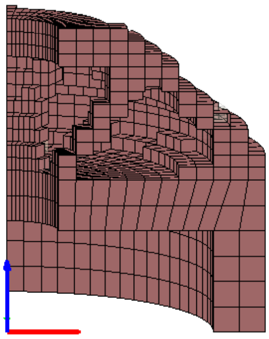


Figure 8.26. Section view along y-axis of design for case 2.

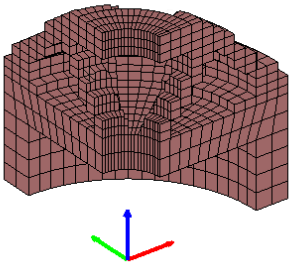


Figure 8.27. Frontal isometric view of design for case 2.

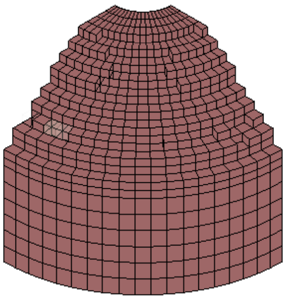


Figure 8.28. Rear isometric view of design for case 2.

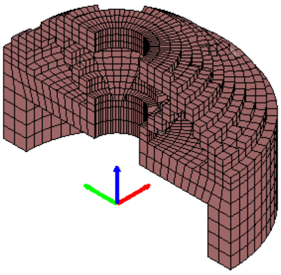


Figure 8.29. Isometric view of design flipped around XZ-symmetry plane for case 2.

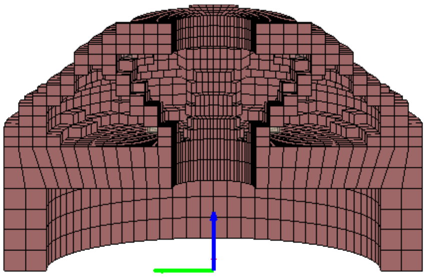


Figure 8.30. Section view of design flipped around XZ-symmetry plane for case 2.

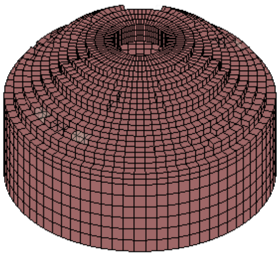


Figure 8.31. Isometric view of design flipped around both symmetry axis for case 2.

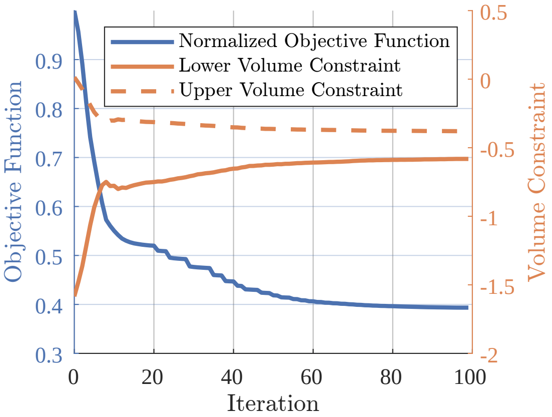


Figure 8.32. Iteration history for case 2.

Evidently, the design shows a large resemblance to the deviatoric work optimized design. Furthermore, a slight asymmetry can be observed as was also present for the deviatoric work. Likewise, neither volume constraint is active for the converged design. The deformation is compared to the target deformation to gauge how effective the modified design is. Again, all deformations are scaled by four, and the unmodified design's deformation is shown by an orange wireframe. Additionally, the target deformation is shown by a green wireframe. This is done in Figs. 8.33 to 8.34

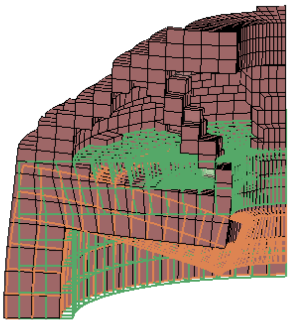


Figure 8.33. Comparison of deformation along x-axis

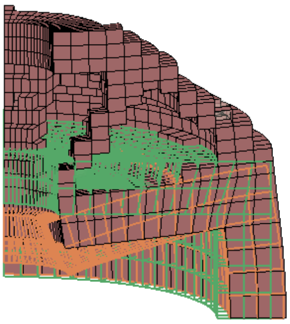


Figure 8.34. Comparison of deformation along y-axis

The result shows that the modified model better agrees with deformations from an idealized free-sintering simulation of the passive domain. Analogous to case 1, the deflection of the overhang is reduced, and the increased gravity in the design domain causes a reduction in height in the outer section. The large similarities to case 1 hint towards an idealized sintering process may correspond to a sintering process where the distortional strain energy is minimized. Thus, an optimal design for sintering may be achieved by minimizing the deviatoric work of the structure, as in case 1. This may be preferable as compared to the example in case 2, as no reference deformations have to be defined.

9 Discussion

The following is a discussion of the contents of the report and subjects that are yet to be discovered or included in the topology optimization problem but are deemed to impact either computational speed or modeling accuracy. Topics for future work are also included in the discussion below.

Line-search

The computational time of the NLFE sintering analysis is critical when performing topology optimization since a NLFEA has to be performed for each optimizer iteration. In certain cases, if too few time-steps are used, the NR-solver can diverge and cause the NLFEA solution to become invalid. To improve the convergence characteristics of the NR solver, it can be combined with an inexact line-search algorithm. Crisfield [1991] proposes to utilize tangent stiffness matrix and force residual information from the previous converged load or time step, to determine a scalar value by which the NR-determined displacement increment can be amplified. The goal is then to reduce the number of equilibrium iterations by reaching a region in solution space where the NR-solver has quadratic convergence, quicker. By reducing the number of equilibrium iterations, the stiffness matrix, internal force and material model have to be evaluated and assembled fewer times which could speed up the analysis.

Adaptive time-stepping

The implementation of numerical time integration in the sintering material model has used constant time steps throughout the NLFEA. As shown in Fig. 5.8 and Fig. 5.9 the deformation evolution during a sintering analysis is not very timestep dependent for the analyzed timesteps. Therefore, numerical stability has largely dominated the choice of time step. As shown in Fig. 5.10 and Fig. 5.11 the NR-solver generally needs fewer equilibrium iterations to converge during heating and cooling. The use of adaptive time-stepping could eliminate some redundant time steps and thus decrease the overall computation time whilst retaining analysis accuracy. Further, fewer time steps would also reduce the computational demand of the DSA since the adjoint vectors have to be computed for each converged timestep.

One approach could be to adjust the time step based on the convergence rate of the force residual as discussed in [Araújo Fernandes et al., 2017]. The proposed approach showed time-stepping and convergence properties for path-dependent problems similar to the commercial FEA program ABAQUS. The general idea in [Araújo Fernandes et al., 2017] is to choose an initial step size which is then increased if the convergence rate is high and decreased if the convergence rate is slow.

Element formulation

The element formulation used in the finite-element implementation, Chap. 4, is simply for ease of implementation. However, it may be deficient for nonlinear analysis due to locking and nearly incompressible behavior of the material model as the RD approaches unity. While it is not definitively known, it is hypothesized that this is the cause of the difference in response of the current implementation, as compared to ANSYS, cf. Figs. 5.15 to 5.16. A direct comparison was infeasible since ANSYS does not support conventional isoparametric elements for sintering, and the current MATLAB implementation does not include enhanced elements.

Shear-locking may be alleviated by using either higher-order or enhanced elements [Lund and Lindgaard, 2022]. Linear elements are preferred due to the lower computational costs and since a high mesh resolution is needed for topology optimization. Therefore, enhanced elements would be the preferred method. As seen in [Crisfield, 2000, pp. 296], conventional isoparametric elements perform exceedingly poorly compared to enhanced elements. Different enhanced formulations exist, and a study on the influence of each in the context of sintering and shear-locking is warranted. A simple comparison of different element formulations, using the benchmark of Figs. 5.15 to 5.16 in Sec. 5.3.3, has been performed in ANSYS, as seen in Figs. 9.1 to 9.2. However, the sintering schedule is slightly altered, using an initial and terminal temperature of 22 C° and an isothermal hold at 1380 C° for 7200 s. For referencing the element formulations, the reader is referred to the ANSYS manual for element technology, [ANSYS, 2022].

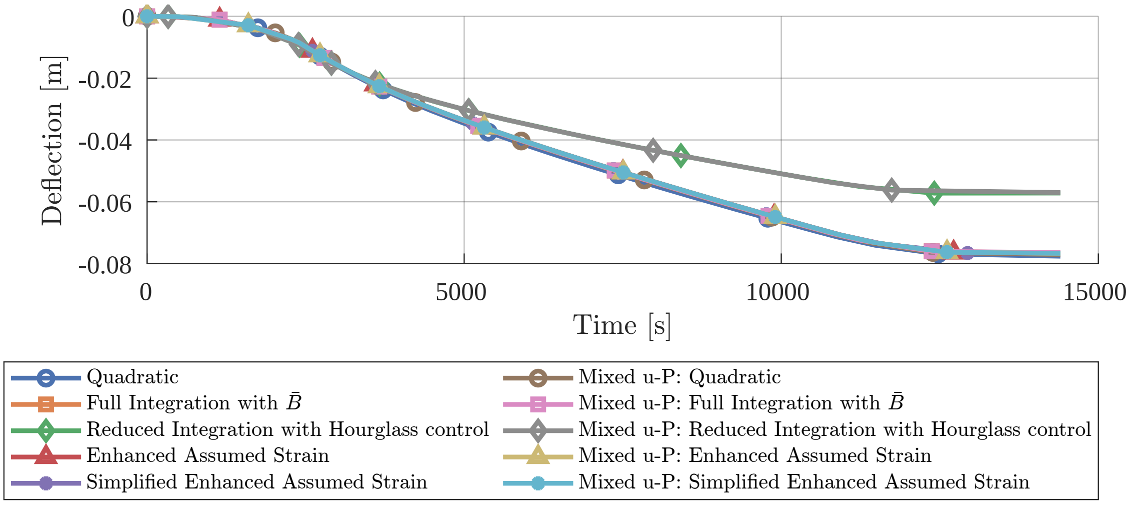


Figure 9.1. Deflection of beam for different element technologies.

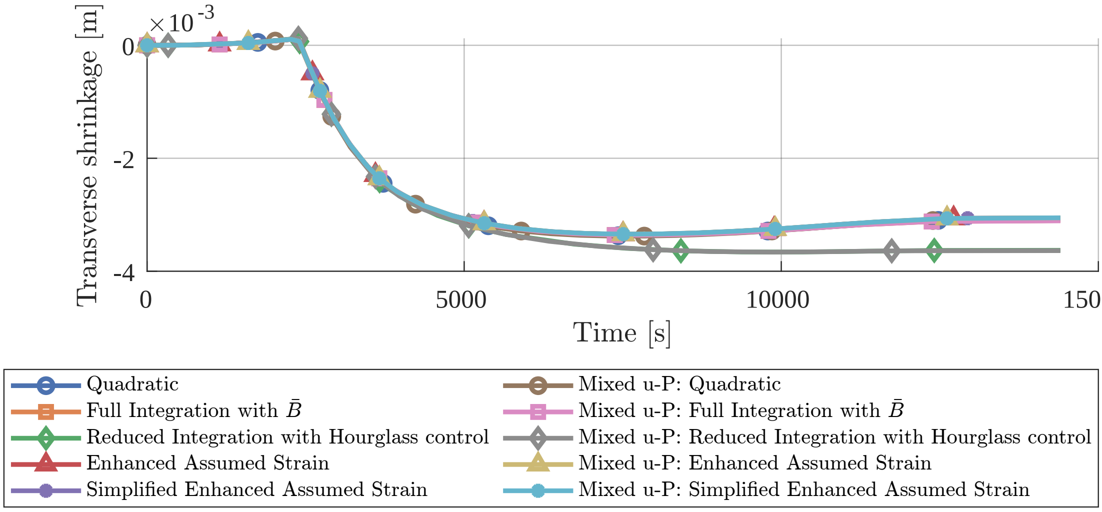


Figure 9.2. Transverse shrinkage of beam for different element technologies.

The difference between the different element formulations is relatively insignificant compared to the difference observed in Figs. 5.15 to 5.16. However, it can be seen that the reduced integration with hourglass control shows less deformation than other element technologies. This may be due to the artificial hourglass stiffness.

As the RD approaches unity, the volumetric viscosity coefficient approaches infinity. Thus, this results in similar incompressible behavior, as seen for, e.g., elasticity as ν approaches 0.5, resulting in the bulk modulus approaching infinity. Combined with the large volumetric strains during sintering, this may result in volumetric locking. As similarly seen in [Crisfield, 2000, pp. 296], for nearly incompressible materials, the deficiency of the conventional linear elements increases. Both linear and higher-order elements may suffer from volumetric locking [Lund and Lindgaard, 2022], further increasing the need for enhanced elements. In Figs. 9.1 to 9.2, no apparent locking for the quadratic elements is observed. However, this is likely due to ANSYS defaulting to uniform reduced integration for solid quadratic elements, which specifically alleviates volumetric locking. However, volumetric locking may also be alleviated by independently interpolating the hydrostatic pressure in the elements and solving it globally using a mixed u-P formulation [Lund and Lindgaard, 2022]. This has likewise been investigated for sintering in ANSYS, as seen in Figs. 9.1 to 9.2. No large differences were seen as compared to the pure displacement formulations. Enhanced elements or a mixed formulation would introduce internal DOFs that would be related to, e.g., incompatible bending modes, enhanced strain interpolations in the case of enhanced elements [Crisfield, 2000] or volumetric pressure in the case of a mixed formulation. This would further complicate the DSA, since these internal DOFs would introduce additional terms in the partial derivatives [Alberdi et al., 2018].

Material formulation

The material model introduced in Chap. 5 is based on Taylor approximations of the viscoplastic strain at the end of the timestep; thus, it is only semi-implicit [Braginsky et al., 2003]. Shi et al. [2023] used a fully implicit backward Euler integration algorithm for sintering. However, as demonstrated in Figs. 5.8 to 5.9, the results were relatively insensitive to the time-step size. This may be because sintering is a slow process, such that the change of the viscoplastic strain rate over a time step is approximately linear, thereby leading to accurate estimates of the strain rate. Nonetheless, this may only be inherent to the specific sintering schedule and low applied load in gravity. If frictional contact or applied pressure is introduced, this accuracy may be lost. Introducing a fully implicit material model comes at the computational expense of inner Newton-Raphson iterations. However, as shown in Shi et al. [2023], the residual equations can be reduced to a single variable, resulting in an efficient numerical integration.

Domain extension

As discussed in Sec. 6.5 it was desired to include domain extension in the topology optimization problem. Due to computational demands and using general discretization i.e. the design domain can be arbitrarily defined, it is difficult to generate the extended domain consistently. The boundary 'sticking' problem implicitly introduced by the Neumann boundary conditions is visible in the results in Figs. 8.2 to 8.9. In 3D there are also more surfaces with implicit Neumann boundary conditions for the design to 'stick' to, compared with 2D design domains. If the geometric length scale constraints are used, as briefly introduced in Sec. 6.4, they could also benefit from using larger extended domains than proposed by Clausen and Andreassen [2017] since the length scale constraint would require a filter diameter width of extended domain to avoid problems with constraint violation [Li et al., 2023]. The computational demand for domain extension in 3D can exceed the computation demand for the design domain since the extended domain is external to the design domain.

Length Scale Constraint

The minimum length scale constraint briefly introduced in Sec. 6.4 and further elaborated upon in Appendix B was investigated in this project to ensure minimum length scale when using projection filtering. The length scale constraint appeared promising, based on the results in [Li et al., 2023], however the attempted implementation in this report resulted in constant constraint violations when the constraint relaxation approached zero. For larger relaxation values, the optimizer adhered to the constraint. The sensitivity of the implemented constraint was verified with FDA. The approach utilized in this report used a Kreisselmeier-Steinhauser upper bound aggregation of the element-level defined constraints. Using the P-mean norm as suggested originally by [Li et al., 2023] could result in designs without constraint violation.

The minimum length scale constraint could also serve as an implicit buckling constraint. Typically, topology-optimized designs for linear elastic compliance suffer from buckling failure modes, which are often not considered during optimization, e.g. [Gadegaard and Thuesen, 2022]. By considering the optimized designs in Figs. 8.15 to 8.34, many thin members are present in the design, which could fail due to local buckling. Buckling has not been considered in this project and introducing the length scale constraint might initially alleviate buckling problems by ensuring the minimum length scale without introducing explicit buckling constraints.

Application in practice

The examples of potential applications, shown in Sec. 8.4, raise the question of how to utilize this framework as an industrial design tool. Contrary to most topology optimization, often used to design topologies from scratch, this framework may be the most useful when altering existing topologies to enhance their manufacturability by MBJ.
One prospect is that these modifications should be temporary supports removed from the structure post-sintering. However, this adds additional manufacturing costs for removing supports and surface treatment. Another, perhaps more interesting possibility, would be to use it for permanent design alterations, integrated seamlessly into the existing topology application.
However, when modifying existing topologies, these should not compromise the functionality of the design in its intended use case. Therefore, constraints on modifications may be implemented. In Sec. 8.4, this was solely defined by an allowable design domain. However, it could also be a maximum compliance or stress requirement of the modified as-sintered component. An additional linear analysis could implement these as a constraint on the resulting design. This is, however, not an easy extension, as it involves two couplings to the primal sintering analysis: The nodal terminal locations are a function of the PD, and the resulting material stiffness and strength criterion is a function of the terminal value of the RD from the sintering analysis [Kuhn, 1971][Zhang et al., 2021].

Contact and friction

Perhaps the biggest limitation of this report is the omission of frictional and contact effects between the 3D printed part and the ceramic setter plate during sintering. Due to the scope of the task, including contact and frictional effects was omitted in this report. However, 3D-printed MBJ parts experience distortion during sintering due to friction forces. Therefore, the topology-optimized design will likely also differ if contact and friction are included in the NLFEA. Including contact and friction in the current MATLAB NLFEA topology optimization implementation is not a trivial task. The contact implementation has to support node-to-surface contact detection and sliding frictional modeling. One approach could follow from [Kristiansen et al., 2021] where topology optimization was performed on GNLFEA with frictional contact. The contact formulation follows from [Poulios and Renard, 2015] and is not discontinuous between two bodies which has been realized without smoothing.

Compiled language

As described in Sec. 7.6.1, for an 8-core CPU with parallelized assembly of sensitivity analysis Jacobian matrices and tangent stiffness matrices the duration of a single optimizer iteration is about 3.5 minutes for the current MATLAB implementation. As shown in Figs. 8.10 to 8.13, none of the optimization tests converged within 400 iterations, and the computational time exceeded 24 hours for all the corresponding optimization tests. Thus warranting a discussion on how to decrease computational time. A change of programming language from a Just-In-Time compiled language such as MATLAB to a fully compiled language such as Fortran or C++ could decrease computational time at the cost of debugging possibilities and, therefore, increase development time. If larger models than presented in Chap. 8 have to be analyzed and optimized, the computation time will drastically increase and the developed framework will become infeasible as a design tool. Andrews [2012] shows a 500-fold increase in computational time between C++ and MATLAB, however the exact increase is problem-dependent.

Objective functions

In Sec. 6.9, different objective functions were formulated for optimization, namely displacements, compliance, strain energy, and deviatoric work. These were primarily heuristic, as no previous attempts have been made on the topic. Therefore, further investigations of which objective functions could be relevant are warranted. As mentioned in the context of constraints in Chap. 9, the objective could be measured based on the as-sintered geometry in its application instead of the behavior during sintering. However, refocusing on optimal sintering behavior, several unexplored but beneficial formulations may exist. For example, observing the results in Zhang et al. [2021] and Borujeni et al. [2022], there may be a correlation between RD variation in areas and their predictability by FE. Thus, an RD-based objective formulation may result in higher predictability. Nonetheless, the objective should be based on a measure that can experimentally verify improved behavior during sintering.

Experimental validation

The work done in this thesis is purely numerical simulations. The only validation is the comparison to ANSYS, cf. Figs. 5.15 to 5.16, and the analytical densification model, cf Figs. 5.3 to 5.5. The former did not show agreement, while the latter did. Ideally, validation would be experimental, as done in, e.g., Zhang et al. [2021]. Experimental validation could be performed on several stages of the results in this project. Firstly, to determine accurate material constants. Secondly, to verify the material model. Thirdly, to verify that design modification obtained using the optimization algorithm does indeed enhance performance. Furthermore, as shown in Fig. 1.3, lower nominal compensation results in a higher percentage of achieved tolerance. Experimental investigations could be used to define additional and more accurate measures that result in increased predictability of the sintering simulation and, in turn, a better achievable tolerance.

Damage and fracture mechanics

In [Borujeni et al., 2022] problems with cracks in and fracture of MBJ 3D printed parts after sintering were shown. The problem regarding fracture and cracks is currently not implemented in the topology optimization formulation and the designs presented in Chap. 8 might also suffer from sintering-induced cracks or fracture. To ensure future designs are robust against sintering-driven cracks or fracture, a local damage or fracture constraint could be implemented such that the sintered parts do not experience these failure modes. Including a fracture or damage constraint might require further analysis of the sintering process and the behavior of the sintered parts during sintering since they transition from a porous and brittle state into a more solid and ductile state. Often, failure criteria are split into criteria for brittle and ductile responses whilst fracture mechanics are primarily applied to brittle responses. Addition of damage and fracture constraints might therefore require hybrid approaches or identification of which state the sintered parts experience most damage or crack initiation in. Current work on the subject of fracture resistance in topology optimization is typically aimed toward designing energy-absorbing structures e.g. Li and Khandelwal [2017], so novel methods might be needed.

Unified compensation and optimization

As stated in the introduction, Chap. 1, resulting geometries require deformation compensation before printing, so the sought-after final as-sintered geometry is reached. A possible extension to this framework could be to use the deformations from an optimized design and compensate the nodal locations. This could be the residual deformation from a target geometry as shown in Sec. 8.4.3. The converged design from this case could be used for a sintering analysis, again without gravity, to simulate target deformations from the modified design. The deviations between simulation with- and without gravity could be used for compensation of the design. The compensated, modified design can then be used as an initial guess for further optimization. This iterative procedure could be performed until the residual deformations are as low as can be achieved. This could potentially greatly enhance the accuracy of compensation. Thus, exploring this unified compensation- and optimization may be beneficial for future use.

10 Conclusion

Non-linear sintering theory has been presented and implemented in a non-linear semi-implicit material model. The sintering material model was implemented in a geometrically non-linear finite element analysis such that large displacement gradients could be modeled. Discontinuities in the material model were smoothed such that first-order derivatives could be found for the DSA in the topology optimization. The combination of material model and finite element analysis was implemented in a MATLAB program and validated against an analytical 1D solution and the commercial FEA program ANSYS. The response of the self-developed NLFEA program agreed with the 1D analytical solution for various time-step sizes. In comparison with ANSYS, the internal variables and material properties were in good agreement but the deformation was different. This discrepancy could be due to differences in element technology used in the two different programs. Further, it was shown that the NR-solver has quadratic convergence properties.

The validated material model and finite element analysis were then extended to include topology optimization. Multiple material interpolation schemes were introduced and assessed. Due to the attractive ratio of material density to 'stiffness' at low pseudo-density values and non-zero gradients at zero pseudo-density, the RAMP interpolation scheme was chosen. To avoid checkerboarding problems a linear density filter was used in addition to threshold projection filtering to remove intermediate densities. To promote more natural designs, domain extension was introduced but omitted due to increased computational demand. When gravitational loads are included in topology optimization the loading becomes design-dependent and lower volume constraints were therefore introduced to avoid the trivial solution. Since topology optimization used in conjunction with sintering process simulations is a novel field, multiple criteria functions were introduced to judge their relevance in reducing deformation and distortion. To perform sensitivity analysis for the gradient-based MMA optimizer, the path-dependent adjoint method was used due to the path-dependent response of the NLFEA program. Problems with large condition numbers of the Jacobian matrix for the sensitivity analysis prompted a change of units to increase numerical accuracy and speed. The sensitivities of the implicit structural and material constraints along with all criterion functions, constraints and filters were validated with finite difference approximation.

To show the effect of the implemented topology optimization, the different criteria functions were first tested on a quarter MBB beam. The four criteria functions resulted in four visibly different designs however criteria function f_3 resulted in an unphysical objective function value and a design dominated by intermediate densities. It is difficult to determine which of the four criteria functions are best suited for reducing deformation, without performing further experimental analysis. As shown in the test cases it is very easy to include new criteria functions. To showcase the framework in a practical application, a test geometry was introduced in Fig. 8.14 where the goal was to reinforce the geometry and not completely redesign the entire geometry. It was shown that using criteria function f_4 and the modified f_1 displacement criterion both resulted in lower deformation and distortion of the optimized design

Returning to the problem formulation of Chap. 2 restated here again for convenience:

> ***How can density-based topology optimization be used to reduce the deformation of MBJ 3D-printed components due to sintering?***

In review, it is possible to use density-based topology optimization in the developed framework to reduce deformation induced by sintering of MBJ 3D printed components as shown in Chap. 8. The developed topology optimization framework shows promise in reducing deformation of MBJ 3D printed specimens and therefore greatly improving the future prospects of metal binder jetting 3D printing in industrial settings. While preliminary, it has the potential as a tool to change how designing against sintering in additive manufacturing is approached.

Bibliography

Al-Qudsi et al., 2015. Ahmad Al-Qudsi, Matthias Kammler, Anas Bouguecha, Christian Bonk and Bernd-Arno Behrens. *Comparison between different numerical models of densification during solid-state sintering of pure aluminium powder.* Production Engineering, 9, 2015. doi: 10.1007/s11740-014-0574-7.

Alberdi et al., 2018. Ryan Alberdi, Guodong Zhang, Lei Li and Kapil Khandelwal. *A unified framework for nonlinear path-dependent sensitivity analysis in topology optimization.* International Journal for Numerical Methods in Engineering, 115(1), 1–56, 2018. doi: https://doi.org/10.1002/nme.5794. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.5794`.

Andrews, 06 2012. Tyler Andrews. *Computation Time Comparison Between Matlab and C++ Using Launch Windows.* 2012.

ANSYS, 2022. Inc ANSYS. *Element Reference 2022 R1*, 2022.

ANSYS, 2023. Inc ANSYS. *Material Reference 2023 R2*, 2023.

Araújo Fernandes et al., 2017. M. T. C. Araújo Fernandes, C.O. Cardoso and W. J. Masur. *An Adaptive Load Stepping Algortihm for Path-Dependent Probles Based on Estimated Convergence Rates.* Computer modeling in engineering & sciences, 2017.

Argüello et al., 2009. José Guadalupe Argüello, Markus W. Reiterer and Kevin G. Ewsuk. *Verification, Performance, Validation, and Modifications to the SOVS Continuum Constitutive Model in a Nonlinear Large-Deformation Finite Element Code.* Journal of the American Ceramic Society, 92(7), 1442–1449, 2009. doi: https://doi.org/10.1111/j.1551-2916.2009.03008.x. URL `https://ceramics.onlinelibrary.wiley.com/doi/abs/10.1111/j.1551-2916.2009.03008.x`.

Bandyopadhyay and Bose, 2016. Amit Bandyopadhyay and Susmita Bose. *Additive Manufacturing.* Taylor & Francis Group, 2016.

Bendsøe and Kikuchi, 1988. Martin Philip Bendsøe and Noboru Kikuchi. *Generating optimal topologies in structural design using a homogenization method.* Computer methods in applied mechanics and engineering, Journal 71/1988, pp. 197–224, 1988.

Bendsøe, 1989. M.P. Bendsøe. *Optimal shape design as a material distribution problem.* Structural Optimization, 1, p. 193–202, 1989.

Borujeni et al., **2022**. Shahrooz Sadeghi Borujeni, Anwar Shad, Kiranmayi Abburi Venkata, Nico Günther and Vasily Ploshikin. *Numerical simulation of shrinkage and deformation during sintering in metal binder jetting with experimental validation.* Materials & Design, 216, 2022.

Borujeni et al., **2023**. Shahrooz Sadeghi Borujeni, Gursimran Singh Saluja and Vasily Ploshikhin. *Compensation of sintering deformation for components manufactured by metal binder jetting using numerical simulations.* Rapid Prototyping Journal, 29(3), 612–625, 2023.

Bourdin, **2001**. Blaise Bourdin. *Filters in topology optimization.* International journal for numerical methods in engineering, pages 2143–2158, 2001.

Braginsky et al., **2003**. Michael Braginsky, Veena Tikare, Terry Garino and Jose Arguello. *Three-dimensional simulation of sintering using a continuum modeling approach.* 2003.

Bruggi, **2008**. Matteo Bruggi. *On an alternative approach to stress constraints relaxation in topology optimization.* Structural and multidisciplinary optimization, 36 (2), 125–141, 2008. ISSN 1615-147X.

Bruns and Tortorelli, **2001**. Tyler E. Bruns and Daniel A. Tortorelli. *Topology optimization of non-linear elastic structures and compliant mechanisms.* Computer Methods in Applied Mechanics and Engineering, 190(26), 3443–3459, 2001. ISSN 0045-7825. doi: https://doi.org/10.1016/S0045-7825(00)00278-4. URL `https://www.sciencedirect.com/science/article/pii/S0045782500002784`.

Bruns, **2005**. Tyler Eilert Bruns. *A reevaluation of the SIMP method with filtering and an alternative formulation for solid-void topology optimization.* Structural and multidisciplinary optimization, 30, p. 428–436, 2005.

Bruyneel and Duysinx, **2005**. M. Bruyneel and P. Duysinx. *Note on topology optimization of continuum structures including self-weight.* Structural and Multidisciplinary Optimization, 2005.

Clausen and Andreassen, **2017**. Anders Clausen and Erik Andreassen. *On filter boundary conditions in topology optimization.* Structural and multidisciplinary optimization, 56(5), 1147–1155, 2017. ISSN 1615-147X.

Crisfield, **1991**. M. A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures*, volume 1. John Wiley and Sons, 1991.

Crisfield, **2000**. M. A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures*, volume 2. John Wiley and Sons, 2000.

de Borst et al., **2012**. René de Borst, Mike A. Crisfield, Joris J. C. Remmers and Clemens V. Verhoosel. *Non-linear Finite Element Analysis of Solid and Structures.* Wiley, 2012.

de Borst et al., **2023**. René de Borst, Mike A. Crisfield, Joris J. C. Remmers and Clemens V. Verhoosel. *PyFEM*, 2023. URL `https://github.com/jjcremmers/PyFEM`. GitHub repository.

Deng et al., 2019. Hao Deng, Lin Cheng and Albert C. To. *Distortion energy-based topology optimization design of hyperelastic materials.* Structural and multidisciplinary optimization, 59(6), 1895–1913, 2019. ISSN 1615-147X.

Duretz et al., 2018. Thibault Duretz, Alban Souche, René de Borst and Laetitia Le Pourhiet. *The Benefits of Using a Consistent Tangent Operator for Viscoelastoplastic Computations in Geodynamics.* Geochemistry, Geophysics, Geosystems, 19(12), 4904–4924, 2018. doi: https://doi.org/10.1029/2018GC007877. URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GC007877`.

Félix et al., 2020. Luís Félix, Alexandra A. Gomes and Afzal Suleman. *Topology optimization of the internal structure of an aircraft subjected to self-weight load.* Engineering Optimization, 2020.

Gadegaard and Thuesen, 2022. Frederik Juel Gadegaard and Jan Thuesen. *Strength based optimization and test of fiber reinforced additively manufactured structures considering topology and fiber orientation*, 2022.

German, 1996. R. M. German. *Sintering Theory and Practice.* Wiley, 1. edition, 1996. ISBN 978-0-471-05786-4.

Guest et al., 2004. J. K. Guest, J. H. Prévost and T. Belytschko. *Achieving minimum length scale in topology optimization using nodal design variables and projection functions.* International journal for numerical methods in engineering, 61(2), 238–254, 2004. ISSN 0029-5981.

Jiang et al., 2019. D. Jiang, R. Hoglund and D. Smith. *Continuous Fiber Angle Topology Optimization for Polymer Composite Deposition Additive Manufacturing Applications.* Fibers, 7(2), 2019.

Kristiansen et al., 2021. Hansotto Kristiansen, Konstantinos Poulios and Niels Aage. *Topology optimization of structures in transient impacts with Coulomb friction.* International journal for numerical methods in engineering, 122(18), 5053–5075, 2021. ISSN 0029-5981.

Kuhn, 1971. HA Kuhn. *Deformation characteristics and plasticity theory of sintered powder materials.* Int. J. Powder Metall., 7, 15–26, 1971.

Lazarov and Sigmund, 2011. B. S. Lazarov and O. Sigmund. *Filters in topology optimization based on Helmholtz-type differential equations.* International Journal for Numerical Methods in Engineering, 2011.

Le et al., 2010. Chau Le, Julian Norato, Tyler Bruns, Christopher Ha and Daniel Tortorelli. *Stress-based topology optimization for continua.* Structural and multidisciplinary optimization, 41(4), 605–620, 2010. ISSN 1615-147X.

Lee et al., 2022. Yousub Lee, Peeyush Nandwana and Srdjan Simunovic. *Powder spreading, densifcation, and part deformation in binder jetting additive manufacturing.* Progress in Additive Manufacturing, 7, 2022.

Li and Khandelwal, 2017. Lei Li and Kapil Khandelwal. *Design of fracture resistant energy absorbing structures using elastoplastic topology optimization.* Structural and Multidisciplinary Optimization, 2017.

Li et al., 2020. Ming Li, Wenchao Do, Alaa Elwany, Zhijian Pei and Chao Ma. *Metal Binder Jetting Additive Manufacturing: A Literature Review.* Journal of Manufacturing Science and Engineering, 142, 2020.

Li et al., 2023. Quhao Li, Guowei Liang, Yunfeng Luo, Fengtong Zhang and Shutian Liu. *An explicit formulation for minimum length scale control in density-based topology optimization.* Computer Methods in Applied Mechanics and Engineering, 404, 115761, 2023. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2022.115761. URL `https://www.sciencedirect.com/science/article/pii/S0045782522007174`.

Lindgaard and Dahl, 2013. Esben Lindgaard and Jonas Dahl. *On compliance and buckling objective functions in topology optimization of snap-through problems.* Structural and multidisciplinary optimization, 47(3), 409–421, 2013. ISSN 1615-147X.

Liu et al., 2019. Baoshou Liu, Di Guo, Chao Jiang, Guangyao Li and Xiaodong Huang. *Stress optimization of smooth continuum structures based on the distortion strain energy density.* Computer methods in applied mechanics and engineering, 343, 276–296, 2019. ISSN 0045-7825.

Lund and Lindgaard, 2022. Erik Lund and Esben Lindgaard. *Notes for course on Finite Element Methods.* Course Notes, 2022.

McHugh and Riedel, 1997. Peter E. McHugh and Hermann Riedel. *A liquid phase sintering model: application to Si3N4 and WC-Co.* Acta Materialia, 45(7), 2995–3003, 1997. ISSN 1359-6454. doi: https://doi.org/10.1016/S1359-6454(96)00378-3. URL `https://www.sciencedirect.com/science/article/pii/S1359645496003783`.

Michaleris et al., 1994. Panagiotis Michaleris, Daniel A. Tortorelli and Creto A. Vidal. *Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity.* International Journal for Numerical Methods in Engineering, 37(14), 2471–2499, 1994. doi: https://doi.org/10.1002/nme.1620371408. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620371408`.

Munro, 2024. Dirk Munro. *A simple QP modification of the OC update to permit treatment of the topology design problem of self-weight.* Strctural and Multidisciplinary Optimization, 67, 2024.

Oest and Lund, 2017. Jacob Oest and Erik Lund. *Topology optimization with finite-life fatigue constraints.* Structural and Multidisciplinary Optimization, 2017. ISSN 1615147X.

Olesen et al., 2021. Asbjørn M. Olesen, Sebastian M. Hermansen and Erik Lund. *Simultaneous optimization of topology and print orientation for transversely isotropic fatigue.* Structural and Multidisciplinary Optimization, Journal 64, p. 1041–1062, 2021.

Olesen and Hermansen, 2020. Asbjørn Malte Olesen and Sebastian Malte Hermansen. *Topology Optimization of General Structures with Anisotropic Fatigue Constraints*, 2020.

Olevsky et al., 2000. Eugene Olevsky, Veena Tikare, Terry J Garino and Michael V. Braginsky. *Simulation of Sintering of Layered Structures.* 2000.

Olevsky, 1997. Eugene A. Olevsky. *Theory of Sintering: From discrete to continuum.* Materials Science and Engineering, 23, 1997.

Pedersen, 2024. *Mikkel Pedersen.* ANSYSimport (`https://www.mathworks.com/matlabcentral/fileexchange/66659-ansysimport`, MATLAB Central File Exchange, 2024. Retrieved February 1, 2024.

Pedersen and Pedersen, 2012. Pauli Pedersen and Niels L. Pedersen. *Interpolation/penalization applied for strength design of 3D thermoelastic structures.* Structural and multidisciplinary optimization, 45(6), 773–786, 2012. ISSN 1615-147X.

Poulios and Renard, 2015. Konstantinos Poulios and Yves Renard. *An unconstrained integral approximation of large sliding frictional contact between deformable solids.* Computers & Structures, 2015.

Rahaman, 2015. Mohamed N. Rahaman. *Sintering Theory and Fundamentals.* ASM International, 2015. ISBN 978-1-62708-175-7. doi: 10.31399/asm.hb.v07.a0006117.

Rios et al., 2021. Alberto Vabo Rios, Eduard Hryha, Eugene Olevsky and Peter Harlin. *Sintering anisotropy of binder jetted 316L stainless steel: part I - sintering anisotropy.* Powder Metallurgy, 65, p.273–282, 2021.

Sahli et al., 2015. M. Sahli, A. Lebied, J-C. Gelin, T. Barrière and B. Necib. *Numerical simulation and experimental analysis of solid-state sintering response of 316 L stainless steelmicro-parts manufactured by metal injection molding.* The International Journal of Advanced Manufacturing Technology, 79, 2015.

Schmidt et al., 2020. Martin-Pierre Schmidt, Laura Coret, Christian Gout and Claus B.W. Pedersen. *Structural topology optimization with smoothly varying fiber orientations.* Structural and Multidisciplinary Optimization, 62, p. 3105–3126, 2020.

Shi et al., 2023. Hao Shi, Diletta Giuntini, Hans van Dommelen, Marc G.D. Geers and Joris J.C. Remmers. *Efficient modelling of ceramic sintering processes: Application to bilayers and membranes.* Journal of the European Ceramic Society, 43(11), 4939–4949, 2023. ISSN 0955-2219. doi: https://doi.org/10.1016/j.jeurceramsoc.2023.03.053. URL `https://www.sciencedirect.com/science/article/pii/S0955221923002492`.

Sigmund, 2001. O. Sigmund. *A 99 line topology optimization code written in Matlab.* Struct Multidisc Optim, Journal 21, p. 120–127, 2001.

Sigmund and Petersson, 1998. O. Sigmund and J. Petersson. *Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima*, 1998.

Sigmund, 2007. Ole Sigmund. *Morphology-based black and white filters for topology optimization.* Structural and Multidisciplinary Optimization, 33, p. 401–424, 2007.

Sigmund, 1994. Ole Sigmund. *Design of Material Structures Using Topology Optimization*, 1994. PhD thesis.

Sigmund and Maute, 2013. Ole Sigmund and Kurt Maute. *Topology optimization approaches A comparative review.* Structural Multidisciplinary Optimization, Journal 48/2013, pp. 1031–1055, 2013.

Simo and Taylor, 1985. J.C. Simo and R.L. Taylor. *Consistent tangent operators for rate-independent elastoplasticity.* Computer Methods in Applied Mechanics and Engineering, 48(1), 101–118, 1985. ISSN 0045-7825. doi: https://doi.org/10.1016/0045-7825(85)90070-2. URL `https://www.sciencedirect.com/science/article/pii/0045782585900702`.

Skorokhod, 1972. Valeriy V. Skorokhod. *Rheological Basis of the Theory of Sintering.* 1972.

Song et al., 2006. J. Song, J. C. Gelin, T. Barrière and B. Liu. *Experiments and numerical modelling of solid state sintering for 316L stainless steel components.* Journal of Materials Processing Technoology, 133, 2006.

Stolpe and Svanberg, 2001a. Mathias Stolpe and Krister Svanberg. *An alternative interpolation scheme for minimum compliance topology optimization.* Structural and multidisciplinary optimization, 22, p. 116–124, 2001a.

Stolpe and Svanberg, 2001b. Mathias Stolpe and Krister Svanberg. *On the trajectories of penalization methods for topology optimization.* Structural and multidisciplinary optimization, 21, p. 128–139, 2001b.

Svanberg, 1987. Krister Svanberg. *The method of moving asymptotes - a new method for structural optimization.* International journal for numerical methods in engineering, 24, p. 359–373, 1987.

Svanberg, ND. Krister Svanberg. *MMA and GCMMA: two methods for nonlinear optimization*, ND.

Troelsgaard and Elmstrøm, 2023. Christian Troelsgaard and Frederik T. Elmstrøm. *Inverse modeling of sintered MBJ 3D printed specimens.* 2023.

Troelsgaard et al., 2023. Christian Troelsgaard, Frederik Østergaard, Frederik Elmstrøm, Jacob Lykkegård Hansen and Rasmus Kaalund Schøn. *The Science of Compliance: Continuous Fiber Angle Topology Optimization with Stress Constraints and Path-Planning.* MechMan Symposium, 11th, 2023. URL `https://www.mp.aau.dk/education/student-symposiums/mechman#2023`.

Vidal et al., 1991. C.A. Vidal, H.-S. Lee and R.B. Haber. *The consistent tangent operator for design sensitivity analysis of history-dependent response.* Computing Systems in Engineering, 2(5), 509–523, 1991. ISSN 0956-0521. doi: https://doi.org/10.1016/0956-0521(91)90053-8. URL https://www.sciencedirect.com/science/article/pii/0956052191900538. Computational Structures Technology Part 2.

Wang et al., 2010. Fengwen Wang, Boyan Stefanov Lazarov and Ole Sigmund. *On projection methods, convergence and robust formulations in topology optimization.* Structural and multidisciplinary optimization, 43, p. 767–784, 2010.

Wu et al., 2017. Jun Wu, Anders Clausen and Ole Sigmund. *Minimum compliance topology optimization of shell–infill composites for additive manufacturing.* Computer methods in applied mechanics and engineering., 326, p. 358–375, 2017.

Zago et al., 2021. Marco Zago, Nora Francesca Maria Lecis, Maurizio Vedani and Ilaria Cristofolini. *Dimensional and geometrical precision of parts produced by binder jetting process as affected by the anisotropic shrinkage on sintering.* Additive Manufacturing, 43, 102007, 2021. ISSN 2214-8604. doi: https://doi.org/10.1016/j.addma.2021.102007. URL https://www.sciencedirect.com/science/article/pii/S221486042100172X.

Zhang et al., 2021. Kaiwen Zhang, Wei Zhang, Ryan Brune, Edward Herderick, Xu Zhang, John Cornell and Jay Forsmark. *Numerical simulation and experimental measurement of pressureless sintering of stainless steel part printed by binder jetting additive manufacturing.* Additive Manufacturing, 47, 2021.

Zhang, 2005. Rui Zhang. *Numerical simulation of solid-state sintering of metal powder compact dominated by grain boundary diffusion.* Pennsylvania State University, 2005.

Zhou et al., 2015. Mingdong Zhou, Boyan S. Lazarov, Fengwen Wang and Ole Sigmund. *Minimum length scale in topology optimization by geometric constraints.* Computer Methods in Applied Mechanics and Engineering, 293, 266–282, 2015. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2015.05.003. URL https://www.sciencedirect.com/science/article/pii/S0045782515001693.

A Partial derivatives

This appendix states the explicit form of the equations and derivatives used in the structural- and design sensitivity analysis. Since more details are included, it warrants a slight change of notation for readability and ease of derivation.

A.1 General notation

There will be used a different notation for vectors and matrices in this appendix, for ease of notation. A vector will be denoted by a subscript $[\,]_i$.
A superscript $[\,]^{(i)}$ denotes that it is evaluated for variables at the i^{th} loadstep.

$$f(x,y,g(x,y))\ , \qquad f^{(i)} = f\Big|_{\substack{x=x^{(i)}\\ y=y^{(i)}\\ g=g(x^{(i)},y^{(i)})}} \qquad \left(\frac{\partial f}{\partial x}\right)^{(i)} = \frac{\partial f}{\partial x}\Big|_{\substack{x=x^{(i)}\\ y=y^{(i)}\\ g=g(x^{(i)},y^{(i)})}} \tag{A.1}$$

A.2 From tensor to matrix notation

Firstly, the tensor notation of the viscoplastic- and relative density equation has to be converted to a matrix notation that are consistent with the current implementation. The tensor notation of the viscoplastic strain rate is given by:

$$\dot{\varepsilon}^{vp}_{ij} = \frac{1}{2\eta\phi}S'_{ij} + \frac{1}{18\eta\psi}(S_{kk} - 3P_L)\delta_{ij} \tag{A.2}$$

$$= \frac{1}{2\eta\phi}\left(S_{ij} - \frac{1}{3}S_{kk}\delta_{ij}\right) + \frac{1}{18\eta\psi}S_{kk}\delta_{ij} - \frac{1}{6\eta\psi}P_L\delta_{ij} \tag{A.3}$$

$$= \frac{1}{2\eta\phi}S_{ij} + \left(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\right)S_{kk}\delta_{ij} - \frac{1}{6\eta\psi}P_L\delta_{ij} \tag{A.4}$$

The trace operator $[\,]_{kk}$ in matrix notation is given by $\mathbf{A} = \begin{Bmatrix}1 & 1 & 1 & 0 & 0 & 0\end{Bmatrix}$. Hereby, Kronecker delta in matrix notation can be described as $\mathbf{A}^T$. Hereby, Eq. (A.2) can be described in vector notation:

$$\dot{\varepsilon}^{vp}_i = \frac{1}{2\eta\phi}S_i + \left(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\right)\mathbf{A}S_i\mathbf{A}^T - \frac{1}{6\eta\psi}P_L\mathbf{A}^T \tag{A.5}$$

$$= D^{vp}S_i - S^s_i \tag{A.6}$$

where :

$$D^{vp} = \frac{1}{2\eta\phi}\mathbf{I} + \left(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\right)\mathbf{A}^T\mathbf{A} \qquad S^s_i = \mathbf{A}^T\frac{1}{6\eta\psi}P_L \tag{A.7}$$

Where I is the usual identity matrix. The densification rate, $\dot{d}$ is defined in tensor form as:

$$\dot{d} = -d\dot{\varepsilon}_{kk}^{vp} \tag{A.8}$$

By a similar approach, this can be rewritten in vector format, using the previous results:

$$\begin{aligned}
\dot{d} &= -d\mathbf{A}\dot{\varepsilon}_i^{vp} && \text{(A.9)}\\
&= -d\mathbf{A}\bigg(\Big(\frac{1}{2\eta\phi}\mathbf{I} + \Big(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\Big)\mathbf{A}^T\mathbf{A}\Big)S_i - \mathbf{A}^T\frac{1}{6\eta\psi}P_L\bigg) && \text{(A.10)}\\
&= -d\bigg(\Big(\frac{1}{2\eta\phi}\mathbf{A} + \Big(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\Big)\mathbf{A}\mathbf{A}^T\mathbf{A}\Big)S_i - \mathbf{A}\mathbf{A}^T\frac{1}{6\eta\psi}P_L\bigg) && \text{(A.11)}\\
&= -d\bigg(\Big(\frac{1}{2\eta\phi}\mathbf{A} + \Big(\frac{1}{18\eta\psi} - \frac{1}{6\eta\phi}\Big)3\mathbf{A}\Big)S_i - 3\frac{1}{6\eta\psi}P_L\bigg) && \text{(A.12)}\\
&= -d\bigg(\Big(\frac{1}{2\eta\phi}\mathbf{A} + \Big(\frac{1}{6\eta\psi} - \frac{1}{2\eta\phi}\Big)\mathbf{A}\Big)S_i - \frac{1}{2\eta\psi}P_L\bigg) && \text{(A.13)}\\
&= -d\bigg(\Big(\frac{1}{6\eta\psi}\mathbf{A}S_i - \frac{1}{2\eta\psi}P_L\bigg) && \text{(A.14)}\\
&= C^d - D^d S_i && \text{(A.15)}
\end{aligned}$$

where:

$$C^d = d\frac{P_L}{2\eta\psi} \qquad D^d = \frac{d}{6\eta\psi}\mathbf{A} \tag{A.16}$$

It is useful to note two things; The densification rate is independent of the shear-viscosity moduli, ϕ, and this results in the following relation:

$$\dot{d} = \frac{d}{2\eta\psi}(P_L - \frac{1}{3}\mathbf{A}S_i) \tag{A.17}$$

This means that the densification rate is positive if the sintering stress is larger than the mean stress. It also means that compressive mean stress increases densification. Further, since $P_L \geq 0$, then the densification rate is always positive for free sintering.

A.3 Structural analysis

To update the material model during the structural analysis, three residual or constraint equations are defined, which govern the evolution of the internal variables. The internal variables are defined as:

$$\{\mathbf{c}\}^{(i)} = \Big\{S^{(i)} \quad d^{(i)} \quad G^{(i)}\Big\}^T \tag{A.18}$$

And their corresponding residual being:

$$\{\mathbf{H}\}^{(i)} = \Big\{h_\varepsilon^{(i)} \quad h_d^{(i)} \quad h_G^{(i)}\Big\}^T = \{\mathbf{0}\} \tag{A.19}$$

Which for the proposed implementation results in a linear system of equations that can be solved for the increment in the internal variables over a time step. The explicit linear equations being solved are given by:

$$h_{\varepsilon}^{(i)} = \begin{aligned} &\left(D^{-1} + \Delta t\left(\frac{\partial \dot{\varepsilon}^{vp}}{\partial S}\right)^{(i-1)}\right)(S^{(i)} - S^{(i-1)}) + \left(\Delta t\left(\frac{\partial \dot{\varepsilon}^{vp}}{\partial d}\right)^{(i-1)}\right)(d^{(i)} - d^{(i-1)}) \\ &+\left(\Delta t\left(\frac{\partial \dot{\varepsilon}^{vp}}{\partial G}\right)^{(i-1)}\right)(G^{(i)} - G^{(i-1)}) - (\varepsilon^{(i)} - \varepsilon^{(i-1)}) \\ &+\Delta t\left(\left(\dot{\varepsilon}^{vp}\right)^{(i-1)} + \left(\frac{\partial \dot{\varepsilon}^{vp}}{\partial T}\right)^{(i-1)}(T^{(i)} - T^{(i-1)})\right) \end{aligned} \tag{A.20}$$

$$h_{d}^{(i)} = \begin{aligned} &-\Delta t\left(\frac{\partial \dot{d}}{\partial S}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \left(1 - \Delta t\left(\frac{\partial \dot{d}}{\partial d}\right)^{(i-1)}\right)(d^{(i)} - d^{(i-1)}) \\ &-\Delta t\left(\frac{\partial \dot{d}}{\partial G}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) \\ &-\Delta t\left(\left(\dot{d}\right)^{(i-1)} + \left(\frac{\partial \dot{d}}{\partial T}\right)^{(i-1)}(T^{(i)} - T^{(i-1)})\right) \end{aligned} \tag{A.21}$$

$$h_{G}^{(i)} = \begin{aligned} &-\Delta t\left(\frac{\partial \dot{G}}{\partial S}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) - \Delta t\left(\frac{\partial \dot{G}}{\partial d}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ &+\left(1 - \Delta t\left(\frac{\partial \dot{G}}{\partial G}\right)^{(i-1)}\right)(G^{(i)} - G^{(i-1)}) \\ &-\Delta t\left(\left(\dot{G}\right)^{(i-1)} + \left(\frac{\partial \dot{G}}{\partial T}\right)^{(i-1)}(T^{(i)} - T^{(i-1)})\right) \end{aligned} \tag{A.22}$$

It is implicitly implied that it is the derived vector quantities that are used. Thus, a total of 12 partial derivatives have to be derived for implementing the material model in the structural analysis. This will be done in the following section.

A.3.1 Material Residual equation

Essential terms

The goal of the following section is to derive and reduce the partial derivatives in the material model, down to its most basic part. Thus, every derivative in the following section consists of the following terms.

$$\phi = d^2 \qquad \frac{\partial \phi}{\partial d} = 2d \tag{A.23}$$

$$\psi = \frac{2d^3}{3(1-d)} \qquad \frac{\partial \psi}{\partial d} = \frac{2d^2(3-2d)}{3(1-d)^2} \tag{A.24}$$

$$P_L = \frac{\alpha d^2}{G} \qquad \frac{\partial P_L}{\partial d} = \frac{2\alpha d}{G} \qquad \frac{\partial P_L}{\partial G} = -\frac{\alpha d^2}{G^2} \qquad \frac{\partial P_L}{\partial T} = \frac{\partial \alpha}{\partial T}\frac{d^2}{G} \tag{A.25}$$

$$\eta = A_\eta \exp\left(\frac{Q_\eta}{RT}\right) \qquad \frac{\partial \eta}{\partial T} = -\frac{Q_\eta A_\eta}{RT^2}\exp\left(\frac{Q_\eta}{RT}\right) \tag{A.26}$$

$$\dot{G} = \frac{A_G}{G}\exp\left(\frac{Q_G(T)}{RT}\right) \tag{A.27}$$

Strain residual equation

Four partial derivatives has to be evaluated for the strain residual equation. These are:

$$\frac{\partial \dot{\varepsilon}^{vp}}{\partial S} \qquad \frac{\partial \dot{\varepsilon}^{vp}}{\partial d} \qquad \frac{\partial \dot{\varepsilon}^{vp}}{\partial G} \qquad \frac{\partial \dot{\varepsilon}^{vp}}{\partial T} \tag{A.28}$$

Which are differentiated below.

$$\frac{\partial \dot{\varepsilon}^{vp}}{\partial S} = D^{vp} \quad \frac{\partial \dot{\varepsilon}^{vp}}{\partial d} = \frac{\partial D^{vp}}{\partial d} S_i - \frac{\partial S_i^s}{\partial d} \quad \frac{\partial \dot{\varepsilon}^{vp}}{\partial G} = -\frac{\partial S_i^s}{\partial G} \quad \frac{\partial \dot{\varepsilon}^{vp}}{\partial T} = \frac{\partial D^{vp}}{\partial T} S_i - \frac{\partial S_i^s}{\partial T} \tag{A.29}$$

Thus, this reveals 5 additional derivatives that needs to be evaluated. These are derived to be:

$$\frac{\partial D^{vp}}{\partial d} = \frac{1}{\eta}\left(-\frac{\partial \phi}{\partial d}\frac{1}{2\phi^2}\mathbf{I} + \left(-\frac{\partial \psi}{\partial d}\frac{1}{18\psi^2} + \frac{\partial \phi}{\partial d}\frac{1}{6\phi^2}\right)\mathbf{A}^T\mathbf{A}\right) \tag{A.30}$$

$$\frac{\partial S_i^s}{\partial d} = \frac{1}{\eta}\left(-\frac{\partial \psi}{\partial d}\frac{1}{6\psi^2}P_L + \frac{1}{6\psi}\frac{\partial P_L}{\partial d}\right)\mathbf{A}^T \tag{A.31}$$

$$\frac{\partial S_i^s}{\partial G} = \frac{1}{\eta}\left(\frac{1}{6\psi}\frac{\partial P_L}{\partial G}\right)\mathbf{A}^T \tag{A.32}$$

$$\frac{\partial D^{vp}}{\partial T} = -\frac{\partial \eta}{\partial T}\frac{1}{\eta^2}\left(\frac{1}{2\phi}\mathbf{I} + \left(\frac{1}{18\psi} - \frac{1}{6\phi}\right)\mathbf{A}^T\mathbf{A}\right) \tag{A.33}$$

$$\frac{\partial S_i^s}{\partial T} = \frac{1}{\eta}\left(-\frac{\partial \eta}{\partial T}\frac{1}{\eta}\frac{1}{6\psi}P_L + \frac{1}{6\psi}\frac{\partial P_L}{\partial T}\right)\mathbf{A}^T \tag{A.34}$$

Relative density residual equation

Likewise, four partial derivatives have to be derived for the relative density residual equation. These are given by:

$$\frac{\partial \dot{d}}{\partial S} \qquad \frac{\partial \dot{d}}{\partial d} \qquad \frac{\partial \dot{d}}{\partial G} \qquad \frac{\partial \dot{d}}{\partial T} \tag{A.35}$$

Which are differentiated below.

$$\frac{\partial \dot{d}}{\partial S} = -D^d \quad \frac{\partial \dot{d}}{\partial d} = \frac{\partial C^d}{\partial d} - \frac{\partial D^d}{\partial d} S_i \quad \frac{\partial \dot{d}}{\partial G} = \frac{\partial C^d}{\partial G} \quad \frac{\partial \dot{d}}{\partial T} = \frac{\partial C^d}{\partial T} - \frac{\partial D^d}{\partial T} S_i \tag{A.36}$$

Thus, this reveals 5 additional derivatives that need to be evaluated. These are derived to be:

$$\frac{\partial C^d}{\partial d} = \frac{1}{\eta}\left(\frac{P_L}{2\psi} - d\frac{\partial \psi}{\partial d}\frac{P_L}{2\psi^2} + d\frac{\partial P_L}{\partial d}\frac{1}{2\psi}\right) \tag{A.37}$$

$$\frac{\partial D^d}{\partial d} = \frac{1}{\eta}\left(\frac{1}{6\psi} - d\frac{\partial \psi}{\partial d}\frac{1}{6\psi^2}\right)\mathbf{A} \tag{A.38}$$

$$\frac{\partial C^d}{\partial G} = \frac{1}{\eta}\frac{d}{2\psi}\frac{\partial P_L}{\partial G} \tag{A.39}$$

$$\frac{\partial C^d}{\partial T} = \frac{1}{\eta}\left(-\frac{\partial \eta}{\partial T}\frac{d}{2\eta\psi}P_L + \frac{d}{2\psi}\frac{\partial P_L}{\partial T}\right) \tag{A.40}$$

$$\frac{\partial D^d}{\partial T} = -\frac{\partial \eta}{\partial T}\frac{1}{\eta^2}\frac{d}{6\psi}\mathbf{A} \tag{A.41}$$

Grain-size residual equation

Likewise, four partial derivatives have to be derived for the grain size residual equation. These are given by:

$$\frac{\partial \dot{G}}{\partial S} \qquad \frac{\partial \dot{G}}{\partial d} \qquad \frac{\partial \dot{G}}{\partial G} \qquad \frac{\partial \dot{G}}{\partial T} \tag{A.42}$$

Which are derived to be:

$$\frac{\partial \dot{G}}{\partial S} = 0 \cdot \mathbf{A} \quad \frac{\partial \dot{G}}{\partial d} = 0 \quad \frac{\partial \dot{G}}{\partial G} = -\frac{\dot{G}}{G} \quad \frac{\partial \dot{G}}{\partial T} = \frac{A_G\left(Q_G - T\frac{\partial Q_G}{\partial T}\right)}{GRT^2}\exp\left(-\frac{Q_G}{RT}\right) \tag{A.43}$$

A.4 Design sensitivity analysis

The derivative of the material residual, for the current internal variables, as well as both the former and current global variables are stated in the main report. However, the design sensitivity analysis also necessitates the derivative of the residual equations, for the former internal variables. Due to the partial derivatives of the former section being evaluated for the former internal variables, this derivative is non-trivial. This is therefore explicitly derived in this section. It is derived as:

$$\frac{\partial \{\mathbf{H}\}^{(i)}}{\partial \{\mathbf{c}\}^{(i-1)}} = \begin{bmatrix} \frac{\partial h_\varepsilon^{(i)}}{\partial S^{(i-1)}} & \frac{\partial h_\varepsilon^{(i)}}{\partial d^{(i-1)}} & \frac{\partial h_\varepsilon^{(i)}}{\partial G^{(i-1)}} \\ \frac{\partial h_d^{(i)}}{\partial S^{(i-1)}} & \frac{\partial h_d^{(i)}}{\partial d^{(i-1)}} & \frac{\partial h_d^{(i)}}{\partial G^{(i-1)}} \\ \frac{\partial h_G^{(i)}}{\partial S^{(i-1)}} & \frac{\partial h_G^{(i)}}{\partial d^{(i-1)}} & \frac{\partial h_G^{(i)}}{\partial G^{(i-1)}} \end{bmatrix} \tag{A.44}$$

Where each term is given by:

$$\frac{\partial h_\varepsilon^{(i)}}{\partial S^{(i-1)}} = \begin{matrix} -D^{-1} + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial S^2}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d \partial S}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial S}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial S}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{matrix} \tag{A.45}$$

$$\frac{\partial h_\varepsilon^{(i)}}{\partial d^{(i-1)}} = \begin{matrix} \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial S \partial d}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d^2}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial d}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial d}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{matrix} \tag{A.46}$$

$$\frac{\partial h_\varepsilon^{(i)}}{\partial G^{(i-1)}} = \begin{matrix} \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial S \partial G}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d \partial G}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G^2}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial G}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{matrix} \tag{A.47}$$

$$\frac{\partial h_d^{(i)}}{\partial S^{(i-1)}} = \begin{matrix} \Delta t\left(\frac{\partial^2 \dot{d}}{\partial S^2}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{d}}{\partial d \partial S}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\frac{\partial^2 \dot{d}}{\partial G \partial S}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\frac{\partial^2 \dot{d}}{\partial T \partial S}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{matrix} \tag{A.48}$$

$$\frac{\partial h_d^{(i)}}{\partial d^{(i-1)}} = \begin{array}{c} \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial S \partial d}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) - 1 + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial d^2}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial G \partial d}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial T \partial d}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{array} \tag{A.49}$$

$$\frac{\partial h_d^{(i)}}{\partial G^{(i-1)}} = \begin{array}{c} \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial S \partial G}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial d \partial G}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial G^2}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial T \partial G}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{array} \tag{A.50}$$

$$\frac{\partial h_G^{(i)}}{\partial S^{(i-1)}} = \begin{array}{c} \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial S^2}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial d \partial S}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial G \partial S}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{d}}{\partial T \partial S}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{array} \tag{A.51}$$

$$\frac{\partial h_G^{(i)}}{\partial d^{(i-1)}} = \begin{array}{c} \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial S \partial d}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial d^2}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ +\Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial G \partial d}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial T \partial d}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{array} \tag{A.52}$$

$$\frac{\partial h_G^{(i)}}{\partial G^{(i-1)}} = \begin{array}{c} \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial S \partial G}\right)^{(i-1)}(S^{(i)} - S^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial d \partial G}\right)^{(i-1)}(d^{(i)} - d^{(i-1)}) \\ -1 + \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial G^2}\right)^{(i-1)}(G^{(i)} - G^{(i-1)}) + \Delta t\left(\dfrac{\partial^2 \dot{G}}{\partial T \partial G}\right)^{(i-1)}(T^{(i)} - T^{(i-1)}) \end{array} \tag{A.53}$$

A.4.1 Derivative of Material Residual equation

The goal of the following section is to derive each of the second-order partial derivatives explicitly.

Essential terms

As for the material residual, each derivative is stated with regards to a set of basic terms, given below.

$$\frac{\partial^2 \phi}{\partial d^2} = 2 \qquad \frac{\partial^2 \psi}{\partial d^2} = -\frac{4d(d^2 - 3d + 3)}{3(d-1)^3} \qquad \frac{\partial^2 P_L}{\partial G^2} = 2\frac{\alpha d^2}{G^3} \qquad \frac{\partial^2 P_L}{\partial G \partial d} = -2\frac{\alpha d}{G^2} \tag{A.54}$$

$$\frac{\partial^2 P_L}{\partial d^2} = 2\frac{\alpha}{G} \qquad \frac{\partial^2 P_L}{\partial G \partial T} = -\frac{\partial \alpha}{\partial T}\frac{d^2}{G^2} \qquad \frac{\partial^2 P_L}{\partial d \partial T} = 2\frac{\partial \alpha}{\partial T}\frac{d}{G} \tag{A.55}$$

The following is divided into partial derivatives for each row of $\frac{\partial \{\mathbf{H}\}^{(i)}}{\partial \{\mathbf{c}\}^{(i-1)}}$.

Strain residual

The partial derivatives that needs to be derived are

$$\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial S^2} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d \partial S} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial S} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial S} \tag{A.56}$$

$$\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d^2} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial d} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial d} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G^2} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial G} \tag{A.57}$$

Since the order of differentiation is arbitrary.

$$\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial S^2} = 0 \cdot \mathbf{A}^T \mathbf{A} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d \partial S} = \frac{\partial D^{vp}}{\partial d} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial S} = 0 \cdot \mathbf{A} \tag{A.58}$$

$$\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial d^2} = \frac{\partial^2 D^{vp}}{\partial d^2} S_i - \frac{\partial^2 S_i^s}{\partial d^2} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial S} = \frac{\partial D^{vp}}{\partial T} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G \partial d} = -\frac{\partial^2 S_i^s}{\partial G \partial d} \tag{A.59}$$

$$\frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial d} = \frac{\partial^2 D^{vp}}{\partial d \partial T} S_i - \frac{\partial^2 S_i^s}{\partial d \partial T} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial G^2} = -\frac{\partial^2 S_i^s}{\partial G^2} \qquad \frac{\partial^2 \dot{\varepsilon}^{vp}}{\partial T \partial G} = -\frac{\partial^2 S_i^s}{\partial T \partial G} \tag{A.60}$$

Which reveals seven additional derivatives that need to be evaluated.

$$\frac{\partial^2 D^{vp}}{\partial d^2} = \frac{1}{\eta}\left(\left(\frac{\partial \phi}{\partial d}^2 \frac{1}{\phi^3} - \frac{\partial^2 \phi}{\partial d^2}\frac{1}{2\phi^2}\right)\mathbf{I} + \left(\frac{\partial \psi}{\partial d}^2 \frac{1}{9\psi^3} - \frac{\partial^2 \psi}{\partial d^2}\frac{1}{18\psi^2} + \frac{\partial^2 \phi}{\partial d^2}\frac{1}{6\phi^2} - \frac{\partial \phi}{\partial d}^2 \frac{1}{3\phi^3}\right)\mathbf{A}^T\mathbf{A}\right) \tag{A.61}$$

$$\frac{\partial^2 S_i^s}{\partial d^2} = \frac{1}{\eta}\left(\frac{\partial^2 P_L}{\partial d^2}\frac{1}{6\psi} - \frac{\partial \psi}{\partial d}\frac{\partial P_L}{\partial d}\frac{1}{3\psi^2} - \frac{\partial^2 \psi}{\partial d^2}\frac{P_L}{6\psi^2} + \frac{\partial^2 \psi}{\partial d^2}^2 \frac{P_L}{3\psi^3}\right)\mathbf{A}^T \tag{A.62}$$

$$\frac{\partial^2 S_i^s}{\partial G \partial d} = -\frac{1}{\eta}\left(\frac{\partial P_L}{\partial G}\frac{\partial \psi}{\partial d}\frac{1}{6\psi^2} - \frac{\partial^2 P_L}{\partial G \partial d}\frac{1}{6\psi}\right)\mathbf{A}^T \tag{A.63}$$

$$\frac{\partial^2 D^{vp}}{\partial d \partial T} = -\frac{\partial \eta}{\partial T}\frac{1}{\eta^1}\left(-\frac{\partial \phi}{\partial d}\frac{1}{2\phi^2}\mathbf{I} + \left(-\frac{\partial \psi}{\partial d}\frac{1}{18\psi^2} + \frac{\partial \phi}{\partial d}\frac{1}{6\phi^2}\right)\mathbf{A}^T\mathbf{A}\right) \tag{A.64}$$

$$\frac{\partial^2 S_i^s}{\partial d \partial T} = \frac{1}{\eta}\left(-\frac{\partial \eta}{\partial T}\frac{1}{\eta}\left(-\frac{\partial \psi}{\partial d}\frac{1}{6\psi^2}P_L + \frac{1}{6\psi}\frac{\partial P_L}{\partial d}\right) - \frac{\partial \psi}{\partial d}\frac{1}{6\psi^2}\frac{\partial P_L}{\partial T} + \frac{1}{6\psi}\frac{\partial^2 P_L}{\partial d \partial T}\right)\mathbf{A}^T \tag{A.65}$$

$$\frac{\partial^2 S_i^s}{\partial G^2} = \frac{1}{\eta}\frac{\partial^2 P_L}{\partial G^2}\frac{1}{6\psi}\mathbf{A}^T \tag{A.66}$$

$$\frac{\partial^2 S_i^s}{\partial T \partial G} = \frac{1}{\eta}\left(\frac{\partial^2 P_L}{\partial G \partial T}\frac{1}{6\psi} - \frac{\partial \eta}{\partial T}\frac{1}{\eta}\frac{\partial P_L}{\partial G}\frac{1}{6\psi}\right)\mathbf{A}^T \tag{A.67}$$

Relative density residual

The partial derivatives that needs to be derived are

$$\frac{\partial^2 \dot{d}}{\partial S^2} \qquad \frac{\partial^2 \dot{d}}{\partial d \partial S} \qquad \frac{\partial^2 \dot{d}}{\partial G \partial S} \qquad \frac{\partial^2 \dot{d}}{\partial T \partial S} \tag{A.68}$$

$$\frac{\partial^2 \dot{d}}{\partial d^2} \qquad \frac{\partial^2 \dot{d}}{\partial G \partial d} \qquad \frac{\partial^2 \dot{d}}{\partial T \partial d} \qquad \frac{\partial^2 \dot{d}}{\partial G^2} \qquad \frac{\partial^2 \dot{d}}{\partial T \partial G} \tag{A.69}$$

Since the order of differentiation is arbitrary.

$$\frac{\partial^2 \dot{d}}{\partial S^2} = 0 \cdot \mathbf{A}^T\mathbf{A} \qquad \frac{\partial^2 \dot{d}}{\partial d \partial S} = -\frac{\partial D^d}{\partial d} \qquad \frac{\partial^2 \dot{d}}{\partial G \partial S} = 0 \cdot \mathbf{A} \tag{A.70}$$

$$\frac{\partial^2 \dot{d}}{\partial T \partial S} = -\frac{\partial D^d}{\partial T} \qquad \frac{\partial^2 \dot{d}}{\partial d^2} = \frac{\partial^2 C^d}{\partial d^2} - \frac{\partial^2 D^d}{\partial d^2} S_i \qquad \frac{\partial^2 \dot{d}}{\partial G \partial d} = \frac{\partial^2 C^d}{\partial G \partial d} \tag{A.71}$$

$$\frac{\partial^2 \dot{d}}{\partial T \partial d} = \frac{\partial^2 C^d}{\partial T \partial d} - \frac{\partial^2 D^d}{\partial T \partial d} S_i \qquad \frac{\partial^2 \dot{d}}{\partial G^2} = \frac{\partial^2 C^d}{\partial G^2} \qquad \frac{\partial^2 \dot{d}}{\partial T \partial G} = \frac{\partial^2 C^d}{\partial T \partial G} \tag{A.72}$$

Thus, this reveals 7 additional derivatives that has to be derived.

$$\frac{\partial^2 C^d}{\partial d^2} = \frac{1}{\eta}\left(\frac{\partial P_L}{\partial d}\frac{d}{2\psi} - d\frac{\partial \psi}{\partial d}\frac{P_L}{2\psi^2} + \frac{P_L}{2\psi}\right) \tag{A.73}$$

$$\frac{\partial^2 D^d}{\partial d^2} = \frac{1}{\eta}\left(-\frac{\partial \psi}{\partial d}\frac{1}{3\psi^2} - d\frac{\partial^2 \psi}{\partial d^2}\frac{1}{6\psi^2} + d\frac{\partial \psi}{\partial d}^2\frac{1}{3\psi^3}\right)\mathbf{A} \tag{A.74}$$

$$\frac{\partial^2 C^d}{\partial G \partial d} = \frac{1}{\eta}\left(\frac{1}{2\psi}\frac{\partial P_L}{\partial G} - \frac{\partial \psi}{\partial d}\frac{d}{2\psi^2}\frac{\partial P_L}{\partial G} + \frac{d}{2\psi^2}\frac{\partial^2 P_L}{\partial d \partial G}\right) \tag{A.75}$$

$$\frac{\partial^2 C^d}{\partial T \partial d} = \frac{1}{\eta}\left(\frac{\partial^2 P_L}{\partial d \partial T}\frac{d}{2\psi} - d\frac{\partial \psi}{\partial d}\frac{\partial P_L}{\partial T}\frac{1}{2\psi^2} + \frac{\partial P_L}{\partial T}\frac{1}{2\psi} - \frac{\partial \eta}{\partial T}\frac{1}{\eta}\left(\frac{\partial P_L}{\partial d}\frac{d}{2\psi} - d\frac{\partial \psi}{\partial d}\frac{P_L}{2\psi^2} + \frac{P_L}{2\psi}\right)\right) \tag{A.76}$$

$$\frac{\partial^2 D^d}{\partial T \partial d} = -\frac{\partial \eta}{\partial T}\frac{1}{\eta^2}\left(\frac{1}{6\psi} - \frac{\partial \psi}{\partial d}\frac{d}{6\psi^2}\right)\mathbf{A} \tag{A.77}$$

$$\frac{\partial^2 C^d}{\partial G^2} = \frac{d}{2\eta\psi}\frac{\partial^2 P_L}{\partial G^2} \tag{A.78}$$

$$\frac{\partial^2 C^d}{\partial T \partial G} = \frac{1}{\eta}\left(-\frac{\partial \eta}{\partial T}\frac{1}{\eta}\frac{d}{2\psi}\frac{\partial P_L}{\partial G} + \frac{d}{2\psi}\frac{\partial^2 P_L}{\partial G \partial T}\right) \tag{A.79}$$

Grain Size residual

The partial derivatives that needs to be derived are:

$$\frac{\partial^2 \dot{G}}{\partial S^2} \qquad \frac{\partial^2 \dot{G}}{\partial d \partial S} \qquad \frac{\partial^2 \dot{G}}{\partial G \partial S} \qquad \frac{\partial^2 \dot{G}}{\partial T \partial S} \tag{A.80}$$

$$\frac{\partial^2 \dot{G}}{\partial d^2} \qquad \frac{\partial^2 \dot{G}}{\partial G \partial d} \qquad \frac{\partial^2 \dot{G}}{\partial T \partial d} \qquad \frac{\partial^2 \dot{G}}{\partial G^2} \qquad \frac{\partial^2 \dot{G}}{\partial T \partial G} \tag{A.81}$$

Since the order of differentiation is arbitrary.

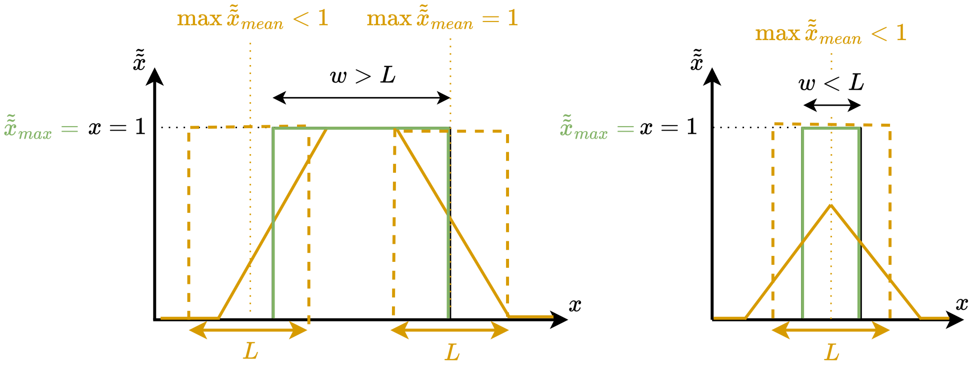
$$\frac{\partial^2 \dot{G}}{\partial S^2} = 0 \cdot \mathbf{A}^T\mathbf{A} \qquad \frac{\partial^2 \dot{d}}{\partial d \partial S} = 0 \cdot \mathbf{A}^T \qquad \frac{\partial^2 \dot{d}}{\partial G \partial S} 0 \cdot \mathbf{A}^T \tag{A.82}$$

$$\frac{\partial^2 \dot{d}}{\partial T \partial S} 0 \cdot \mathbf{A}^T \qquad \frac{\partial^2 \dot{G}}{\partial d^2} = 0 \qquad \frac{\partial^2 \dot{G}}{\partial G^2} = 2\frac{\dot{G}}{G^2} \tag{A.83}$$

$$\frac{\partial^2 \dot{G}}{\partial T \partial d} = 0 \qquad \frac{\partial^2 \dot{G}}{\partial G \partial d} = 0 \qquad \frac{\partial^2 \dot{G}}{\partial T \partial G} = -\frac{A_G\left(Q_G - T\frac{\partial Q_G}{\partial T}\right)}{G^2 R T^2}\exp\left(-\frac{Q_G}{RT}\right) \tag{A.84}$$

B Length scale constraint

In the present work, a similar approach to that of Li et al. [2023] was trialled, which imposes direct geometric constraints. Herein the minimum length scale is explicitly introduced by imposing a constraint on the maximum mean projected PD, $\tilde{\bar{x}}$, within the minimum explicitly imposed length scale L. In Fig. B.1, the filtered PDs from Fig. 6.3 are projected with $\eta = 0.5$ and $\beta \rightarrow \infty$, which effectively returns them to the unfiltered values (green). Then, the mean of the projected PDs inside the minimum length scale is calculated (yellow). As can be seen, for PDs inside the member, whose width is larger than the minimum length-scale (left), the maximum mean projected PD is equal to 1, while for PDs outside the member, it is less than 1. However, for the member whose width is smaller than the minimum length scale (right), the maximum mean projected PD is below 1 everywhere. Thus, a minimum length scale of L can be achieved by imposing a constraint in each element that the maximum mean projected PD within the minimum length scale of that element has to be equal to 1.



$\max \tilde{\bar{x}}_{mean} < 1$
$\max \tilde{\bar{x}}_{mean} = 1$
$\tilde{\bar{x}}$
$w > L$
$\tilde{\bar{x}}_{max} = x = 1$
x
L
L
$\max \tilde{\bar{x}}_{mean} < 1$
$\tilde{\bar{x}}$
$w < L$
$\tilde{\bar{x}}_{max} = x = 1$
x
L

Figure B.1. Member width larger than minimum length-scale constraint (left). Member width smaller than length-scale constraint (right).

If $L = 2r$, the neighboring elements from the linear density filter can be reused, and the mean can be formulated as follows.

$$\tilde{\bar{x}}^{mean}_{(e)} = \frac{\sum_{l \in N_{(e)}} V^{(l)} \tilde{\bar{x}}_{(l)}}{\sum_{l \in N_{(e)}} V^{(l)}} \tag{B.1}$$

This effectively constructs the filled yellow line in Fig. B.1. Then, the maximum of the mean has to be evaluated in each element, corresponding to checking the values of the full yellow line within the area of the dashed yellow line. Since the max-function is non-differentiable a P-norm is used. The expression used in Eq. (B.2) differs slightly from Li et al. [2023].

$$\tilde{\tilde{x}}^{mean,max}_{(e)} = \left(\sum_{l\in N_{(e)}} \left(\tilde{\tilde{x}}^{mean}_{(l)} \right)^{P_1} \right)^{\frac{1}{P_1}} \tag{B.2}$$

The P-norm overestimates the maximum mean value in each element, thus, it can attain values larger than unity. The overestimation is directly related to the value of P_1 and the number of neighboring elements in the set $N_{(e)}$. If ignoring potential differences in element volume, the maximum overestimation is given by $n^{1/P_1} - 1$, where n is the number of terms included in the summation. Using Eq. (B.2), a constraint can be formulated in each element.

$$g^{s}_{(e)} = \tilde{\tilde{x}}_{(e)} \left(1 - \tilde{\tilde{x}}^{mean,max}_{(e)} \right) \leq 0 \tag{B.3}$$

The constraints are multiplied by the projected PDs, $\tilde{\tilde{x}}_{(e)}$, since the constraint should be inactive if the element is already void. In Li et al. [2023], a P-norm is used to reduce the number of constraints to a single global constraint. However, a P-norm does not accurately approximate the largest constraint violation, if it also aggregates negative constraint values. The Kreisselmeier–Steinhauser function is used in this current work instead since it can aggregate negative and positive values, as given in Eq. (B.3). The Kreisselmeier-Steinhauser like the P-norm aggregation function overpredicts the maximum value for P-values lower than ∞.

$$g^{s} = \frac{1}{P_2} \log \left(\sum_{i=1}^{N_{elm}} \exp \left(P_2 \, g^{s}_{(e)} \right) \right) \leq \xi \tag{B.4}$$

g^{s} is the global solid region constraint and ξ is a relaxation parameter that can be used for, e.g., continuation (Sec. 6.7) to ensure smooth convergence of the constraint. As $P_2 \to \infty$ this returns the largest positive constraint value. This can further be extended to define a constraint on the minimum length scale of the void regions, g^{v}, as well, through an analogous approach by substituting the projected PDs, $\tilde{\tilde{x}}$, with $1 - \tilde{\tilde{x}}$. [Li et al., 2023]
The constraint may be overly restrictive to the problem if it is strictly enforced at all iterations. Thus, it may be beneficial to initialize ξ to the maximum constraint violation $\max(g^{s}_{(e)})$ and gradually decrease it. Aggregation functions, Eq. (B.2) and Eq. (B.4), are used to estimate the maximum values but often overestimate the maximum value unless high P-values are used which conversely makes the problem more non-linear and difficult to solve.

When topology-optimized designs are tested in practice they can suffer from undesirable geometric failure modes such as buckling as described in [Gadegaard and Thuesen, 2022]. The minimum length scale constraint discussed above might be used as a heuristic buckling constraint by imposing a user-specified minimum cross-sectional area on all members such that geometric failure is avoided without including explicit buckling constraints on the topology optimization.

B.1 Minimum length scale constraint

As the minimum length scale constraint is imposed on both the solid and void phases, the sensitivities of both must be calculated. Even though they are similar, there are some differences. The minimum length scale constraint equations were described in Eqs. (B.1) to (B.4).

Solid phase minimum length scale

Firstly, the solid phase minimum length scale sensitivities will be treated. Using the chain rule of differentiation, the total sensitivity of the constraint can be written as Eq. (B.5).

$$\frac{\partial g^s}{\partial \tilde{\tilde{x}}_{(e)}} = \sum_{i=1}^{n_{elm}} \frac{\partial g^s}{\partial g^s_{(i)}} \left(\frac{\partial g^s_{(i)}}{\partial \tilde{\tilde{x}}^{mean,max}_{(i)}} \sum_{j\in N_{(i)}} \frac{\partial \tilde{\tilde{x}}^{mean,max}_{(i)}}{\partial \tilde{\tilde{x}}^{mean}_{(j)}} \sum_{e\in N_{(j)}} \frac{\partial \tilde{\tilde{x}}^{mean}_{(j)}}{\partial \tilde{\tilde{x}}_{(e)}} + \frac{\partial g^s_{(i)}}{\partial \tilde{\tilde{x}}_{(e)}} \right) \tag{B.5}$$

The first term on the RHS of Eq. (B.5) is given by Eq. (B.6).

$$\frac{\partial g^s}{\partial g^s_{(i)}} = \frac{\exp(P_2 g^s_{(i)})}{\sum_{j=1}^{N_{elm}} \exp(P_2 g^s_{(j)})} \tag{B.6}$$

The first term inside the parenthesis of the RHS of Eq. (B.5) is given in Eq. (B.7).

$$\frac{\partial g^s_{(i)}}{\partial \tilde{\tilde{x}}^{mean,max}_{(i)}} = -\tilde{\tilde{x}}_{(j)} \tag{B.7}$$

The second term inside the parenthesis of the RHS of Eq. (B.5) is given in Eq. (B.8).

$$\frac{\partial \tilde{\tilde{x}}^{mean,max}_{(i)}}{\partial \tilde{\tilde{x}}^{mean}_{(j)}} = (\tilde{\tilde{x}}^{mean}_{(j)})^{P_1-1} \left(\sum_{s\in N_{(i)}} \left(\tilde{\tilde{x}}^{mean}_{(s)} \right)^{P_1} \right)^{(\frac{1}{P_1}-1)} \tag{B.8}$$

The third term inside the parenthesis of the RHS of Eq. (B.5) is given in Eq. (B.9).

$$\frac{\partial \tilde{\tilde{x}}^{mean}_{(j)}}{\partial \tilde{\tilde{x}}_{(e)}} = \frac{V^{(e)}}{\sum_{j\in N_{(i)}} V^{(j)}} \tag{B.9}$$

The final term inside the parenthesis of the RHS of Eq. (B.5) is given in Eq. (B.10).

$$\frac{\partial g^s_{(i)}}{\partial \tilde{\tilde{x}}_{(e)}} = \begin{cases} 1 - \tilde{\tilde{x}}^{mean,max}_{(e)}, & \text{if } i = e \\ 0, & \text{if } i \neq e \end{cases} \tag{B.10}$$

Void phase minimum length scale

Similar to the solid phase minimum length scale constraint, the sensitivities for the void phase length scale constraint have to be derived. The total derivative is the same for the solid and void phase, however in each term in Eqs. (B.1) to (B.4) that contains $\tilde{\tilde{x}}_{(e)}$ is substituted by $1 - \tilde{\tilde{x}}_{(e)}$ for the void phase constraint. Thus when compared to the solid phase length scale constraint, only the derivatives in Eq. (B.9) and Eq. (B.10) are different. In the following, if a v is used as a subscript or a '$, v$' is added to the existing subscript it refers to the void phase values. The total derivative can be seen in Eq. (B.11) and the two derivatives that are different from the solid phase length scale constraint are seen in Eq. (B.12)

$$\frac{\partial g^s}{\partial \tilde{\tilde{x}}_{(e)}} = \sum_{i=1}^{n_{elm}} \frac{\partial g^v}{\partial g^v_{(i)}} \left(\frac{\partial g^v_{(i)}}{\partial \tilde{\tilde{x}}^{mean,max,v}_{(i)}} \sum_{j\in N_{(i)}} \frac{\partial \tilde{\tilde{x}}^{mean,max,v}_{(i)}}{\partial \tilde{\tilde{x}}^{mean,v}_{(j)}} \sum_{e\in N_{(j)}} \frac{\partial \tilde{\tilde{x}}^{mean,v}_{(j)}}{\partial \tilde{\tilde{x}}_{(e)}} + \frac{\partial g^v_{(i)}}{\partial \tilde{\tilde{x}}_{(e)}} \right) \tag{B.11}$$

$$\frac{\partial \tilde{\tilde{x}}^{mean,v}_{(i)}}{\partial \tilde{\tilde{x}}_{(e)}} = -\frac{V^{(e)}}{\sum_{j\in N_{(i)}} V^{(j)}} \qquad \frac{\partial g^{v}_{(i)}}{\partial \tilde{\tilde{x}}_{(e)}} = \begin{cases} \tilde{\tilde{x}}^{mean,max,v}_{(e)} - 1, & \text{if } i = e \\ 0, & \text{if } i \neq e \end{cases} \tag{B.12}$$

B.1.1 Verification

In Fig. B.2 the relative error can be seen between the analytical and FDA sensitivities for the solid and void length scale constraints respectively. The constraint scaling factor has been omitted in the shown error plots.

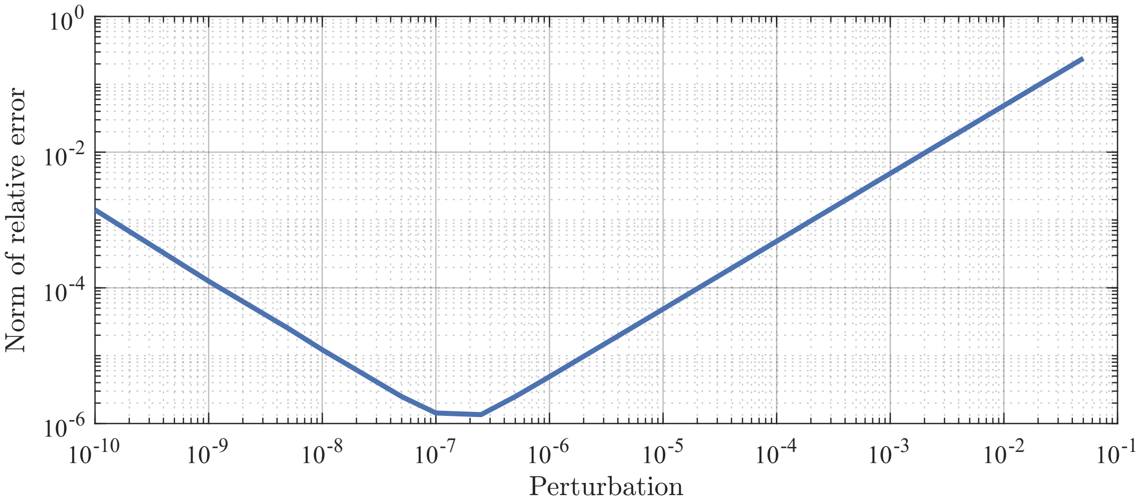


Figure B.2. Norm of the relative difference between analytical and FDA approximated sensitivities for solid phase length scale constraint.

C MMA optimizer

Within structural optimization a popular optimizer is the MMA optimizer [Svanberg, 1987] as already mentioned in Sec. 6.6. The notation used in this section is directly from [Svanberg, 1987] and does not refer to any previously used symbols. The MMA optimizer approximates the original optimization problem e.g. Eq. (6.1) with a strictly convex subproblem [Svanberg, 1987]. Due to the convex nature of the subproblem, it is easier to solve than the original problem. The convex subproblem can be stated as Eq. (C.1).

$$\text{minimize:} \qquad \tilde{f}_0^{(k)}(\{\mathbf{x}\}) + a_0 z + \sum_{i=1}^{m}\Big(c_i y_i + \frac{d_i}{2}y_i^2\Big) \tag{C.1a}$$

$$\text{subject to:} \qquad \tilde{f}_i^{(k)}(\{\mathbf{x}\}) - a_i z - y_i \leq 0, \quad i = 1, \ldots, m \tag{C.1b}$$

$$\alpha_j \leq x_j \leq \beta_j \tag{C.1c}$$

$$y_i \geq 0 \tag{C.1d}$$

$$z \geq 0 \tag{C.1e}$$

In Eq. (C.1a), $\tilde{f}_o^{(k)}(\{\mathbf{x}\})$ is the convex approximation of the objective function, z and $\{\mathbf{y}\}$ are 'artificial' optimization variables added to aid solving and formulating the optimization problem [Svanberg, ND]. a_0, c_i and d_i are all real-valued constants. In Eq. (C.1b), $\tilde{f}_i^{(k)}(\{\mathbf{x}\})$ are the convex continuously differentiable approximation functions of the constraints dependent on the design variables $\{\mathbf{x}\}$ at optimizer iteration k, m is the number of constraints imposed from the optimization formulation and a_i is also a real-valued constant. In Eq. (C.1c), α_j and β_j are the lower and upper move limit respectively, which are used to formulate the 'moving asymptotes' that are in turn used to formulate the approximating functions $\tilde{f}_i^{(k)}(\{\mathbf{x}\})$. Eqs. (C.1d) to (C.1e) are constraints on the values of the 'artifical' variables such that they can not aid in minimizing the objective function by becoming negative. The approximating function $\tilde{f}_i^{(k)}(\{\mathbf{x}\})$ is a first order approximation of the original function f_i and is defined as seen in Eq. (C.2) [Svanberg, 1987].

$$\tilde{f}_i^{(k)}(\{\mathbf{x}\}) = \sum_{j=1}^{(n)}\Big(\frac{p_{ij}^{(k)}}{u_j^{(k)} - x_j} + \frac{q_{ij}^{(k)}}{x_j - l_j^{(k)}}\Big) + r_i^{(k)}, i = 0, 1, \ldots, m \tag{C.2}$$

In Eq. (C.2), n is the number of design variables, $u_j^{(k)}$ is the upper moving asymptote, $l_j^{(k)}$ is the lower moving asymptote and the two functions $p_{ij}^{(k)}$ and $q_{ij}^{(k)}$ are functions of the design variables, the moving asymptotes and the gradients of the original non-approximated problem and $r_i^{(k)}$ is a residual that describes the error between the original function $f_i(\{\mathbf{x}\})$ and the approximated function. Due to the manner in which $p_{ij}^{(k)}$ and $q_{ij}^{(k)}$ are updated, the original MMA optimizer can only make monotonous approximations. For further information and derivation of the MMA optimizer cf. [Svanberg, 1987] [Svanberg, ND].

As described in Sec. 6.6, the compliance minimization problem can no longer be ensured to be monotonous when including design dependent loading and the original MMA optimizer might incur oscillatory behavior during optimization [Bruyneel and Duysinx, 2005]. Instead, the Globally Convergent Method of Moving Asymptotes (GCMMA) can be used. The GCMMA optimizer can converge to a solution from any initial design guess. According to [Bruyneel and Duysinx, 2005], the GCMMA optimizer uses non-monotonous approximations to the original problem. However, these benefits comes with increased computational time. As noted in Sec. 6.6 initial testing with the MMA optimizer did not suffer from oscillatory behavior during optimization with self-weight.

D Adjoint tests

To verify the implementation and formulation of the adjoint sensitivity analysis it is verified by the implementation of a simple linear material. For this purpose, the simple benchmark geometry of Sec. 5.3 will be readopted. However, another element has been added in the lengthwise direction.
Modified SIMP has been used to penalize the modulus of elasticity with $x_{min} = 1e - 3$ and $p = 3$. The linear elastic material is used with $E = 200\,\mathrm{GPA}$ and $\nu = 0.3$. A simple surrogate objective function has been used, which is formulated as in Eq. (D.1).

$$f = \frac{1}{2}\sum_{i=1}^{n}\{\mathbf{U}\}^{(i)T}\{\mathbf{U}\}^{(i)} \tag{D.1}$$

This has been used simply because it is directly given during the solution of the primal problem and since it depends on each time step (i). This could equally as well be done for compliance. The resulting sensitivity, $\frac{df}{d\{\mathbf{x}\}}$ is compared to the forward difference approximation, $\frac{df}{d\{\mathbf{x}\}}_{\text{fda}}$, which is calculated as:

$$\frac{df}{dx_{(i)}}_{\text{fda}} = \frac{f(x_{(i)} + \Delta x) - f(x_{(i)})}{\Delta x} \tag{D.2}$$

For each sensitivity of the variable $x_{(i)}$, keeping all other entries in $\{\mathbf{x}\}$ constant, and is then assembled into the vector of sensitivities $\frac{df}{d\{\mathbf{x}\}}_{\text{fda}}$ The error between the forward difference approximation and analytical sensitivity is calculated as:

$$\text{error} = \left\| \frac{\frac{df}{d\{\mathbf{x}\}}_{\text{fda}} - \frac{df}{d\{\mathbf{x}\}}}{\frac{df}{d\{\mathbf{x}\}}} \right\|_2 \tag{D.3}$$

Which is compared by plotting the error for increasing small perturbations Δx

D.1 Non-incremental stress-update

Stress is calculated purely based on current displacements, which means $\frac{\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i-1)}} = [\mathbf{0}]$. Since the material is linearly elastic, that means that there are no inner residuals, and these terms vanish. Therefore, the adjoint formulation is given as:

$$\hat{f} = f + \sum_{i=1}^{n}\{\boldsymbol{\lambda}\}^{(i)^T}\{\mathbf{R}\}^{(i)} \tag{D.4}$$

$$\frac{d\hat{f}}{d\{\mathbf{x}\}} = \frac{\partial f}{\partial\{\mathbf{x}\}} + \sum_{i=1}^{n}\left(\frac{\partial f}{\partial\{\mathbf{u}\}^{(i)}}\frac{d\{\mathbf{u}\}^{(i)}}{d\{\mathbf{x}\}} + \{\boldsymbol{\lambda}\}^{(i)T}\left(\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{x}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i)}}\frac{d\{\mathbf{u}\}^{(i)}}{d\{\mathbf{x}\}}\right)\right) \tag{D.5}$$

There are no explicit dependencies of f on $\{\mathbf{x}\}$. The formulation is, therefore, simplified into the explicit and adjoint equation.

$$\frac{d\hat{f}}{d\{\mathbf{x}\}} = \sum_{i=1}^{n} \{\boldsymbol{\lambda}\}^{(i)T} \frac{\partial \{\mathbf{R}\}^{(i)}}{\partial \{\mathbf{x}\}} \tag{D.6}$$

$$\frac{\partial f}{\partial \{\mathbf{u}\}^{(i)}} + \{\boldsymbol{\lambda}\}^{(i)T} \frac{\partial \{\mathbf{R}\}^{(i)}}{\partial \{\mathbf{u}\}^{(i)}} = \{\mathbf{0}\} \tag{D.7}$$

This could also be solved simultaneously with the analysis since there is no terminal condition. The result is given in Fig. D.1.

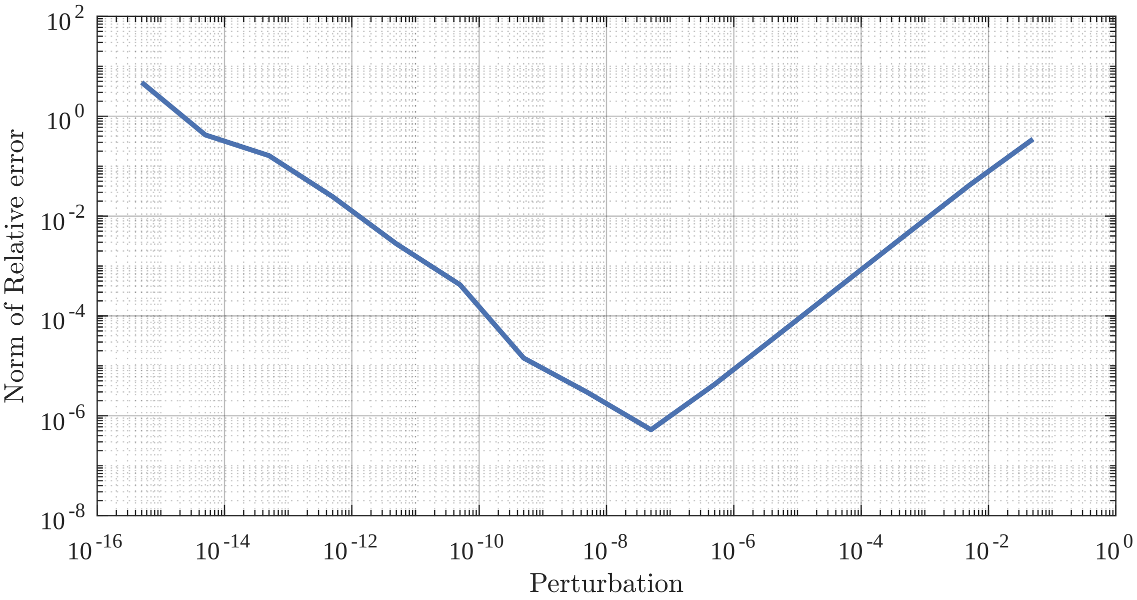


Figure D.1. Norm of relative error for adjoint sensitivities.

D.2 Incremental stress-update

The stress is now calculated in the form:

$$\{\mathbf{S}\}^{(i)} = \{\mathbf{S}\}^{(i-1)} + [\mathbf{D}]\,\{\boldsymbol{\Delta\varepsilon}\} \tag{D.8}$$

This yields primal analysis results identical to those of the previous one. However, it changes the adjoint formulation, as the stress is saved in the integration points; it effectively includes an internal residual $\{\mathbf{h}\}$, defined at each integration point. These are collected in a global vector $\{\mathbf{H}\}$ as defined in Eq. (D.9).

$$\{\mathbf{H}\}^{(i)} = \left\{ \{\mathbf{h}\}^{(i)}_{(1)} \quad \cdots \quad \{\mathbf{h}\}^{(i)}_{(n_{elm})} \right\}^T \tag{D.9}$$

$$\{\mathbf{h}\}^{(i)}_{(e)} = \left\{ \{\mathbf{h}\}^{(i)}_{(e,1)} \quad \cdots \quad \{\mathbf{h}\}^{(i)}_{(e,n_{ipts})} \right\}^T \tag{D.10}$$

$$\{\mathbf{h}\}^{(i)}_{(e,k)} = \{\mathbf{S}\}^{(i)}_{(e,k)} - \{\mathbf{S}\}^{(i-1)}_{(e,k)} - [\mathbf{D}]\,\{\boldsymbol{\Delta\varepsilon}\}^{(i)}_{(e,k)} = \{\mathbf{0}\} \tag{D.11}$$

The local variables are likewise defined at each integration point and collected in a global vector, as defined in Eq. (D.12).

$$\{\mathbf{c}\}^{(i)} = \Big\{\{\mathbf{c}\}^{(i)}_{(1)} \quad ... \quad \{\mathbf{c}\}^{(i)}_{(n_{elm})}\Big\}^T \tag{D.12}$$

$$\{\mathbf{c}\}^{(i)}_{(e)} = \Big\{\{\mathbf{c}\}^{(i)}_{(e,1)} \quad ... \quad \{\mathbf{c}\}^{(i)}_{(e,n_{ipts})}\Big\}^T \tag{D.13}$$

$$\{\mathbf{c}\}^{(i)}_{(e,k)} = \{\mathbf{S}\}^{(i)}_{(e,k)}{}^T \tag{D.14}$$

The residual is now also a function of both the current and previous configuration, through the strain increment $\{\boldsymbol{\Delta\varepsilon}\}$, denoted by (i) and $(i-1)$, respectively. Thus, no terms can be neglected in the sensitivity analysis. The Lagrangian is given as

$$\hat{f} = f + \sum_{i=1}^{n} \{\boldsymbol{\lambda}\}^{(i)^T}\{\mathbf{R}\}^{(i)} + \sum_{i=1}^{n} \{\boldsymbol{\mu}\}^{(i)^T}\{\mathbf{H}\}^{(i)} \tag{D.15}$$

The sensitivity is given as in Eq. (D.16).

$$\begin{aligned}
\frac{d\hat{f}}{d\{\mathbf{x}\}} &= \sum_{i=1}^{n}\left(\frac{\partial f}{\partial\{\mathbf{u}\}^{(i)}}\frac{d\{\mathbf{u}\}^{(i)}}{d\{\mathbf{x}\}} + \frac{\partial f}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\mathbf{x}\}}\right)\\
&+ \sum_{i=1}^{n}\{\boldsymbol{\lambda}\}^{(i)T}\left(\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{x}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i)}}\frac{d\{\mathbf{u}\}^{(i)}}{d\{\mathbf{x}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i-1)}}\frac{d\{\mathbf{u}\}^{(i-1)}}{d\{\mathbf{x}\}}\right.\\
&\qquad\left. + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\mathbf{x}\}} + \frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}}\frac{d\{\mathbf{c}\}^{(i-1)}}{d\{\mathbf{x}\}}\right)\\
&+ \sum_{i=1}^{n}\{\boldsymbol{\mu}\}^{(i)T}\left(\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{x}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i)}}\frac{d\{\mathbf{u}\}^{(i)}}{d\{\mathbf{x}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i-1)}}\frac{d\{\mathbf{u}\}^{(i-1)}}{d\{\mathbf{x}\}}\right.\\
&\qquad\left. + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}}\frac{d\{\mathbf{c}\}^{(i)}}{d\{\mathbf{x}\}} + \frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i-1)}}\frac{d\{\mathbf{c}\}^{(i-1)}}{d\{\mathbf{x}\}}\right)
\end{aligned} \tag{D.16}$$

By factoring out the common factors, the implicit derivatives, and setting up the adjoint equations, Eqs. (D.17) to (D.19) can be formed.

$$\frac{d\hat{f}}{d\{\mathbf{x}\}} = \sum_{i=1}^{n} \{\boldsymbol{\lambda}\}^{(i)T}\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{x}\}} + \{\boldsymbol{\mu}\}^{(i)T}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{x}\}} \tag{D.17}$$

$$\begin{aligned}
&\frac{\partial f}{\partial\{\mathbf{u}\}^{(i)}} + \{\boldsymbol{\lambda}\}^{(i)T}\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i)}} + \{\boldsymbol{\mu}\}^{(i)T}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{u}\}^{(i)}} = \{\mathbf{0}\} \\
&\frac{\partial f}{\partial\{\mathbf{c}\}^{(i)}} + \{\boldsymbol{\lambda}\}^{(i)T}\frac{\partial\{\mathbf{R}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}} + \{\boldsymbol{\mu}\}^{(i)T}\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{c}\}^{(i)}} = \{\mathbf{0}\}
\end{aligned} \qquad \text{for } i = n \tag{D.18}$$

$$\begin{aligned}
&\frac{\partial f}{\partial\{\mathbf{u}\}^{(k)}} + \{\boldsymbol{\lambda}\}^{(k)T}\frac{\partial\{\mathbf{R}\}^{(k)}}{\partial\{\mathbf{u}\}^{(k)}} + \{\boldsymbol{\mu}\}^{(k)T}\frac{\partial\{\mathbf{H}\}^{(k)}}{\partial\{\mathbf{u}\}^{(k)}} \\
&+ \{\boldsymbol{\lambda}\}^{(k+1)T}\frac{\partial\{\mathbf{R}\}^{(k+1)}}{\partial\{\mathbf{u}\}^{(k)}} + \{\boldsymbol{\mu}\}^{(k+1)T}\frac{\partial\{\mathbf{H}\}^{(k+1)}}{\partial\{\mathbf{u}\}^{(k)}} = \{\mathbf{0}\} \\
&\frac{\partial f}{\partial\{\mathbf{c}\}^{(k)}} + \{\boldsymbol{\lambda}\}^{(k)T}\frac{\partial\{\mathbf{R}\}^{(k)}}{\partial\{\mathbf{c}\}^{(k)}} + \{\boldsymbol{\mu}\}^{(k)T}\frac{\partial\{\mathbf{H}\}^{(k)}}{\partial\{\mathbf{c}\}^{(k)}} \\
&+ \{\boldsymbol{\lambda}\}^{(k+1)T}\frac{\partial\{\mathbf{R}\}^{(k+1)}}{\partial\{\mathbf{c}\}^{(k)}} + \{\boldsymbol{\mu}\}^{(k+1)T}\frac{\partial\{\mathbf{H}\}^{(k+1)}}{\partial\{\mathbf{c}\}^{(k)}} = \{\mathbf{0}\}
\end{aligned} \qquad \text{for } k = n-1, ...1 \tag{D.19}$$

Arranging in matrix-vector format for solution. The terminal condition is solved as:

$$\begin{bmatrix} \left(\frac{\partial\{\mathbf{R}\}^{(n)}}{\partial\{\mathbf{u}\}^{(n)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(n)}}{\partial\{\mathbf{u}\}^{(n)}}\right)^T \\ \left(\frac{\partial\{\mathbf{R}\}^{(n)}}{\partial\{\mathbf{c}\}^{(n)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(n)}}{\partial\{\mathbf{c}\}^{(n)}}\right)^T \end{bmatrix} \begin{Bmatrix} \{\boldsymbol{\lambda}\}^{(n)} \\ \{\boldsymbol{\mu}\}^{(n)} \end{Bmatrix} = - \begin{Bmatrix} \left(\frac{\partial f}{\partial\{\mathbf{u}\}^{(n)}}\right)^T \\ \left(\frac{\partial f}{\partial\{\mathbf{c}\}^{(n)}}\right)^T \end{Bmatrix} \tag{D.20}$$

And the rest of the solution steps can then be back-stepped through.

$$\begin{aligned}
\begin{bmatrix} \left(\frac{\partial\{\mathbf{R}\}^{(k)}}{\partial\{\mathbf{u}\}^{(k)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(k)}}{\partial\{\mathbf{u}\}^{(k)}}\right)^T \\ \left(\frac{\partial\{\mathbf{R}\}^{(k)}}{\partial\{\mathbf{c}\}^{(k)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(k)}}{\partial\{\mathbf{c}\}^{(k)}}\right)^T \end{bmatrix} \begin{Bmatrix} \{\boldsymbol{\lambda}\}^{(k)} \\ \{\boldsymbol{\mu}\}^{(k)} \end{Bmatrix} = &- \begin{Bmatrix} \left(\frac{\partial f}{\partial\{\mathbf{u}\}^{(k)}}\right)^T \\ \left(\frac{\partial f}{\partial\{\mathbf{c}\}^{(k)}}\right)^T \end{Bmatrix} \\
&- \begin{bmatrix} \left(\frac{\partial\{\mathbf{R}\}^{(k+1)}}{\partial\{\mathbf{u}\}^{(k)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(k+1)}}{\partial\{\mathbf{u}\}^{(k)}}\right)^T \\ \left(\frac{\partial\{\mathbf{R}\}^{(k+1)}}{\partial\{\mathbf{c}\}^{(k)}}\right)^T & \left(\frac{\partial\{\mathbf{H}\}^{(k+1)}}{\partial\{\mathbf{c}\}^{(k)}}\right)^T \end{bmatrix} \begin{Bmatrix} \{\boldsymbol{\lambda}\}^{(k+1)} \\ \{\boldsymbol{\mu}\}^{(k+1)} \end{Bmatrix}
\end{aligned} \tag{D.21}$$

To obtain a better conditioned matrix inversion, all local constraints are scaled by their corresponding integration weight factor. The result is given in Fig. D.2.

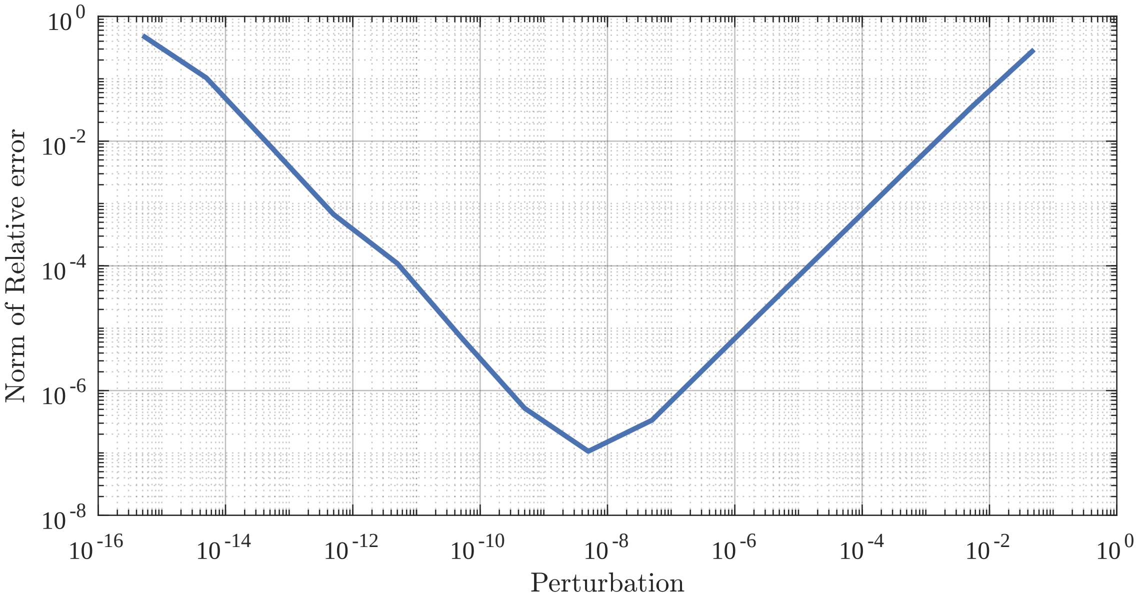


Figure D.2. Norm of relative error for adjoint sensitivities.

E FEA derivatives

During the design sensitivity analysis, it is necessary to formulate the derivatives of the global and local residual wrt. the global variables. These are more specifically derivatives concerning the FEA formulation. Firstly, the derivative of the local residual w.r.t. global variables is given in Appendix E.1 while the derivative of the global residual. w.r.t. local variables is given in Appendix E.2.

E.1 Strain residual derivative

In Eq. (7.32), the derivative of the local constraints wrt. to the global variables (nodal displacements) must be derived. The only local constraint that is explicitly a function of the nodal displacements, whilst keeping the stress constant, is $\{\mathbf{h}_{\boldsymbol{\varepsilon}}\}$. More specifically only the strain increment in Eq. (7.19) is a function of the nodal displacements. Therefore, taking the derivative $\frac{\partial\{\mathbf{H}\}^{(i)}}{\partial\{\mathbf{U}\}^{(i-1)}}$ can be rewritten as $\frac{\partial\{\boldsymbol{\Delta\varepsilon}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e,k)}}$. The strain increment can further be described as $\{\boldsymbol{\Delta\varepsilon}\}^{(i)}_{(e,k)} = \{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} - \{\boldsymbol{\varepsilon}\}^{(i-1)}_{(e,k)}$. The derivation will only be performed for the current strain, but is equally valid for the previous strain.

In [Crisfield, 1991], the strains are decomposed into a linear and non-linear part, see Eq. (E.1).

$$\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} = \{\boldsymbol{\varepsilon}_{\mathbf{L}}\}^{(i)}_{(e,k)} + \{\boldsymbol{\varepsilon}_{\mathbf{NL}}\}^{(i)}_{(e,k)} \tag{E.1}$$

However, in [Crisfield, 1991] a single strain-displacement matrix is used to calculate the total strain, see Eq. (E.2).

$$\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} = \left([\mathbf{H}] + \frac{1}{2}\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}\right)\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} \tag{E.2}$$

In the remainder of this derivation, it is assumed a 2D quadrilateral element with only 4 nodes is used for conciseness. The derivation can be expanded to 3D elements but will become more tedious.

$[\mathbf{H}]$ in Eq. (E.2) is a Boolean matrix, see Eq. (E.3)

$$[\mathbf{H}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{E.3}$$

$[\mathbf{A}(\boldsymbol{\theta})]$ in Eq. (E.2) contains the displacement derivatives, see Eq. (E.4)

$$[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)} = \begin{bmatrix} \frac{\partial u}{\partial x} & 0 & \frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial u}{\partial y} & 0 & \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} & \frac{\partial u}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial x} \end{bmatrix} \tag{E.4}$$

Finally, the $\{\boldsymbol{\theta}\}$ vector also contains the displacement derivatives but organized in a vector, see Eq. (E.5)

$$\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{bmatrix} \tag{E.5}$$

Further, the displacement derivatives can be described by the nodal displacements $\{\mathbf{a_1 \ldots a_8}\}$ and the derivatives of the shape functions, see Eq. (E.6).

$$\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} = [\mathbf{G}]_{(e,k)} \{\mathbf{a}\}^{(i)}_{(e)} = \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{Bmatrix} \tag{E.6}$$

The entries in the nodal displacement vector are organized such that a_1 and a_2 are the x- and y-displacement components (or u_1 and v_1) for the first node, a_3 and a_4 are the x- and y-displacement components for the second node etc. The comma notation used in $[\mathbf{G}]_{(e,k)}$, is simply the derivative such that $N_{1,x}$ is the first shape function differentiated w.r.t. the x material coordinate.

Now, the strain displacement relation from Eq. (E.2) can be modified using Eq. (E.6), see Eq. (E.7).

$$\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)} = \left([\mathbf{H}] [\mathbf{G}]_{(e,k)} + \frac{1}{2} [\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)} [\mathbf{G}]_{(e,k)} \right) \{\mathbf{a}\}^{(i)}_{(e)} \tag{E.7}$$

The derivative of the strain expression in Eq. (E.7) can be differentiated w.r.t. the nodal displacements. By using the product rule and realizing that only $\{\mathbf{a}\}^{(i)}_{(e)}$ and $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ yield non-zero derivatives wrt. the nodal displacements, Eq. (E.8) can be formulated.

$$\frac{\partial\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}} = \left([\mathbf{H}]\,[\mathbf{G}]_{(e,k)} + \frac{1}{2}\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}\,[\mathbf{G}]_{(e,k)}\right) + \frac{1}{2}\left(\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\,[\mathbf{G}]_{(e,k)}\right)\{\mathbf{a}\}^{(i)}_{(e)} \quad \text{(E.8)}$$

From Eq. (E.8) it can immediately be recognized that the derivative in the second term will yield a 3D matrix. By expanding the matrix-vector product from Eq. (E.6), the $\frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}$ term becomes more clear.

$$\{\boldsymbol{\theta}\} = \begin{Bmatrix} N_{1,x}a_1 + N_{2,x}a_3 + N_{3,x}a_5 + N_{4,x}a_7 \\ N_{1,y}a_1 + N_{2,y}a_3 + N_{3,y}a_5 + N_{4,y}a_7 \\ N_{1,x}a_2 + N_{2,x}a_4 + N_{3,x}a_6 + N_{4,x}a_8 \\ N_{1,y}a_2 + N_{2,y}a_4 + N_{3,y}a_6 + N_{4,y}a_8 \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{Bmatrix} \quad \text{(E.9)}$$

When taking the derivative $\frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}$, it will be done for each entry in the nodal displacement vector $\{\mathbf{a}\}^{(i)}_{(e)}$. This produces a set of 2D matrices containing the derivatives of $\frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_1}$ to $\frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_8}$.

Each of these derivatives are written as seen in Eqs. (E.10) to (E.17).

$$\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_1} = \begin{bmatrix} \frac{\partial}{\partial a_1}\frac{\partial u}{\partial x} & 0 & \frac{\partial}{\partial a_1}\frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial a_1}\frac{\partial u}{\partial y} & 0 & \frac{\partial}{\partial a_1}\frac{\partial v}{\partial y} \\ \frac{\partial}{\partial a_1}\frac{\partial u}{\partial y} & \frac{\partial}{\partial a_1}\frac{\partial u}{\partial x} & \frac{\partial}{\partial a_1}\frac{\partial v}{\partial y} & \frac{\partial}{\partial a_1}\frac{\partial v}{\partial x} \end{bmatrix} = \begin{bmatrix} N_{1,x} & 0 & 0 & 0 \\ 0 & N_{1,y} & 0 & 0 \\ N_{1,y} & N_{1,x} & 0 & 0 \end{bmatrix} \quad \text{(E.10)}$$

$$\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_2} = \begin{bmatrix} \frac{\partial}{\partial a_2}\frac{\partial u}{\partial x} & 0 & \frac{\partial}{\partial a_2}\frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial a_2}\frac{\partial u}{\partial y} & 0 & \frac{\partial}{\partial a_2}\frac{\partial v}{\partial y} \\ \frac{\partial}{\partial a_2}\frac{\partial u}{\partial y} & \frac{\partial}{\partial a_2}\frac{\partial u}{\partial x} & \frac{\partial}{\partial a_2}\frac{\partial v}{\partial y} & \frac{\partial}{\partial a_2}\frac{\partial v}{\partial x} \end{bmatrix} = \begin{bmatrix} 0 & 0 & N_{1,x} & 0 \\ 0 & 0 & 0 & N_{1,y} \\ 0 & 0 & N_{1,y} & N_{1,x} \end{bmatrix} \quad \text{(E.11)}$$

$$\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_3} = \begin{bmatrix} N_{2,x} & 0 & 0 & 0 \\ 0 & N_{2,y} & 0 & 0 \\ N_{2,y} & N_{2,x} & 0 & 0 \end{bmatrix} \quad \text{(E.12)}$$

$$\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_4} = \begin{bmatrix} 0 & 0 & N_{2,x} & 0 \\ 0 & 0 & 0 & N_{2,y} \\ 0 & 0 & N_{2,y} & N_{2,x} \end{bmatrix} \quad \text{(E.13)}$$

$$\frac{\partial\,[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_5} = \begin{bmatrix} N_{3,x} & 0 & 0 & 0 \\ 0 & N_{3,y} & 0 & 0 \\ N_{3,y} & N_{3,x} & 0 & 0 \end{bmatrix} \quad \text{(E.14)}$$

$$\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial a_6} = \begin{bmatrix} 0 & 0 & N_{3,x} & 0 \\ 0 & 0 & 0 & N_{3,y} \\ 0 & 0 & N_{3,y} & N_{3,x} \end{bmatrix} \tag{E.15}$$

$$\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial a_7} = \begin{bmatrix} N_{4,x} & 0 & 0 & 0 \\ 0 & N_{4,y} & 0 & 0 \\ N_{4,y} & N_{4,x} & 0 & 0 \end{bmatrix} \tag{E.16}$$

$$\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial a_8} = \begin{bmatrix} 0 & 0 & N_{4,x} & 0 \\ 0 & 0 & 0 & N_{4,y} \\ 0 & 0 & N_{4,y} & N_{4,x} \end{bmatrix} \tag{E.17}$$

Now that the derivatives of $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ have been determined, they have to be collected into a single matrix. If the product between a 3D matrix and a vector is considered, it is assumed that the products between each 2D slice of the 3D matrix and a vector will return a vector. Returning to Eq. (E.8), each of these vectors can be concatenated into a 2D matrix, see Eq. (E.18).

$$\frac{1}{2}\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} = \left[\frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_1}\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} \quad \cdots \quad \frac{\partial[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}}{\partial a_8}\{\boldsymbol{\theta}\}^{(i)}_{(e,k)}\right] \tag{E.18}$$

The contents of the matrix in Eq. (E.18) can be expanded as seen in Eq. (E.19).

$$\frac{1}{2}\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} = \\ \frac{1}{2}\begin{bmatrix} \frac{\partial u}{\partial x}N_{1,x} & \frac{\partial v}{\partial x}N_{1,x} & \cdots & \frac{\partial u}{\partial x}N_{4,x} & \frac{\partial v}{\partial x}N_{4,x} \\ \frac{\partial u}{\partial y}N_{1,y} & \frac{\partial v}{\partial y}N_{1,y} & \cdots & \frac{\partial u}{\partial y}N_{1,y} & \frac{\partial v}{\partial y}N_{1,y} \\ \frac{\partial u}{\partial y}N_{1,x} + \frac{\partial u}{\partial x}N_{1,y} & \frac{\partial v}{\partial y}N_{1,x} + \frac{\partial v}{\partial x}N_{1,y} & \cdots & \frac{\partial u}{\partial y}N_{4,x} + \frac{\partial u}{\partial x}N_{4,y} & \frac{\partial v}{\partial y}N_{4,x} + \frac{\partial v}{\partial x}N_{4,y} \end{bmatrix} \tag{E.19}$$

Coincidentally, the contents in matrix in Eq. (E.19) are equal to the product of $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ and $[\mathbf{G}]_{(e,k)}$, see also Eq. (E.20).

$$[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}\,[\mathbf{G}]_{(e,k)} = \\ \begin{bmatrix} \frac{\partial u}{\partial x} & 0 & \frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial u}{\partial y} & 0 & \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} & \frac{\partial u}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial x} \end{bmatrix} \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \tag{E.20}$$

Returning to Eq. (E.19), this expression can now be rewritten as seen in Eq. (E.21):

$$\frac{1}{2}\frac{\partial\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\{\boldsymbol{\theta}\}^{(i)}_{(e,k)}=\frac{1}{2}\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\left[\mathbf{G}\right]_{(e,k)} \tag{E.21}$$

Considering Eq. (E.21) and returning to Eq. (E.8) it can be reformulated as seen in Eq. (E.22).

$$\frac{\partial\{\boldsymbol{\varepsilon}\}^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}=\left[\mathbf{H}\right]\left[\mathbf{G}\right]_{(e,k)}+\frac{1}{2}\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\left[\mathbf{G}\right]_{(e,k)}+\frac{1}{2}\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\left[\mathbf{G}\right]_{(e,k)} \tag{E.22}$$

$$=\left(\left[\mathbf{H}\right]+\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\right)\left[\mathbf{G}\right]_{(e,k)}=\left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)} \tag{E.23}$$

Thus, the derivative of the local variables w.r.t. the current and previous displacements have been found.

E.2 Force residual derivative

The derivative of the global residual w.r.t. the global variables have to be derived in Eq. (7.30). The force residual on element level, internal force on element level and external force on element level are given in Eqs. (7.14) to (7.16) but are restated here for convenience in Eqs. (E.24) to (E.26).

$$\{\mathbf{r}\}^{(i)}_{(e)}=\{\mathbf{f_{int}}\}^{(i)}_{(e)}-\{\mathbf{f_{ext}}\}_{(e)} \tag{E.24}$$

$$\{\mathbf{f_{ext}}\}_{(e)}=\sum_{k=1}^{n_{ipts}}\tilde{\tilde{x}}_{(e)}d_0\rho\left[\mathbf{N}\right]^T_{(e,k)}\{\mathbf{g}\}w_{(e,k)} \tag{E.25}$$

$$\{\mathbf{f_{int}}\}^{(i)}_{(e)}=\sum_{k=1}^{n_{ipts}}v_{(e)}\left(\left[\mathbf{B_{NL}}\right]^{(i)}_{(e.k)}\right)^T\{\boldsymbol{\tau}\}^{(i)}_{(e,k)}w_{(e,k)} \tag{E.26}$$

$$=\sum_{k=1}^{n_{ipts}}v_{(e)}\left(\left(\left[\mathbf{H}\right]+\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\right)\left[\mathbf{G}\right]_{(e,k)}\right)^T\{\mathbf{S}\}^{(i)}_{(e,k)}w_{(e,k)} \tag{E.27}$$

The matrices in Eq. (E.27) are as given in Appendix E.1. When taking the derivative of the force residual on element level w.r.t. the nodal displacements on element level, only the term related to the internal force yields a non-zero contribution. Thus, only $\frac{\partial\{\mathbf{f_{int}}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}$ is of interest.

Now, the derivative of the force residual and therefore the internal force vector can be taken, as seen in Eq. (E.28).

$$\frac{\partial\{\mathbf{r}\}^{(i)}_{(e)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}=\frac{\partial}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\left(\sum_{k=1}^{n_{ipts}}v_{(e)}\left(\left(\left[\mathbf{H}\right]+\left[\mathbf{A}(\boldsymbol{\theta})\right]^{(i)}_{(e,k)}\right)\left[\mathbf{G}\right]_{(e,k)}\right)^T\{\mathbf{S}\}^{(i)}_{(e,k)}w_{(e,k)}\right) \tag{E.28}$$

As we are only differentiating w.r.t. explicit dependencies of the nodal displacement and the stress has to be kept constant as it is a local variable, it is only the term containing $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ and $[\mathbf{G}]_{(e,k)}$ that has to be differentiated.

Firstly, if the terms containing $[\mathbf{H}]$, $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ and $[\mathbf{G}]_{(e,k)}$ are expanded, it yields Eq. (E.29) where $[\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}$ and $[\mathbf{H}]$ have already been added together, which in this specific case is no different than multiplying each term in the parenthesis first.

$$\left([\mathbf{H}] + [\mathbf{A}(\boldsymbol{\theta})]^{(i)}_{(e,k)}\right)[\mathbf{G}]_{(e,k)} = [\mathbf{B_{NL}}]^{(i)}_{(e,k)} \tag{E.29}$$

$$= \begin{bmatrix} \frac{\partial u}{\partial x}+1 & 0 & \frac{\partial v}{\partial x} & 0 \\ 0 & \frac{\partial u}{\partial y} & 0 & \frac{\partial v}{\partial y+1} \\ \frac{\partial u}{\partial y} & \frac{\partial u}{\partial x}+1 & \frac{\partial v}{\partial y}+1 & \frac{\partial v}{\partial x} \end{bmatrix} \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \tag{E.30}$$

$$= \begin{bmatrix} N_{1,x}(\frac{\partial u}{\partial x}+1) & N_{1,x}\frac{\partial v}{\partial x} & \dots & \dots & N_{4,x}(\frac{\partial u}{\partial x}+1) & N_{4,x}\frac{\partial v}{\partial x} \\ N_{1,y}\frac{\partial u}{\partial y} & N_{1,y}(\frac{\partial v}{\partial y}+1) & \dots & \dots & N_{4,y}\frac{\partial u}{\partial y} & N_{4,y}(\frac{\partial v}{\partial y}+1) \\ N_{1,y}(\frac{\partial u}{\partial x}+1)+N_{1,x}\frac{\partial u}{\partial y} & N_{1,x}(\frac{\partial v}{\partial y}+1)+N_{1,y}\frac{\partial v}{\partial x} & \dots & \dots & N_{4,y}(\frac{\partial u}{\partial x}+1)+N_{4,x}\frac{\partial u}{\partial y} & N_{4,x}(\frac{\partial v}{\partial y}+1)+N_{4,y}\frac{\partial v}{\partial x} \end{bmatrix} \tag{E.31}$$

Now that the matrix has been expanded it can be recognized that in all entries at least one term of the displacement derivative terms recur. In Appendix E.1 the gradient deformation components were written as Eq. (E.32).

$$\{\boldsymbol{\theta}\}^{(i)}_{(e,k)} = \begin{Bmatrix} N_{1,x}a_1 + N_{2,x}a_3 + N_{3,x}a_5 + N_{4,x}a_7 \\ N_{1,y}a_1 + N_{2,y}a_3 + N_{3,y}a_5 + N_{4,y}a_7 \\ N_{1,x}a_2 + N_{2,x}a_4 + N_{3,x}a_6 + N_{4,x}a_8 \\ N_{1,y}a_2 + N_{2,y}a_4 + N_{3,y}a_6 + N_{4,y}a_8 \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{Bmatrix} \tag{E.32}$$

Utilizing the same approach as in Appendix E.1, the derivative of Eq. (E.31) w.r.t the vector nodal displacements will become a 3D matrix. In Appendix E.1 the approach was used in Eq. (E.18). Instead, Eq. (E.31) will be differentiated w.r.t each entry of the nodal displacement vector and then, in this case, multiplied with the stress vector from Eq. (E.28).

$$\begin{gathered} \left(\frac{\partial\,[\mathbf{B_{NL}}]^{(i)}_{(e,k)}}{\partial\{\mathbf{a}\}^{(i)}_{(e)}}\right)^T \{\mathbf{S}\}^{(i)}_{(e,k)} = \\ \left[\left(\frac{\partial[\mathbf{B_{NL}}]^{(i)}_{(e,k)}}{\partial a_1}\right)^T \{\mathbf{S}\}^{(i)}_{(e,k)} \quad \frac{\partial[\mathbf{B_{NL}}]^{(i)}_{(e,k)}}{\partial a_2}\{\mathbf{S}\}^{(i)}_{(e,k)} \quad \dots \quad \frac{\partial[\mathbf{B_{NL}}]^{(i)}_{(e,k)}}{\partial a_8}\{\mathbf{S}\}^{(i)}_{(e,k)}\right] \end{gathered} \tag{E.33}$$

Each derivative in Eq. (E.33) is given in Eqs. (E.34) to (E.41)

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_1} = \begin{bmatrix} N_{1,x}N_{1,x} & 0 & \dots & N_{4,x}N_{1,x} & 0 \\ N_{1,y}N_{1,y} & 0 & \dots & N_{4,y}N_{1,y} & 0 \\ N_{1,x}N_{1,y} + N_{1,y}N_{1,x} & 0 & \dots & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} & 0 \end{bmatrix} \tag{E.34}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_2} = \begin{bmatrix} 0 & N_{1,x}N_{1,x} & \dots & 0 & N_{4,x}N_{1,x} \\ 0 & N_{1,y}N_{1,y} & \dots & 0 & N_{4,y}N_{1,y} \\ 0 & N_{1,x}N_{1,y} + N_{1,y}N_{1,x} & \dots & 0 & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} \end{bmatrix} \tag{E.35}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_3} = \begin{bmatrix} N_{1,x}N_{2,x} & 0 & \dots & N_{4,x}N_{2,x} & 0 \\ N_{1,y}N_{2,y} & 0 & \dots & N_{4,y}N_{2,y} & 0 \\ N_{1,x}N_{2,y} + N_{1,y}N_{2,x} & 0 & \dots & N_{4,x}N_{2,y} + N_{4,y}N_{2,x} & 0 \end{bmatrix} \tag{E.36}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_4} = \begin{bmatrix} 0 & N_{1,x}N_{2,x} & \dots & 0 & N_{4,x}N_{2,x} \\ 0 & N_{1,y}N_{2,y} & \dots & 0 & N_{4,y}N_{2,y} \\ 0 & N_{1,x}N_{2,y} + N_{1,y}N_{2,x} & \dots & 0 & N_{4,x}N_{2,y} + N_{4,y}N_{2,x} \end{bmatrix} \tag{E.37}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_5} = \begin{bmatrix} N_{1,x}N_{3,x} & 0 & \dots & N_{4,x}N_{3,x} & 0 \\ N_{1,y}N_{3,y} & 0 & \dots & N_{4,y}N_{3,y} & 0 \\ N_{1,x}N_{3,y} + N_{1,y}N_{3,x} & 0 & \dots & N_{4,x}N_{3,y} + N_{4,y}N_{3,x} & 0 \end{bmatrix} \tag{E.38}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_6} = \begin{bmatrix} 0 & N_{1,x}N_{3,x} & \dots & 0 & N_{4,x}N_{3,x} \\ 0 & N_{1,y}N_{3,y} & \dots & 0 & N_{4,y}N_{3,y} \\ 0 & N_{1,x}N_{3,y} + N_{1,y}N_{3,x} & \dots & 0 & N_{4,x}N_{3,y} + N_{4,y}N_{3,x} \end{bmatrix} \tag{E.39}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_7} = \begin{bmatrix} N_{1,x}N_{4,x} & 0 & \dots & N_{4,x}N_{4,x} & 0 \\ N_{1,y}N_{4,y} & 0 & \dots & N_{4,y}N_{4,y} & 0 \\ N_{1,x}N_{4,y} + N_{1,y}N_{4,x} & 0 & \dots & N_{4,x}N_{4,y} + N_{4,y}N_{4,x} & 0 \end{bmatrix} \tag{E.40}$$

$$\frac{\partial \left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_8} = \begin{bmatrix} 0 & N_{1,x}N_{4,x} & \dots & 0 & N_{4,x}N_{4,x} \\ 0 & N_{1,y}N_{4,y} & \dots & 0 & N_{4,y}N_{4,y} \\ 0 & N_{1,x}N_{4,y} + N_{1,y}N_{4,x} & \dots & 0 & N_{4,x}N_{4,y} + N_{4,y}N_{4,x} \end{bmatrix} \tag{E.41}$$

As an example, the derivatives of $\left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}$ w.r.t. the first and second nodal displacement will be multiplied by the stress vector in Eq. (E.42) and Eq. (E.46).

$$\left(\frac{\partial\left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_1}\right)^T \{\mathbf{S}\}^{(i)}_{(e,k)} \tag{E.42}$$

$$= \begin{bmatrix} N_{1,x}N_{1,x} & 0 & \dots & N_{4,x}N_{1,x} & 0 \\ N_{1,y}N_{1,y} & 0 & \dots & N_{4,y}N_{1,y} & 0 \\ N_{1,x}N_{1,y} + N_{1,y}N_{1,x} & 0 & \dots & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} & 0 \end{bmatrix}^T \begin{Bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{Bmatrix} \tag{E.43}$$

$$= \begin{bmatrix} N_{1,x}N_{1,x} & N_{1,y}N_{1,y} & N_{1,x}N_{1,y} + N_{1,y}N_{1,x} \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ N_{4,x}N_{1,x} & N_{4,y}N_{1,y} & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{Bmatrix} \tag{E.44}$$

$$= \begin{Bmatrix} (N_{1,x}N_{1,x})S_{11} + (N_{1,y}N_{1,y})S_{22} + (N_{1,x}N_{1,y} + N_{1,y}N_{1,x})S_{12} \\ 0 \\ (N_{2,x}N_{1,x})S_{11} + (N_{2,y}N_{1,y})S_{22} + (N_{2,x}N_{1,y} + N_{2,y}N_{1,x})S_{12} \\ 0 \\ (N_{3,x}N_{1,x})S_{11} + (N_{3,y}N_{1,y})S_{22} + (N_{3,x}N_{1,y} + N_{3,y}N_{1,x})S_{12} \\ 0 \\ (N_{4,x}N_{1,x})S_{11} + (N_{4,y}N_{1,y})S_{22} + (N_{4,x}N_{1,y} + N_{4,y}N_{1,x})S_{12} \\ 0 \end{Bmatrix} \tag{E.45}$$

$$\left(\frac{\partial\left[\mathbf{B_{NL}}\right]^{(i)}_{(e,k)}}{\partial a_2}\right)^T \{\mathbf{S}\}^{(i)}_{(e,k)} \tag{E.46}$$

$$= \begin{bmatrix} 0 & N_{1,x}N_{1,x} & \dots & 0 & N_{4,x}N_{1,x} \\ 0 & N_{1,y}N_{1,y} & \dots & 0 & N_{4,y}N_{1,y} \\ 0 & N_{1,x}N_{1,y} + N_{1,y}N_{1,x} & \dots & 0 & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} \end{bmatrix}^T \begin{Bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{Bmatrix} \tag{E.47}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ N_{1,x}N_{1,x} & N_{1,y}N_{1,y} & N_{1,x}N_{1,y} + N_{1,y}N_{1,x} \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ N_{4,x}N_{1,x} & N_{4,y}N_{1,y} & N_{4,x}N_{1,y} + N_{4,y}N_{1,x} \end{bmatrix} \begin{Bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{Bmatrix} \tag{E.48}$$

$$= \begin{Bmatrix} 0 \\ (N_{1,x}N_{1,x})S_{11} + (N_{1,y}N_{1,y})S_{22} + (N_{1,x}N_{1,y} + N_{1,y}N_{1,x})S_{12} \\ 0 \\ (N_{2,x}N_{1,x})S_{11} + (N_{2,y}N_{1,y})S_{22} + (N_{2,x}N_{1,y} + N_{2,y}N_{1,x})S_{12} \\ 0 \\ (N_{3,x}N_{1,x})S_{11} + (N_{3,y}N_{1,y})S_{22} + (N_{3,x}N_{1,y} + N_{3,y}N_{1,x})S_{12} \\ 0 \\ (N_{4,x}N_{1,x})S_{11} + (N_{4,y}N_{1,y})S_{22} + (N_{4,x}N_{1,y} + N_{4,y}N_{1,x})S_{12} \end{Bmatrix} \tag{E.49}$$

The contents in Eq. (E.42) and Eq. (E.46) looks like it could be described by a product between the derivative of the shape function matrices and the stresses contained in a matrix. Therefore, it is trialled to expand the stress stiffening part of the stiffness matrix to see if its matrix description is equal to the contents of Eq. (E.42) and Eq. (E.46). The stress stiffening part of the tangent stiffness matrix is given in Eq. (E.50).

$$\int_{V_0} ([\mathbf{G}]_{(e,k)})^T \left[\hat{\mathbf{S}}\right]^{(i)}_{(e,k)} [\mathbf{G}]_{(e,k)}\, dV \tag{E.50}$$

The part within the integral will be expanded in Eq. (E.51).

$$([\mathbf{G}]_{(e,k)})^T [\mathbf{S}]^{(i)}_{(e,k)} [\mathbf{G}]_{(e,k)} = \tag{E.51}$$

$$\begin{bmatrix} N_{1,x} & N_{1,y} & 0 & 0 \\ 0 & 0 & N_{1,x} & N_{1,y} \\ N_{2,x} & N_{2,y} & 0 & 0 \\ 0 & 0 & N_{2,x} & N_{2,y} \\ N_{3,x} & N_{3,y} & 0 & 0 \\ 0 & 0 & N_{3,x} & N_{3,y} \\ N_{4,x} & N_{4,y} & 0 & 0 \\ 0 & 0 & N_{4,x} & N_{4,y} \end{bmatrix} \begin{bmatrix} S_{11} & S_{12} & 0 & 0 \\ S_{12} & S_{22} & 0 & 0 \\ 0 & 0 & S_{11} & S_{12} \\ 0 & 0 & S_{12} & S_{22} \end{bmatrix} \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \tag{E.52}$$

Next, only the two first columns of the product between $\left[\hat{\mathbf{S}}\right]^{(i)}_{(e,k)}$ and $[\mathbf{G}]_{(e,k)}$ will be written for brevity.

$$\begin{bmatrix} N_{1,x} & N_{1,y} & 0 & 0 \\ 0 & 0 & N_{1,x} & N_{1,y} \\ N_{2,x} & N_{2,y} & 0 & 0 \\ 0 & 0 & N_{2,x} & N_{2,y} \\ N_{3,x} & N_{3,y} & 0 & 0 \\ 0 & 0 & N_{3,x} & N_{3,y} \\ N_{4,x} & N_{4,y} & 0 & 0 \\ 0 & 0 & N_{4,x} & N_{4,y} \end{bmatrix} \begin{bmatrix} S_{11} & S_{12} & 0 & 0 \\ S_{12} & S_{22} & 0 & 0 \\ 0 & 0 & S_{11} & S_{12} \\ 0 & 0 & S_{12} & S_{22} \end{bmatrix} \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} & 0 \\ N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} & 0 \\ 0 & N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 & N_{4,x} \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} & 0 & N_{4,y} \end{bmatrix} \tag{E.53}$$

$$= \begin{bmatrix} N_{1,x} & N_{1,y} & 0 & 0 \\ 0 & 0 & N_{1,x} & N_{1,y} \\ N_{2,x} & N_{2,y} & 0 & 0 \\ 0 & 0 & N_{2,x} & N_{2,y} \\ N_{3,x} & N_{3,y} & 0 & 0 \\ 0 & 0 & N_{3,x} & N_{3,y} \\ N_{4,x} & N_{4,y} & 0 & 0 \\ 0 & 0 & N_{4,x} & N_{4,y} \end{bmatrix} \begin{bmatrix} N_{1,x}S_{11} + N_{1,y}S_{12} & 0 & \dots \\ N_{1,x}S_{12} + N_{1,y}S_{22} & 0 & \dots \\ 0 & N_{1,x}S_{11} + N_{1,y}S_{12} & \dots \\ 0 & N_{1,x}S_{12} + N_{1,y}S_{22} & \dots \end{bmatrix} \tag{E.54}$$

$$= \begin{bmatrix} N_{1,x}(N_{1,x}S_{11} + N_{1,y}S_{12}) + N_{1,y}(N_{1,x}S_{12} + N_{1,y}S_{22}) & 0 & \dots \\ 0 & N_{1,x}(N_{1,x}S_{11} + N_{1,y}S_{12}) + N_{1,y}(N_{1,x}S_{12} + N_{1,y}S_{22}) & \dots \\ N_{2,x}(N_{1,x}S_{11} + N_{1,y}S_{12}) + N_{2,y}(N_{1,x}S_{12} + N_{1,y}S_{22}) & 0 & \dots \\ 0 & N_{2,x}(N_{1,x}S_{11} + N_{1,y}S_{12}) + N_{2,y}(N_{1,x}S_{12} + N_{1,y}S_{22}) & \dots \end{bmatrix} \tag{E.55}$$

If the contents of Eq. (E.42) and Eq. (E.46) are compared with the contents in Eq. (E.55) it can be concluded that the contents are equal, at least for the first two columns. However the same pattern should repeat in both Eq. (E.42), Eq. (E.46) and Eq. (E.55). Therefore, it can be concluded that the derivative in Eq. (E.33) can be rewritten using Eq. (E.51), as seen in Eq. (E.56).

$$\frac{\partial\{\mathbf{r}\}_{(e)}^{(i)}}{\partial\{\mathbf{a}\}_{(e)}^{(i)}} = \sum_{k=1}^{n_{ipts}} v_{(e)} \frac{\partial}{\partial\{\mathbf{a}\}_{(e)}^{(i)}} \left(\left([\mathbf{H}] + [\mathbf{A}(\boldsymbol{\theta})]_{(e,k)}^{(i)} \right) [\mathbf{G}]_{(e,k)} \right)^T \{\mathbf{S}\}_{(e,k)}^{(i)} w_{(e,k)} \tag{E.56}$$

$$= \sum_{k=1}^{n_{ipts}} v_{(e)} ([\mathbf{G}]_{(e,k)})^T \left[\hat{\mathbf{S}}\right]_{(e,k)}^{(i)} [\mathbf{G}]_{(e,k)} \, w_{(e,k)} \tag{E.57}$$

The derivative of the elemental force residual differentiated w.r.t to the elemental nodal displacements can thus be expressed as Eq. (E.56). The derivation has been performed for 2D 4-noded elements however if the $[\mathbf{H}]$, $[\mathbf{A}(\boldsymbol{\theta})]_{(e,k)}^{(i)}$, $[\mathbf{G}]_{(e,k)}$ and $\left[\hat{\mathbf{S}}\right]_{(e,k)}^{(i)}$ matrices are extended to 3D in a consistent manner, the same results should recur.