Department of Construction Sciences
Solid Mechanics

ISRN LUTFD2/TFHF-16/5215-SE(1-56)

A study of Heaviside projection methods in finite strain plasticity topology optimization

Master's Dissertation by

Jakob Engström

Supervisors:
Mathias Wallin, Professor. Division of Solid Mechanics

Examiner:
Håkan Hallberg, Docent. Division of Solid Mechanics

Printed by Media-Tryck AB, Lund, Sweden
For information, address:
Division of Solid Mechanics, Lund University, Box 118, SE-221 00 Lund, Sweden
Webpage: www.solid.lth.se

Abstract

Topology optimization using the density based method with penalization and filters result in non-discrete densities which is an issue since they can't be manufactured. These intermediate densities are forced to become discrete with Heaviside projections. The Heaviside function used in this work is a continuous differentiable function that starts with a linear behaviour and approaches a Heaviside step function with the optimization iterations. A finite strain plasticity problem is solved and optimized, with the objective being to maximize the plastic work in the structure. The optimization problem has been solved using the method of moving asymptotes. Since the structure has an elasto- plastic response the stresses depend on the load history, the optimization therefore becomes path dependant as well. The sensitivities has been calculated using the adjoint method.

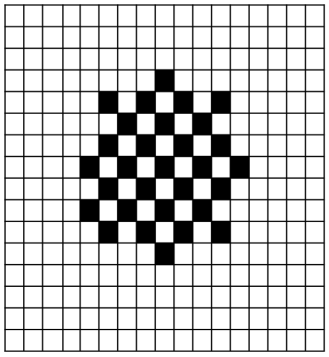
Contents

Introduction

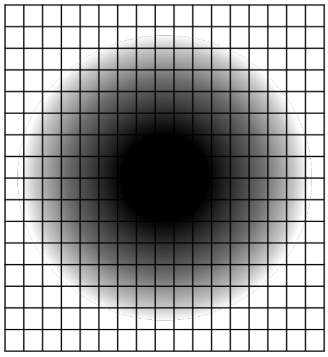
The goal of topology optimization is often to find the most optimal geometry or distribution of material within specified boundaries. It can be applied in an early stage in the design process to arrive close to an optimal design without iterations of design concepts. The most common and most researched method in topology optimization is the density based method used in this thesis [8]. It uses the density as the design variable and it varies between one and zero representing material and void in the structure. Most research done on the topic handles small elastic strains, there is however very little research that deals with large and plastic deformations in topology optimization. Another method that is promising and growing is the level-set method [12], [14].

The density based method combined with penalization struggles with a few problems such as non-discrete solutions i.e. gray regions with intermediate densities, mesh dependence and checkerboard patterns [2], [9]. The problems above can be overcome by different techniques such as filtering. The problem to be solved in this thesis is a topology optimization problem with large plastic deformations. As earlier mentioned the density based approach has been used together with SIMP (Solid isotropic material with penalization) to punish intermediate densities, see figure 1a. A Helmholtz density filter is implemented to deal with the checkerboard issue and gain control over the length scale, declaring an average minimum thickness of the material parts in the structure, see figure 1b. The benefit of using a Helmholtz PDE as filter is that it utilize the already existing FEM-frame and is therefore very efficient, other filters require large arrays of information on surrounding elements [5]. The intermediate densities caused by the density filter has been threshold using a Heaviside function, see figure 1c, forcing the densities to become one or zero [15], [9], [2]. An import part of topology optimization is the sensitivity analysis, which is the derivatives of the functionals. Two analytical methods exist to compute the sensitivities, the direct and adjoint method. In comparison with numerical methods such as the finite difference method which is easy to implement but is an approximation and suffers from round-off errors and truncation errors the analytical methods are more efficient and exact. The functions of the topology problem has been solved using MMA (Method of moving asymptotes) [11]. An isotropic material model has been used and the yield surface for evaluating the plastic response is of Von Mises type. Since the deformation is path dependant the sensitivity analysis becomes path dependant as well. The adjoint method is used in this

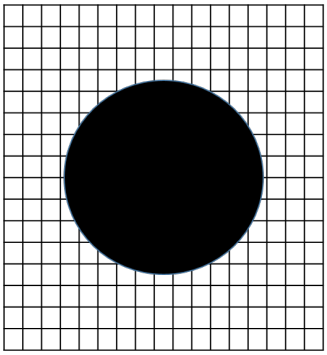
work, thus, in order to compute the sensitivities the primary structural problem must be solved first then the sensitivities are calculated after [6]. The objective of the optimization is to maximize the absorbed plastic work in the structure. The aim of this thesis is to implement a Heaviside filter in the topology optimization problem described above with large plastic strains, the program is written in fortran at the solid mechanics department at LTH. The work can be viewed as a continuation of the paper by Wallin et al. [13] where the structural problem and the topology optimization with density filters has been covered.



(a) Density based method with penalization ρ.



(b) Filtered densities $\tilde{\rho}$



(c) Heaviside projected densities. $\hat{\rho} = H(\tilde{\rho})$

Figure 1: Illustration of the checkerboard pattern, the non-discrete filtered densities and the discrete Heaviside projected densities.

Chapter 1

Continuum Mechanics

Continuum mechanics deals with the analysis of the kinematics and mechanical behaviour of materials which are modeled as a continuous mass. The motion of the continuous body from the undeformed configuration ($\mathcal{B}_0$), also called reference configuration to the deformed configuration ($\mathcal{B}$) called current configuration is shown in figure 1.1. Each point in the reference configuration of the body has a position vector $\boldsymbol{r_0}$ and in the current configuration $\boldsymbol{r}$. The vector mapping the motion from a point in the reference configuration to a point in the current configuration is $\boldsymbol{\varphi}_i$. A set of orthogonal unit base vectors $\boldsymbol{e_x}, \boldsymbol{e_y}, \boldsymbol{e_z}$ are introduced forming a Cartesian coordinate system [1].

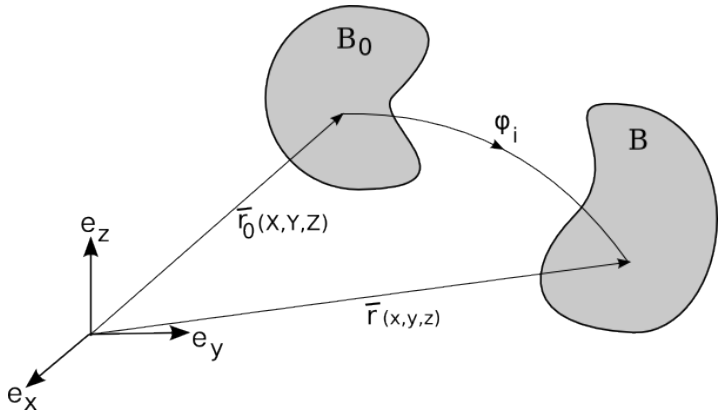


Figure 1.1: The deformed and undeformed configuration

The components of $\boldsymbol{r}_0$ are called material coordinates and the components of $\boldsymbol{r}$ are called spatial coordinates. The vectors $\boldsymbol{r}_0, \boldsymbol{r}$ can be expressed as

$$\boldsymbol{r}_0 = \boldsymbol{e}_x X + \boldsymbol{e}_y Y + \boldsymbol{e}_z Z$$

$$\boldsymbol{r} = \boldsymbol{e}_x x + \boldsymbol{e}_y y + \boldsymbol{e}_z z$$

The deformation of the body is expressed through a relation between the reference and the current configuration

$$\boldsymbol{r} = \boldsymbol{r}(\boldsymbol{r}_0)$$

and locally the deformation can be approximated by a linear mapping

$$\begin{aligned} dx &= \frac{\partial x}{\partial X}dX + \frac{\partial x}{\partial Y}dY + \frac{\partial x}{\partial Z}dZ \\ dy &= \frac{\partial y}{\partial X}dX + \frac{\partial y}{\partial Y}dY + \frac{\partial y}{\partial Z}dZ \\ dz &= \frac{\partial z}{\partial X}dX + \frac{\partial z}{\partial Y}dY + \frac{\partial z}{\partial Z}dZ \end{aligned} \tag{1.1}$$

which can also be written as

$$d\boldsymbol{r} = \boldsymbol{F}d\boldsymbol{r}_0 \tag{1.2}$$

where $\boldsymbol{F}$ is the deformation gradient, or in index notation where $x_i \in \mathcal{B}$ and $X_i \in \mathcal{B}_0$

$$F_{ij} = \frac{\partial x_i}{\partial X_j}. \tag{1.3}$$

The displacement between the two configuration is expressed by $\boldsymbol{u}$

$$\boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{u}(\boldsymbol{r}_0) \tag{1.4}$$

thus the differential becomes

$$d\boldsymbol{r} = (\boldsymbol{I} + \boldsymbol{D})d\boldsymbol{r}_0 \tag{1.5}$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{D}$ is the displacement gradient and defined by

$$D_{ij} = \frac{\partial u_i}{\partial X_j}. \tag{1.6}$$

As can be seen above the deformation gradient can be expressed by the displacement gradient

$$F_{ij} = \delta_{ij} + \frac{\partial u_i}{\partial X_j} \tag{1.7}$$

where δ_{ij} is the Kronecker delta. Green's strain is defined by

$$\varepsilon_G = \frac{l^2 - l_0^2}{2l_0^2} = \frac{d\boldsymbol{r}^T d\boldsymbol{r} - d\boldsymbol{r}_0^T d\boldsymbol{r}_0}{2d\boldsymbol{r}_0^T d\boldsymbol{r}_0} = \frac{d\boldsymbol{r}_0^T}{d\boldsymbol{s}_0^2}\frac{1}{2}(\boldsymbol{F}^{\boldsymbol{T}}\boldsymbol{F} - \boldsymbol{I})\frac{d\boldsymbol{r}_0}{d\boldsymbol{s}_0^2} \tag{1.8}$$

so the Green strain tensor is defined as

$$\boldsymbol{E}_G = \frac{1}{2}(\boldsymbol{F}^{\boldsymbol{T}}\boldsymbol{F} - \boldsymbol{I}) = \frac{1}{2}(\boldsymbol{D} + \boldsymbol{D}^{\boldsymbol{T}}) + \frac{1}{2}\boldsymbol{D}^{\boldsymbol{T}}\boldsymbol{D} \tag{1.9}$$

1.1 Weak form of the balance equations

The weak form is the base for the FEM formulation, it will be derived from the equation of balance of linear momentum [3]

$$\int_V \rho \dot{\boldsymbol{v}} dV = \int_S \boldsymbol{t} dS + \int_V \boldsymbol{b} dV. \tag{1.10}$$

Where $\rho = \rho(\boldsymbol{x}, t)$ is mass density, $\dot{\boldsymbol{v}} = \dot{\boldsymbol{v}}(\boldsymbol{X}, t)$ is the acceleration field, $\boldsymbol{t} = \boldsymbol{t}(\boldsymbol{x}, t)$ the Cauchy traction vector and $\boldsymbol{b} = \boldsymbol{b}(\boldsymbol{x}, t)$ the body force, with a given motion $\boldsymbol{x} = \mathcal{X}(\boldsymbol{X}, t)$ in spatial coordinates. Since there exists a spatial tensor field $\boldsymbol{\sigma}$ such that $\boldsymbol{t}(\boldsymbol{x}, t, \boldsymbol{n}) = \boldsymbol{\sigma}(\boldsymbol{x}, t)\boldsymbol{n}$, where $\boldsymbol{n}$ is the outward normal. The surface integral in (1.10) can be converted into a volume integral by using the divergence theorem

$$\int_S \boldsymbol{t}(\boldsymbol{x}, t, \boldsymbol{n}) dS = \int_S \boldsymbol{\sigma}(\boldsymbol{x}, t)\boldsymbol{n} dS = \int_V div\boldsymbol{\sigma}(\boldsymbol{x}, t) dV. \tag{1.11}$$

With the result from (1.11) into (1.10) Cauchy's first equation of motion on global form is obtained

$$\int_V (div\boldsymbol{\sigma} + \boldsymbol{b} - \rho \dot{\boldsymbol{v}}) dV = \boldsymbol{0}, \tag{1.12}$$

the relation hold for any volume V therefore the local form is

$$div\boldsymbol{\sigma} + \boldsymbol{b} = \rho \dot{\boldsymbol{v}}. \tag{1.13}$$

using that the velocity field $\boldsymbol{v}$ can be expressed by time rate of change of the displacement field $\boldsymbol{u}$ and equation (1.13) is written as

$$div\boldsymbol{\sigma} + \boldsymbol{b} = \rho \ddot{\boldsymbol{u}}, \tag{1.14}$$

equation (1.14) is multiplied with an arbitrary weight function $\boldsymbol{c} = \boldsymbol{c}(\boldsymbol{x})$ and integrated over the body

$$f(\boldsymbol{u}, \boldsymbol{c}) = \int_V (div\boldsymbol{\sigma} - \boldsymbol{b} + \rho \ddot{\boldsymbol{u}}) \cdot \boldsymbol{c} \, dV = 0 \tag{1.15}$$

equation (1.15) is the weak form of the equation of motion in spatial coordinates. Using the chain rule to the term $div\boldsymbol{\sigma} \cdot \boldsymbol{c}$ can be expressed as

$$div\boldsymbol{\sigma} \cdot \boldsymbol{c} = div(\boldsymbol{\sigma}\boldsymbol{c}) - \boldsymbol{\sigma} : grad\boldsymbol{c} \tag{1.16}$$

which together with the divergence theorem results in that equation (1.15) can be rewritten to

$$f(\boldsymbol{u}, \boldsymbol{c}) = \int_V [\boldsymbol{\sigma} : grad\boldsymbol{c} - (\boldsymbol{b} - \rho \ddot{\boldsymbol{u}}) \cdot \boldsymbol{c}] dV - \int_S \boldsymbol{t} \cdot \boldsymbol{c} \, dS = 0 \tag{1.17}$$

where it was used that $\boldsymbol{t} = \boldsymbol{\sigma}\boldsymbol{n}$ on the boundary of the surface. The arbitrary function $\boldsymbol{c}$ can be viewed as the virtual displacement field $\delta\boldsymbol{u}$ in the current configuration, which leads to the weak form of the equations of motion. Due to the symmetry of $\boldsymbol{\sigma}$ equation (1.17) can be expressed as

$$f(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_V [\boldsymbol{\sigma} : \frac{1}{2}(grad^T \delta\boldsymbol{u} + grad\delta\boldsymbol{u}) - (\boldsymbol{b} - \rho\ddot{\boldsymbol{u}}) \cdot \delta\boldsymbol{u}]dV - \int_S \boldsymbol{t} \cdot \delta\boldsymbol{u} dS = 0 \tag{1.18}$$

where the following functions

$$\delta W_{int}(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_V \boldsymbol{\sigma} : \frac{1}{2}(grad^T \delta\boldsymbol{u} + grad\delta\boldsymbol{u})dV \tag{1.19}$$

$$\delta W_{ext}(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_V \boldsymbol{b}\cdot\delta\boldsymbol{u} dV + \int_S \boldsymbol{t}\cdot\delta\boldsymbol{u} dS \tag{1.20}$$

are called the internal and external virtual work. From equation (1.18) one can see that if the acceleration of the displacement field is zero $\ddot{\boldsymbol{u}} = \boldsymbol{0}$ the internal virtual work is equal to the external virtual work $\delta W_{int}(\boldsymbol{u}, \delta\boldsymbol{u}) = \delta W_{ext}(\boldsymbol{u}, \delta\boldsymbol{u})$. In this work it is preferred to have the formulation of (1.18) in the reference configuration when solving it. The formulation in the reference configuration

$$f(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_{V_0} [\boldsymbol{P} : grad\delta\boldsymbol{u} - (\boldsymbol{B} - \rho_0\ddot{\boldsymbol{u}}) \cdot \delta\boldsymbol{u}]dV_0 - \int_{S_0} \boldsymbol{T} \cdot \delta\boldsymbol{u} dS_0 = 0 \tag{1.21}$$

where $\boldsymbol{P}$ is the first Piola-Kirchhoff stress tensor, $\boldsymbol{B}$ the body force and $\boldsymbol{T}$ the traction force in the reference configuration. The equation can also be expressed in terms of the second Piola-Kirchhoff stress tensor $\boldsymbol{S}$ instead of the first

$$f(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_{V_0} [\boldsymbol{S} : \delta\boldsymbol{E} - (\boldsymbol{B} - \rho_0\ddot{\boldsymbol{u}}) \cdot \delta\boldsymbol{u}]dV_0 - \int_{S_0} \boldsymbol{T} \cdot \delta\boldsymbol{u} dS_0 = 0 \tag{1.22}$$

where $\boldsymbol{E}$ is the Green-Lagrange strain tensor and

$$\delta\boldsymbol{E} = \frac{1}{2}[(\boldsymbol{F}^T grad\delta\boldsymbol{u})^T + \boldsymbol{F}^T grad\delta\boldsymbol{u}] \tag{1.23}$$

1.2 Plasticity

In the elastic structural problems the stresses and strains has a one-to-one relation, i.e. for a certain strain there exists only one stress. If the stresses reaches above the yield stress however the problem has to deal with plastic deformation and the one-to-one relation no longer holds since a certain strain can have two different stresses depending on the load history please see figure (1.2). To deter-

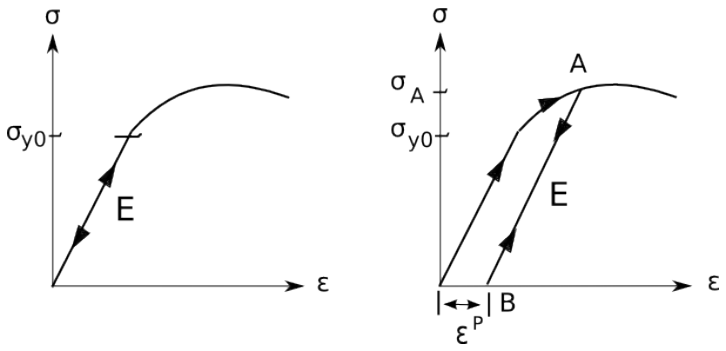


Figure 1.2: The figure to the left is the elastic region and the figure to the right the plastic

mine if a material response is plastic or elastic a yield function f is introduced. For f below zero no yielding occurs and when it is equal to zero the material is yielding, i.e.

$$f(\sigma_{ij}) = 0 \tag{1.24}$$

When the function $f(\sigma_{ij})$ is equal to zero it can be illustrated by a surface in the principal stress space called a yield surface, see figure 1.3.

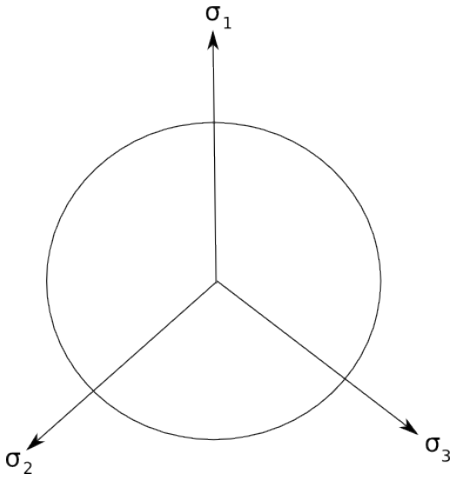


Figure 1.3: Von mises yield criterion in the deviatoric plane

If the response is elasto-plastic the yield function changes in order for the function to remain zero, this change of the yield surface is called the hardening rule. Since the yield surface not only depends on the stresses it becomes

$$\begin{aligned} &f(\sigma_{ij}, K_\beta) = 0 \\ &\beta = 1, 2, ... \end{aligned} \tag{1.25}$$

where K_β are the hardening parameters. When the hardening parameters are equal to zero equation (1.25) is equal to the initial yield function (1.24). Through the hardening parameters the size, shape and position of the surface (1.25) changes with plastic loading. Which implies that the choice of hardening parameters determines the choice of hardening rule and therefore also the change of the yield surface. To model this change some internal variables exist that characterize the state of the material i.e. the load history. The internal variables are expressed as

$$\begin{aligned} &\alpha_\kappa = \textit{internal variables} && (\kappa = 1, 2, ..) \\ &\alpha_\kappa = 0 && \textit{initially}, \end{aligned} \tag{1.26}$$

since the internal variables is assumed to mimic the history of the plastic loading they have to be zero initially and as the internal variables characterize the material the hardening parameters depend on the internal variables

$$K_\beta = K_\beta(\alpha_\kappa) \tag{1.27}$$

it then follows that

$$\dot{K}_\beta = \frac{\partial K_\beta}{\partial \alpha_\kappa} \dot{\alpha}_\kappa, \tag{1.28}$$

since $f = 0$ during plastic loading, the following expression also holds

$$\dot{f} = 0 \tag{1.29}$$

which is called the consistency condition and with the chain rule on equation (1.24) it becomes

$$\frac{\partial f}{\partial \sigma_{ij}} \dot{\sigma}_{ij} + \frac{\partial f}{\partial K_\beta} \dot{K}_\beta = 0. \tag{1.30}$$

Since the hardening parameters depend on the internal variables (1.27) we get the expression

$$\frac{\partial f}{\partial \sigma_{ij}} \dot{\sigma}_{ij} + \frac{\partial f}{\partial K_\beta} \frac{\partial K_\beta}{\partial \alpha_\kappa} \dot{\alpha}_\kappa = 0. \tag{1.31}$$

Some laws are needed that establish how α_κ evolute with plastic deformation, evolution laws. These laws must be based on experimental results and have a general format

$$\dot{\alpha}_\kappa = \gamma k_\kappa(\sigma_{ij}, K_\beta) \tag{1.32}$$

where k_κ is the evolution functions and γ is the plastic multiplier which is given by the Kuhn-Tucker conditions and the consistency condition

$$\gamma \geq 0, \quad f(\sigma_{ij}, K_\beta) \leq 0, \quad \gamma f(\sigma_{ij}, K_\beta) = 0, \quad \gamma \dot{f}(\sigma_{ij}, K_\beta) = 0 \tag{1.33}$$

1.3 Plasticity theory for finite strains

When the strains are small the sum of the plastic strains ε_{ij}^{p} and the elastic strains ε_{ij}^{e} makes up the total strains

$$\varepsilon_{ij} = \varepsilon_{ij}^{e} + \varepsilon_{ij}^{p} \tag{1.34}$$

this however is not valid when the strains are large, instead the multiplicative split of the deformations gradient is employed [10], i.e.

$$\boldsymbol{F}(\boldsymbol{X}, t) = \boldsymbol{F}^{e}(\boldsymbol{X}, t)\boldsymbol{F}^{p}(\boldsymbol{X}, t) \tag{1.35}$$

where $\boldsymbol{F}^{e}(\boldsymbol{X}, t)$ is the elastic and $\boldsymbol{F}^{p}(\boldsymbol{X}, t)$ the plastic part of the deformation gradient. The right Cauchy-Green tensor is defined as

$$\boldsymbol{C} = \boldsymbol{F}^{P^T}\boldsymbol{F}^{e^T}\boldsymbol{F}^{e}\boldsymbol{F}^{P}, \tag{1.36}$$

and

$$\boldsymbol{C}^{P} = \boldsymbol{F}^{P^T}\boldsymbol{F}^{P} \tag{1.37}$$

is the plastic right Cauchy-Green tensor. The Lagrangian strain tensor is also divided into a total and plastic part

$$\begin{aligned} \boldsymbol{E} &= \frac{1}{2}(\boldsymbol{C} - \boldsymbol{1}), \\ \boldsymbol{E}^{P} &= \frac{1}{2}(\boldsymbol{C}^{P} - \boldsymbol{1}), \end{aligned} \tag{1.38}$$

where both tensors are associated with the reference configuration, in the current configuration use will be made of the Eulerian tensors

$$\begin{aligned} \boldsymbol{b} &= \boldsymbol{F}\boldsymbol{F}^{T}, \\ \boldsymbol{b}^{e} &= \boldsymbol{F}^{e}\boldsymbol{F}^{e^T}, \end{aligned} \tag{1.39}$$

which are called the total and elastic left Cauchy-Green tensors. The deformation gradient can be split into a volume-preserving part and a deviatoric part as

$$\bar{\boldsymbol{F}} = J^{-1/3}\boldsymbol{F} \tag{1.40}$$

where $\bar{\boldsymbol{F}}$ is the volume preserving part with $det(\bar{\boldsymbol{F}}) = 1$, J is the Jacobian determinant and describes the local volumetric change $J = det(\boldsymbol{F})$. If the material is assumed to be isotropic, the stresses are independent of the orientation of the current and reference configuration. The plastic flow is isochoric which is a standard assumption in metal plasticity, we get

$$det\boldsymbol{F}^{P} = det\boldsymbol{C}^{P} = 1 \Rightarrow J = det\boldsymbol{F} = det\boldsymbol{F}^{e}. \tag{1.41}$$

Hence the result is no volume change for the plastic gradient. The following part can be seen as a summary of the key equations by Simo & Hughes, for

an in-depth discussion the reader is referred to [10]. The stress response is characterized by a stored-energy function

$$\begin{aligned} W &= U(J^e) + \bar{W}(\bar{\boldsymbol{b}}^e), \\ \bar{\boldsymbol{b}}^e &= J^{e-2/3}\boldsymbol{b}^e, \end{aligned} \tag{1.42}$$

where W is the strain energy, $U(J^e)$ and $\bar{W}(\bar{\boldsymbol{b}}^e)$ are the volumetric and deviatoric parts of W and are defined as

$$\begin{aligned} U(J)^e &= \frac{1}{2}\kappa\left[\frac{1}{2}(J^{e^2}-1) - lnJ^e\right], \\ \bar{W}(\bar{\boldsymbol{b}}^e) &= \frac{1}{2}\mu\bigg(tr[\bar{\boldsymbol{b}}^e] - 3\bigg), \end{aligned} \tag{1.43}$$

where μ and κ are the shear modulus and the bulk modulus. Since $tr[\bar{\boldsymbol{b}}^e] = tr[\bar{\boldsymbol{C}}^e]$ where $\bar{\boldsymbol{C}}^e$ is

$$\bar{\boldsymbol{C}}^e = J^{e-2/3}\boldsymbol{F}^{eT}\boldsymbol{F}^e \tag{1.44}$$

the deviatoric part of W can be written as $\hat{W}(\bar{\boldsymbol{C}}^e)$ instead of $\bar{W}(\bar{\boldsymbol{b}}^e)$ and the strain energy becomes

$$W = U(J^e) + \hat{W}(\bar{\boldsymbol{C}}^e). \tag{1.45}$$

the Kirchhoff stress tensor is given by

$$\begin{aligned} \boldsymbol{\tau} &= 2\boldsymbol{F}^e\frac{\partial W}{\partial \boldsymbol{C}^e}\boldsymbol{F}^{eT} = J^e U'(J^e)\mathbf{1} + \boldsymbol{s}, \\ \boldsymbol{s} &= 2dev\left[\bar{\boldsymbol{F}}^e\frac{\partial \hat{W}}{\partial \bar{\boldsymbol{C}}^e}\bar{\boldsymbol{F}}^{eT}\right]. \end{aligned} \tag{1.46}$$

The uncoupled, stored-energy function in (1.43) becomes an uncoupled volumetric-deviatoric stress-strain relationship in (1.46). The stresses can be rewritten as

$$\begin{aligned} \boldsymbol{\tau} &= J^e p\mathbf{1} + \boldsymbol{s}, \\ p &= U'(J^e) = \frac{\kappa}{2}(J^{e2}-1)/J^e, \\ \boldsymbol{s} &= dev[\boldsymbol{\tau}] = \mu dev[\bar{\boldsymbol{b}^e}] \end{aligned} \tag{1.47}$$

and as mentioned in previous section the yield function keeps track of the plastic response. The yield surface is here given by

$$f(\boldsymbol{\tau}, \alpha) = ||dev[\boldsymbol{\tau}]|| - \sqrt{\frac{2}{3}}[\sigma_{y0} + K\alpha] \tag{1.48}$$

where σ_{y0} is the initial yield stress, K is the hardening modulus and α is the internal variable. The associative flow rule is determined by the principle of maximum plastic dissipation, it takes the form as

$$\begin{aligned} \dot{\bar{\boldsymbol{C}}}^{p-1} &= -\frac{2}{3}\gamma[\boldsymbol{C} : \boldsymbol{C}^{p-1}]\boldsymbol{F}^{-1}\boldsymbol{n}\boldsymbol{F}^{-T}, \\ \boldsymbol{n} &= dev[\boldsymbol{\tau}]/||dev[\boldsymbol{\tau}]|| \end{aligned} \tag{1.49}$$

where γ is the plastic multiplier and $\boldsymbol{C} : \boldsymbol{C}^{p-1} = \sum_i \sum_j (C_{ij} C_{ij}{}^{p-1})$. The internal state variable evolves by the rate equation

$$\dot{\alpha} = \sqrt{\frac{2}{3}}\gamma, \tag{1.50}$$

the γ is subject to the previously discussed Kuhn-Tucker conditions and the consistency condition in the previous section.

1.4 Integration of the flow rule and hardening law

The problem will be solved with the finite element method and therefore it is needed to calculate the stresses at the next state. While everything is known in the current state, only the total displacements are known in the next state which are provided by the FE equations [7]. In order to calculate the stresses which depend on the left Cauchy-Green tensor the constitutive equations must be integrated since the left Cauchy-Green tensor evolves with the flow rule. The equations that need to be integrated are

$$\begin{aligned} \dot{\bar{\boldsymbol{C}}}^{p-1} &= -\frac{2}{3}\gamma[\boldsymbol{C} : \boldsymbol{C}^{p-1}]\boldsymbol{F}^{-1}\boldsymbol{n}\boldsymbol{F}^{-T}, \\ \dot{\alpha} &= \sqrt{\frac{2}{3}}\gamma, \end{aligned} \tag{1.51}$$

which has been defined in the previous section. Along with

$$f(\boldsymbol{\tau}, \alpha) = 0 \tag{1.52}$$

which must hold for a plastic response. The integration is performed using backward Euler difference scheme and results in the discrete evolution equations

$$\begin{aligned} {}^{n+1}\bar{\boldsymbol{C}}^{p-1} - {}^{n}\bar{\boldsymbol{C}}^{p-1} &= -\frac{2}{3}\Delta\gamma[{}^{n+1}\boldsymbol{C}^{p-1} : {}^{n+1}\boldsymbol{C}]^{n+1}\boldsymbol{F}^{-1\,n+1}\boldsymbol{n}^{n+1}\boldsymbol{F}^{-T}, \\ {}^{n+1}\alpha - {}^{n}\alpha &= \sqrt{\frac{2}{3}}\Delta\gamma. \end{aligned} \tag{1.53}$$

If the response is plastic, the equations in (1.53) must be solved in such a way that

$$f({}^{n+1}\boldsymbol{\tau}, {}^{n+1}\alpha) = 0 \tag{1.54}$$

is fulfilled. The equations are preferred in spatial coordinates which can be obtained with some tensor transformations. The expressions

$$\begin{aligned} {}^{n+1}\bar{\boldsymbol{F}}^{n+1}\bar{\boldsymbol{C}}^{p-1\,n+1}\bar{\boldsymbol{F}}^{T} &= {}^{n+1}\bar{\boldsymbol{b}}^{e}, \\ {}^{n+1}\boldsymbol{C} : {}^{n+1}\boldsymbol{C}^{p-1} &= tr[{}^{n+1}\boldsymbol{b}^{e}], \end{aligned} \tag{1.55}$$

are used and the derivation can be found in [10]. The first equation in (1.53) can now be rewritten as

$$ {}^{n+1}\bar{\boldsymbol{C}}^{p-1} - {}^{n}\bar{\boldsymbol{C}}^{p-1} = -\frac{2}{3}\Delta\gamma^{n+1}J^{-2/3}tr[{}^{n+1}\boldsymbol{b}^{e}]^{n+1}\bar{\boldsymbol{F}}^{-1}{}^{n+1}\boldsymbol{n}^{n+1}\bar{\boldsymbol{F}}^{-T}, \quad (1.56) $$

where ${}^{n+1}\boldsymbol{F}^{-1} = {}^{n+1}J^{-1/3}{}^{n+1}\bar{\boldsymbol{F}}^{-1}$ has been used. If ${}^{n+1}\boldsymbol{f}$ and ${}^{n+1}\bar{\boldsymbol{f}}$ is introduced as the relative deformation gradient and its volume preserving part, they are defined as

$$ \begin{aligned} {}^{n+1}\boldsymbol{f} &= {}^{n+1}\boldsymbol{F}^{n}\boldsymbol{F}^{-1}, \\ {}^{n+1}\bar{\boldsymbol{f}} &= {}^{n+1}\bar{\boldsymbol{F}}^{n}\bar{\boldsymbol{F}}^{-1}. \end{aligned} \quad (1.57) $$

If each term in the first equation of (1.56) is pre multiplied with ${}^{n+1}\bar{\boldsymbol{F}}$ and post multiplied by ${}^{n+1}\bar{\boldsymbol{F}}^{T}$ together with equation (1.57) we get

$$ {}^{n+1}\bar{\boldsymbol{b}}^{e} = {}^{n+1}\bar{\boldsymbol{f}}^{n}\bar{\boldsymbol{b}}^{e}{}^{n+1}\bar{\boldsymbol{f}}^{T} - \frac{2}{3}\Delta\gamma tr[{}^{n+1}\bar{\boldsymbol{b}}^{e}]^{n+1}\boldsymbol{n}. \quad (1.58) $$

1.5 Internal Residual

The equations that must be fulfilled in each iteration are

$$ f({}^{n+1}\boldsymbol{\tau}, {}^{n+1}\alpha) = ||dev[{}^{n+1}\boldsymbol{\tau}]|| - \sqrt{\frac{2}{3}}[\sigma_{y0} + K^{n+1}\alpha] = 0, \quad (1.59) $$

$$ {}^{n+1}\bar{\boldsymbol{b}}^{e} = {}^{n+1}\bar{\boldsymbol{f}}^{n}\bar{\boldsymbol{b}}^{e}{}^{n+1}\bar{\boldsymbol{f}}^{T} - \frac{2}{3}\Delta\gamma tr[{}^{n+1}\bar{\boldsymbol{b}}^{e}]^{n+1}\boldsymbol{n}, \quad (1.60) $$

where

$$ \Delta\alpha = {}^{n+1}\alpha - {}^{n}\alpha = \sqrt{\frac{2}{3}}\Delta\gamma, \quad (1.61) $$

$$ {}^{n+1}\boldsymbol{\tau} = {}^{n+1}J^{e}U'({}^{n+1}J^{e})\boldsymbol{1} + {}^{n+1}\boldsymbol{s}. \quad (1.62) $$

In equation (1.60) the $\Delta\gamma$ can be replaced by the expression of $\Delta\alpha$ and become

$$ {}^{n+1}\bar{\boldsymbol{b}}^{e} = {}^{n+1}\bar{\boldsymbol{f}}^{n}\bar{\boldsymbol{b}}^{e}{}^{n+1}\bar{\boldsymbol{f}}^{T} - \sqrt{\frac{2}{3}}\Delta\alpha tr[{}^{n+1}\bar{\boldsymbol{b}}^{e}]^{n+1}\boldsymbol{n}, \quad (1.63) $$

therefore an internal residual is formed that must always be satisfied and will look different for a plastic or elastic response. The residual is called C and depends on the internal state variables that can be altered in order for the residual to be fulfilled. For a plastic response the residual is

$$ {}^{n+1}\boldsymbol{C} = \begin{cases} {}^{n+1}\bar{\boldsymbol{b}}^{e} - {}^{n+1}\bar{\boldsymbol{f}}^{n}\bar{\boldsymbol{b}}^{e}{}^{n+1}\bar{\boldsymbol{f}}^{T} - \sqrt{\frac{2}{3}}\Delta\alpha tr[{}^{n+1}\bar{\boldsymbol{b}}^{e}]^{n+1}\boldsymbol{n} \\ ||dev[{}^{n+1}\boldsymbol{\tau}]|| - \sqrt{\frac{2}{3}}[\sigma_{y0} + K^{n+1}\alpha] \end{cases} \quad (1.64) $$

with the internal state variables are

$$ {}^{n+1}\boldsymbol{v} = \begin{Bmatrix} {}^{n+1}\bar{\boldsymbol{b}}^{e} \\ {}^{n+1}\alpha \end{Bmatrix} \tag{1.65} $$

For an elastic response the residual become

$$ {}^{n+1}\boldsymbol{C} = \{^{n+1}\bar{\boldsymbol{b}}^{e} - {}^{n+1}\bar{\boldsymbol{f}}^{n}\bar{\boldsymbol{b}}^{e}{}^{n+1}\bar{\boldsymbol{f}}^{T} \tag{1.66} $$

Since α is constant for an elastic response, the internal state variable is

$$ {}^{n+1}\boldsymbol{v} = \{^{n+1}\bar{\boldsymbol{b}}^{e} \tag{1.67} $$

Chapter 2

Finite Element Method

The basis for the finite element formulation is the weak form of the equations of motion, ie.

$$\int_{V_0} [\boldsymbol{S} : \delta\boldsymbol{E} - \boldsymbol{b} \cdot \delta\boldsymbol{u}] dV_0 - \int_{S_0} \boldsymbol{T} \cdot \delta\boldsymbol{u} dS_0 = 0 \tag{2.1}$$

where $\delta\boldsymbol{E}$ in equation 2.1 is

$$\delta\boldsymbol{E} = \frac{1}{2}[(\boldsymbol{F}^T grad\delta\boldsymbol{u})^T + \boldsymbol{F}^T grad\delta\boldsymbol{u}] \tag{2.2}$$

For simplicity the acceleration was ignored $\ddot{\boldsymbol{u}} = 0$. The displacement field $\boldsymbol{u}$ is approximated with the shape functions $\boldsymbol{N}$ and the nodal displacements as

$$\boldsymbol{u} = \boldsymbol{N}\boldsymbol{u} \tag{2.3}$$

the virtual displacement is approximated in the same manner as the displacement field

$$\delta\boldsymbol{u} = \boldsymbol{N}\delta\boldsymbol{u}. \tag{2.4}$$

Since the formulation is in the reference configuration the shape functions are dependant on the material coordinates as $\boldsymbol{N} = \boldsymbol{N}(\boldsymbol{X})$. Inserting the equations above into (2.1) results in

$$\delta\boldsymbol{u}\underbrace{\left(\int_{V_0} [\boldsymbol{B}^T\boldsymbol{S} - \boldsymbol{N}^T\boldsymbol{b}] dV_0 - \int_{S_0} \boldsymbol{N}^T\boldsymbol{T} dS_0\right)}_{\boldsymbol{R}} = 0 \tag{2.5}$$

As $\delta\boldsymbol{u}$ is an arbitrary function and the approximation of $\delta\boldsymbol{E} = \boldsymbol{B}\delta\boldsymbol{u}$ has been used. In equation (2.5) $\boldsymbol{R}$ is the outer residual and will be used in the coming chapters.

Chapter 3

Steady-state coupled non-linear system

Combining the mechanics and constitutive equations (2.5) and (1.64) shows that in each time step a set of non-linear coupled system defined as

$$\begin{aligned} {}^{n+1}\boldsymbol{R}({}^{n+1}\boldsymbol{u}, {}^{n}\boldsymbol{u}, {}^{n+1}\boldsymbol{v}, {}^{n}\boldsymbol{v}) &= \boldsymbol{0} \\ {}^{n+1}\boldsymbol{C}({}^{n+1}\boldsymbol{u}, {}^{n}\boldsymbol{u}, {}^{n+1}\boldsymbol{v}, {}^{n}\boldsymbol{v}) &= \boldsymbol{0} \end{aligned} \tag{3.1}$$

must be satisfied. In (3.1), n is the current known state where $\boldsymbol{u}$ and $\boldsymbol{v}$ are known and $n+1$ is the next state where they are unknown. A common procedure to solve the coupled problem is to uncouple it, by treating the response $\boldsymbol{v}$ as a function of the system response $\boldsymbol{u}$, the residuals are then written as

$$\begin{aligned} \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u}) &= \boldsymbol{0}, \\ \boldsymbol{C}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u}) &= \boldsymbol{0}. \end{aligned} \tag{3.2}$$

The system in (3.2) are solved by implementing two Newton-Raphson loops. In the outer loop the residual $\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u})$ is linearized by performing a Taylor expansion around $\boldsymbol{u}$

$$\begin{aligned} &\boldsymbol{R}(\boldsymbol{u}^{I+1}, \boldsymbol{v}(\boldsymbol{u}^{I+1})) \\ &\approx \boldsymbol{R}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I})) + \left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{u}}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I})) + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{v}}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I}))\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^{I})\right] d\boldsymbol{u} \end{aligned} \tag{3.3}$$

where I is the current iteration and $I+1$ is the next iteration, since the residual must be satisfied e.g. (3.2) the residual is assumed to be zero in the next iteration and equation (3.3) becomes

$$\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{u}}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I})) + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{v}}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I}))\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^{I})\right] d\boldsymbol{u} = -\boldsymbol{R}(\boldsymbol{u}^{I}, \boldsymbol{v}(\boldsymbol{u}^{I})). \tag{3.4}$$

To be able to solve the equation above both the system response $\boldsymbol{v}(\boldsymbol{u}^{I})$ and the derivative $\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^{I})$ are needed. Therefore the inner Newton-Raphson loop

is used to solve the residual $\boldsymbol{C(u, v(u))}$ for a fixed $\boldsymbol{u}^I$, first $\boldsymbol{C(u, v(u))}$ is linearized by using a Taylor expansion around the current iterative $\boldsymbol{v}^J(\boldsymbol{u}^I)$

$$\boldsymbol{C}(\boldsymbol{u}^I, \boldsymbol{v}^{J+1}(\boldsymbol{u}^I)) \approx \boldsymbol{C}(\boldsymbol{u}^I, \boldsymbol{v}^J(\boldsymbol{u}^I)) + \frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}^J(\boldsymbol{u}^I))d\boldsymbol{v} \tag{3.5}$$

and as before the residual is assumed to satisfy it's condition and be equal to zero in the next iteration, equation (3.5) becomes

$$\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}^J(\boldsymbol{u}^I))d\boldsymbol{v} = -\boldsymbol{C}(\boldsymbol{u}^I, \boldsymbol{v}^J(\boldsymbol{u}^I)). \tag{3.6}$$

The term $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}$ is called the dependent tangent operator. The next iterate $\boldsymbol{v}^{J+1}(\boldsymbol{u}^I)$ is calculated from

$$\boldsymbol{v}^{J+1}(\boldsymbol{u}^I) = \boldsymbol{v}^J(\boldsymbol{u}^I) + d\boldsymbol{v}. \tag{3.7}$$

The inner Newton-Raphson loop iterates until the solution is obtained, then the derivative $\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^I)$ is calculated by differentiating the residual $\boldsymbol{C}$ in equation (3.2) i.e.

$$\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{u}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I)) + \frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^I) = \boldsymbol{0} \tag{3.8}$$

and the derivative $\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^I)$ become

$$\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^I) = -\left(\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\right)^{-1}\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{u}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I)). \tag{3.9}$$

The dependent operator $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))$ has been used earlier in equation (3.6) when computing the incremental response and can be used again. The expression for the derivative $\frac{D\boldsymbol{v}}{D\boldsymbol{u}}(\boldsymbol{u}^I)$ in equation (3.9) is inserted in equation (3.4) and it becomes

$$\begin{aligned}&\left\{\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{u}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I)) - \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\left[\left(\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{v}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\right)^{-1}\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{u}}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\right]\right\}d\boldsymbol{u} \\ &= -\boldsymbol{R}(\boldsymbol{u}^I, \boldsymbol{v}(\boldsymbol{u}^I))\end{aligned} \tag{3.10}$$

where the term between the braces is called the independent tangent operator, the system response is then updated by

$$\boldsymbol{u}^{I+1} = \boldsymbol{u}^I + d\boldsymbol{u}. \tag{3.11}$$

The loops continue to iterate until both residuals are fulfilled.

Chapter 4

Topology optimization

4.1 General

The idea with optimization in structural mechanics is to produce the optimal load bearing structure possible within the constraints. The general structural optimization problem consists of the objective function f that returns a number that describes how good the design is, usually it's a minimization problem so the lower number that f returns the better the design is. The design variable $\boldsymbol{\phi}$ can be a function or vector that can change value in order to improve the design, it may represent geometry or density in a structure. The state variable $\boldsymbol{\psi}$ represents the response of the structure for a given design $\boldsymbol{\phi}$, it can be the displacement or force etc. The optimization problem is usually presented as

$$(\mathbb{SO}) \begin{cases} \text{minimize } f(\boldsymbol{\phi}, \boldsymbol{\psi}) \text{ with respect to } \boldsymbol{\phi} \text{ and } \boldsymbol{\psi} \\ \text{subject to } \begin{cases} \text{behavioral constraint on } \boldsymbol{\psi} \\ \text{design constraint on } \boldsymbol{\phi} \\ \text{equilibrium constraint.} \end{cases} \end{cases}$$

The formulation above is called simultaneous formulation, $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ are treated as independent variables and the state problem $\boldsymbol{\psi}$ is solved simultaneous as the design problem $\boldsymbol{\phi}$. A common situation however is that the state problem uniquely defines $\boldsymbol{\psi}$ for a given design $\boldsymbol{\phi}$, if the stiffness matrix $\boldsymbol{K}(\boldsymbol{\phi})$ always has an inverse for any given $\boldsymbol{\phi}$ the equilibrium constraint can be left out of the optimization problem. The formulation of $(\mathbb{SO})$ becomes $(\mathbb{SO}_{nf})$ and is called a nested formulation.

$$(\mathbb{SO}_{nf}) \begin{cases} \min\limits_{\phi \in \mathcal{X}} & f(\boldsymbol{\phi}, \boldsymbol{\psi}(\boldsymbol{\phi})) \\ \text{s.t.} & g(\boldsymbol{\phi}, \boldsymbol{\psi}(\boldsymbol{\phi})) \leq 0, \end{cases} \tag{4.1}$$

where $\mathcal{X}$ is the set where ϕ is defined and g is the constraint function.

4.2 Optimal solution

When solving optimization problems it matters whether the set $\mathcal{X}$ of the optimization problem is convex or not. For a set to be convex any linear combination between two points $\phi \in \mathcal{X}$ should also be in the set $\mathcal{X}$, see figure 4.1 below.

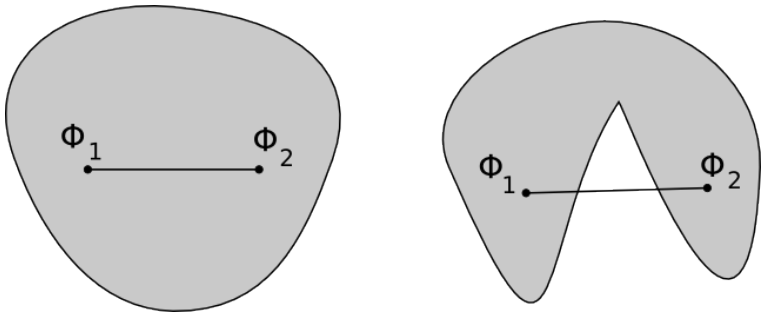


Figure 4.1: The convex set to the left and the non-convex set to the right

The condition for convexity can be mathematically expressed by

$$\lambda\boldsymbol{\phi}_1 + (1-\lambda)\boldsymbol{\phi}_2 \in \mathcal{X} \text{ and } \lambda \in (0,1). \tag{4.2}$$

When solving optimization problems finding the global minimum to the problem is key, if the problem is convex any local minimum is a global minimum. Most of the problems are however non-convex which will be discussed in the next section. For large scale problems one suitable method is Lagrangian duality, the Lagrangian is defined by (4.3) below.

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\lambda}) = g_0(\boldsymbol{\phi}) + \sum_{i=1}^{l} \lambda_i g_i(\boldsymbol{\phi}) \tag{4.3}$$

where λ_i, $i = 1, ..., \ l$ are the Lagrangian multipliers. It can be proven that ($\mathbb{SO}$) is equivalent to the min-max problem (4.4).

$$\mathbb{SO}_L \ \min_{\boldsymbol{\phi}\in\mathcal{X}} \max_{\lambda\geq\mathbf{0}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\lambda}) = \min_{\boldsymbol{\phi}\in\mathcal{X}} \max_{\lambda\geq\mathbf{0}} \left\{ g_0(\boldsymbol{\phi}) + \sum_{i=1}^{l} \lambda_i g_i(\boldsymbol{\phi}) \right\} \tag{4.4}$$

First the Lagrangian $\mathcal{L}$ is maximized with respect to $\boldsymbol{\lambda} \geq 0$ for a fixed $\boldsymbol{\phi}$ and then the maximized function is minimized with respect to $\boldsymbol{x} \in \mathcal{X}$. This is corresponding to the Lagrangian dual problem $\mathbb{D}$ shown in (4.5).

$$(\mathbb{D}) \ \begin{cases} \max\limits_{\boldsymbol{\lambda}} \boldsymbol{\varphi}(\boldsymbol{\lambda}) \\ s.t. \ \boldsymbol{\lambda} \geq \mathbf{0} \end{cases} \tag{4.5}$$

where $\boldsymbol{\varphi}$ is the dual objective function and is defined as

$$\boldsymbol{\varphi}(\boldsymbol{\lambda}) = \min_{\boldsymbol{\phi}\in\boldsymbol{\mathcal{X}}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\lambda}). \tag{4.6}$$

The difference between the solution of the original problem ($\mathbb{SO}$) and the dual problem ($\mathbb{D}$) is called the duality gap. If the problem is convex and Slater's CQ is satisfied then the duality gap becomes zero, which means the solution to ($\mathbb{D}$) and ($\mathbb{SO}$) is equal. So instead of solving the original problem the dual problem is solved instead.

If the objective function g_0 is strictly convex and $g_i i = 1, ..., l$ are convex and separable so they can be written as a sum of functions of a single variable

$$g_i(\boldsymbol{\phi}) = \sum_{j=1}^{n} g_{ij}(\boldsymbol{\phi}_j), \ i = 0, ..., l. \tag{4.7}$$

The Lagrangian $\mathcal{L}$ becomes

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\lambda}) &= g_0(\boldsymbol{\phi}) + \sum_{i=1}^{l} \lambda_i g_i(\boldsymbol{\phi}) \\
&= \sum_{j=1}^{n} g_{0j}(\phi_j) + \sum_{i=1}^{l} \lambda_i \left(\sum_{j=1}^{n} g_{ij}(\phi_j) \right) \\
&= \sum_{j=1}^{n} \left(g_{0j}(\phi_j) + \sum_{i=1}^{l} \lambda_i g_{ij}(\phi_j) \right)
\end{aligned} \tag{4.8}$$

and the dual objective function is now rewritten as

$$\boldsymbol{\varphi}(\boldsymbol{\lambda}) = \min_{\phi \in \mathcal{X}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\lambda}) = \min_{\phi \in \mathcal{X}} \sum_{j=1}^{n} \left(g_{0j}(\phi_j) + \sum_{i=1}^{l} \lambda_i g_{ij}(\phi_j) \right). \tag{4.9}$$

When solving the problem above all that is needed is to minimize the Lagrangian for all $\boldsymbol{\phi}$ which are n times for a single variable. This makes Lagrangian duality a good solving technique for convex, separable problems. With finite upper and lower bounds the minimization is performed as follows by

$$\begin{aligned}
&\text{if} && \frac{\partial \mathcal{L}(\phi_j^{min}, \boldsymbol{\lambda})}{\partial \phi_j} \geq 0, && \text{then } \phi_j^* = \phi_j^{min} \\
&\text{else if} && \frac{\partial \mathcal{L}(\phi_j^{max}, \boldsymbol{\lambda})}{\partial \phi_j} \geq 0, && \text{then } \phi_j^* = \phi_j^{max} \\
&\text{else} && \phi_j^* = \phi_j^*(\boldsymbol{\lambda}) && \text{from } \frac{\partial \mathcal{L}(\phi_j, \boldsymbol{\lambda})}{\partial \phi_j} = 0
\end{aligned} \tag{4.10}$$

4.3 Method of Moving Asymptotes

In the previous section the optimization solving methods handled convex problems, although many of the problems are non-convex. The solution is to make a convex approximation of the original problem. One of the most common approximation is MMA (Method of Moving Asymptotes) introduced by Svanberg [11]. There exist many other approximations such as CONLIN (Convex Linearization) but MMA offers more control on how conservative the approximation should be, and it's the approximation used in this paper. For iteration k, the MMA approximation ($g_i^{M,k}$) to the original function (g_i^k) is

$$g_i^{M,k}(\boldsymbol{\phi}) = r_i^k + \sum_{j=1}^{n}\left(\frac{p_{ij}^k}{U_j^k - \phi_j} + \frac{q_{ij}^k}{\phi_j - L_j^k}\right) \tag{4.11}$$

where L_j and U_j are the moving asymptotes that changes through iterations and satisfies

$$L_j^k < \phi_j^k < U_j^k, \tag{4.12}$$

and

$$p_{ij}^k = \begin{cases} (U_j^k - \phi_j^k)^2 \dfrac{\partial g_i(\boldsymbol{\phi}^k)}{\partial \phi_j} & \text{if } \dfrac{\partial g_i(\boldsymbol{\phi}^k)}{\partial \phi_j} > 0 \\ 0 & \text{otherwise,} \end{cases} \tag{4.13}$$

$$q_{ij}^k = \begin{cases} 0 & \text{if } \dfrac{\partial g_i(\boldsymbol{\phi}^k)}{\partial \phi_j} < 0 \\ -(\phi_j^k - L_j^k)^2 \dfrac{\partial g_i(\boldsymbol{\phi}^k)}{\partial \phi_j} & \text{otherwise,} \end{cases} \tag{4.14}$$

$$r_i^k = g_i(\boldsymbol{\phi}^k) - \sum_{j=1}^{n}\left(\frac{p_{ij}^k}{U_j^k - \phi_j^k} + \frac{q_{ij}^k}{\phi_j^k - L_j^k}\right) \tag{4.15}$$

So either p_{ij}^k or q_{ij}^k is equal to zero. When solving the duality problem in previous section the derivative of the objective function and all constraint functions with respect to the design variable is needed, these derivatives are called sensitivities and will be discussed more in the next section. The sensitivities become

$$\frac{\partial g_i^{M,k}(\boldsymbol{\phi})}{\partial \phi_j} = \frac{p_{ij}^k}{(U_j^k - \phi_j)^2} - \frac{q_{ij}^k}{(\phi_j - L_j^k)^2}, \tag{4.16}$$

The formulation changes from $\mathbb{SO}_{nf}$ to

$$(\mathbb{MMA}) \begin{cases} \min\limits_{\phi} g_0^{M,k}(\boldsymbol{\phi}) \\ s.t. \begin{cases} g_i^{M,k}(\boldsymbol{\phi}) \le 0, & i = 1, ..., l \\ \alpha_j^k \le \phi_j^k \le \beta_j^k, & j = 1, ..., l \end{cases} \end{cases} \tag{4.17}$$

where α_j^k and β_j^k are move limits that will be defined later. The problem above is convex and separable which makes it suitable to solve with Lagrangian duality. Often $\varepsilon(U_j^k - \phi_j^k)^2/(U_j^k - L_j^k)$ is added to p_{0j}^k and $\varepsilon(\phi_j^k - L_j^k)^2/(U_j^k - L_j^k)$ is added to q_{0j}^k where ε is a small positive number, to make the objective function $g_0^{M,k}$ strictly convex. When choosing the asymptotes it's decided how conservative the approximation is. The closer the asymptotes are to the current design the bigger the approximation becomes, and the further apart from the design, the less conservative it gets. There is no certain way to do this but the updating approach proposed by Svanberg follows. For the first to iterations k, the asymptotes are chosen as

$$L_j^k = \phi_j^k - s_{init}(\phi_j^{max} - \phi_j^{min}) \tag{4.18}$$

$$U_j^k = \phi_j^k + s_{init}(\phi_j^{max} - \phi_j^{min}) \tag{4.19}$$

where $0 < s_{init} < 1$, depending on the degree of conservatism, but a typical value is 0.5 and a lower number makes it more conservative and a higher less conservative. ϕ_j^{max} and ϕ_j^{min} are the upper and lower bounds of the design variable. After the first two iterations the signs of $(\phi_j^k - \phi_j^{k-1})$ and $(\phi_j^{k-1} - \phi_j^{k-2})$ are studied, if the two parenthesises has different signs the variable ϕ_j oscillates. When that happens the approximation should be made more conservative by moving the asymptotes closer to the design. That is made by

$$L_j^k = \phi_j^k - s_{slower}(\phi_j^{k-1} - L_j^{k-1})$$

$$U_j^k = \phi_j^k + s_{slower}(\phi_j^{k-1} - U_j^{k-1})$$

where $0 < s_{slower} < 1$. If however the signs of $(\phi_j^k - \phi_j^{k-1})$ and $(\phi_j^{k-1} - \phi_j^{k-2})$ are the same the asymptotes should be moved further from the current design in order to speed up the convergence

$$L_j^k = \phi_j^k - s_{faster}(\phi_j^{k-1} - L_j^{k-1})$$

$$U_j^k = \phi_j^k + s_{faster}(\phi_j^{k-1} - U_j^{k-1})$$

where $s_{faster} > 1$. The move limits α_j^k and β_j^k in the formulation ($\mathbb{MMA}$) are chosen as

$$\alpha_j^k = max(\phi_j^{min}, L_j^k + \mu(\phi_j^k - L_j^k)), \tag{4.20}$$

$$\beta_j^k = min(\phi_j^{max}, U_j^k - \mu(U_j^k - \phi_j^k)), \tag{4.21}$$

where $0 < \mu < 1$, and the design variables are to fulfill the following constraint each iteration

$$\alpha_j^k \leq \phi_j^k \leq \beta_j^k, \tag{4.22}$$

then the following will always be true

$$L_j^k < \alpha_j^k \leq \phi_j^k \leq \beta_j^k < U_j^k \tag{4.23}$$

Since $L_j^k \neq \phi_j^k \neq U_j^k$ the term $(\phi_j^k - L_j^k)$ and $(U_j^k - \phi_j^k)$ can never be zero it prevents division by zero.

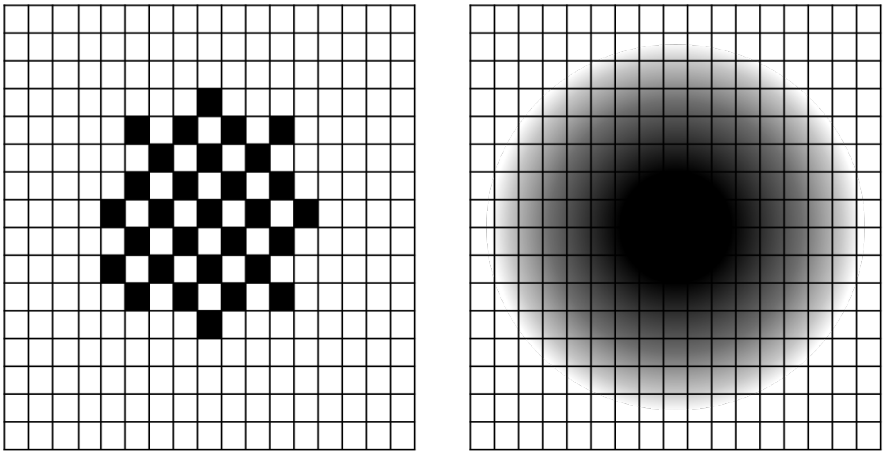
Chapter 5

Regularization and Thresholding

In order to obtain mesh independent, checkerboard-free and discrete solutions filters and Heaviside projections are implemented. When the intermediate densities are penalized with the SIMP method it creates checkerboard patterns, alternating elements with material and void see figure 5.1. In order to get rid of the checkerboard pattern the area around a certain element need to affect the density of the element. A length-scale is needed that create a radius around the element so the elements within that radius have an impact on the density of the element. The length scale is introduced through filters, the filters use the elements within the length scale to create a weighted average that decide the density of the element. The filter will punish elements that are surrounded by void and increase the density of elements surrounded by material.
Since the density of the element will be an average, intermediate densities will appear at the edges between the material parts and void in the structure, see figure 5.1. These intermediate densities are unwanted and are difficult to relate to in the physical world, i.e. they can not be manufactured. In order to obtain a discrete solution with only material or void a Heaviside projection can be applied to the filtered densities, see figure 5.2. The Heaviside function force the intermediate densities towards solid material or void leaving a feasible solution. The Heaviside projection is the emphasis of this thesis and has to the authors knowledge not been implemented in combination with large plastic deformations prior to this thesis. Both the filtering and Heaviside projections will be discussed in more detail in this chapter.

5.1 Filters

The checkerboard patterns that arise in the structure when the SIMP method is used occur due to the artificial stiffness that different elements create when connected by a single node. As mentioned earlier, to overcome the checkerboard patterns filters are applied and with the filters the length scale is introduced. The filter doesn't just remove the checkerboard pattern, through the length scale it also gives control over the minimum thickness or width of the parts in the structure. The filter will create intermediate densities ranging from full material to void over the length scale at the edge of the structure creating a gradient depending which weighted average is used. After filtering the original density no longer holds any physical value for the structure, it's only used as a design variable for the optimization.



(a) The original densities ρ punished by SIMP

(b) Filtered densities $\tilde{\rho}$.

Figure 5.1: Illustration of the densities before and after filtering

5.1.1 Helmholtz PDE filter

The density filter applied in this thesis is implicitly defined as a solution to the Helmholtz PDE with Neumann boundary conditions [4]. The original filtering techniques require large arrays of information from surrounding elements and if parallel computing is used the computations becomes expensive and difficult to implement. The benefit of using a Helmholtz PDE is that it uses the same finite element solver as the structural problem which avoids using extra arrays with information. Since the same solver is used the parallel properties are inherited and the program becomes more computationally efficient [5]. In this work the original density is ρ and the filtered densities will be referred to as $\tilde{\rho}$. The Helmholtz PDE used in this work is defined as

$$-R^2\nabla^2\tilde{\rho} + \tilde{\rho} = \rho \tag{5.1}$$

together with Neumann boundary conditions

$$\frac{\partial\tilde{\rho}}{\partial \boldsymbol{n}} = 0 \tag{5.2}$$

where $\tilde{\rho}$ is the continuous filtered density, while the unfiltered density ρ is piecewise constant, R can be interpreted as the length-scale and ∇^2 is the laplacian. To solve (5.1) the finite element method is used. First (5.1) is multiplied with a weight function v and integrated over the body, it becomes

$$\int_V -\boldsymbol{v}R^2 div(\nabla\tilde{\boldsymbol{\rho}})dV + \int_V \boldsymbol{v}\tilde{\boldsymbol{\rho}}dV = \int_V \boldsymbol{v}dV \tag{5.3}$$

with the Green- Gauss theorem and the Neumann condition.

$$\int_V R^2(\nabla\boldsymbol{v})^T\nabla\tilde{\boldsymbol{\rho}}dV + \int_V \boldsymbol{v}\tilde{\boldsymbol{\rho}}dV = \int_V \boldsymbol{v}\rho dV \tag{5.4}$$

Using the Galerkin-approach where the shape functions is $\boldsymbol{N}$, $\boldsymbol{v} = \boldsymbol{N}\boldsymbol{c}$, $\tilde{\rho} = \boldsymbol{N}\tilde{\boldsymbol{\rho}}$ and $\nabla\tilde{\boldsymbol{\rho}} = \boldsymbol{B}\tilde{\boldsymbol{\rho}}$, equation (5.4) becomes

$$\boldsymbol{c}\left\{\int_V R^2\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}dV\tilde{\boldsymbol{\rho}} + \int_V \boldsymbol{N}^{\boldsymbol{T}}\boldsymbol{N}dV\tilde{\boldsymbol{\rho}} - \int_V \boldsymbol{N}^{\boldsymbol{T}}dV\boldsymbol{\rho}\right\} = 0. \tag{5.5}$$

By denoting the matrices

$$\boldsymbol{K} = \int_V R^2\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}dV$$
$$\boldsymbol{M} = \int_V \boldsymbol{N}^{\boldsymbol{T}}\boldsymbol{N}dV$$
$$\boldsymbol{T} = \int_V \boldsymbol{N}^{\boldsymbol{T}}dV$$

the resulting equation

$$(\boldsymbol{K} + \boldsymbol{M})\tilde{\boldsymbol{\rho}} = \boldsymbol{T}\boldsymbol{\rho} \tag{5.6}$$

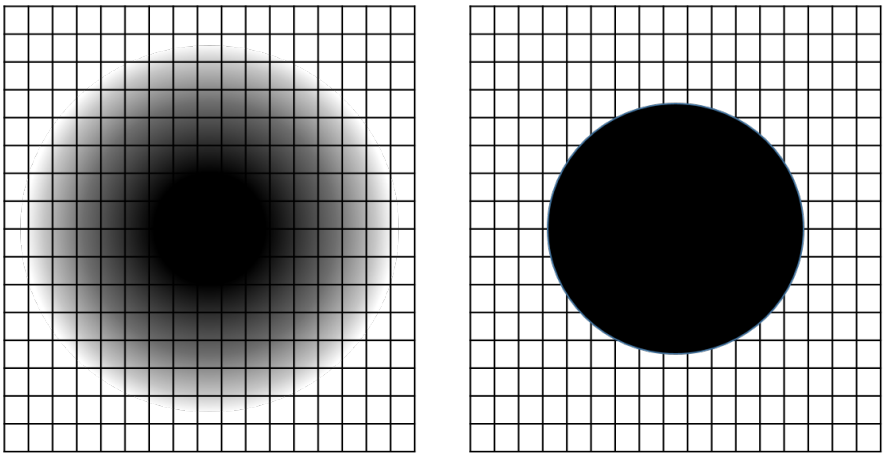
is obtained. For the sensitivity analysis the derivative of equation (5.6) with respect to ρ becomes

$$\frac{\partial \tilde{\rho}}{\partial \rho} = (\boldsymbol{K} + \boldsymbol{M})^{-1}\boldsymbol{T} \tag{5.7}$$

5.2 Thresholding

The density filter makes the solution non-discrete which can be solved by using a Heaviside projection on the filtered densities, see figure 5.2. Most Heaviside functions that are used are approximated as a smooth continuous function [9]. Since the Heaviside function usually is used in the sensitivity analysis it must be a continuous differentiable function. For stable convergence the function starts with almost linear behaviour and then continually moves towards a Heaviside function through the iterations. Therefore there usually exist a built in parameter that dictates the slope of the function. There has however been cases of success with the continuation approach eliminated but has not been applied in this work [2].

The Heaviside projection will be referred to as $\hat{\rho} = H(\tilde{\rho})$. There are many different Heaviside functions that has been tested with linear elastic strains and a few will be shown below.



(a) Filtered densities $\tilde{\rho}$

(b) Heaviside projected densities. $\hat{\rho} = H(\tilde{\rho})$

Figure 5.2: Illustration of the non-discrete filtered densities and the discrete Heaviside projected densities.

5.2.1 Max operating scheme

This Heaviside function was proposed by Guest. K [2].

$$\hat{\rho} = 1 - e^{-\beta\tilde{\rho}} + \tilde{\rho}e^{-\beta} \tag{5.8}$$

Where β is a constant that decides the slope of the function, if β approaches zero the function becomes linear and doesn't affect $\tilde{\rho}$, as it moves towards infinity it becomes a Heaviside step function. It works as a max-operator-based scheme, if the node has zero density then the Heaviside projection becomes zero otherwise it becomes one, see Figure 5.3.

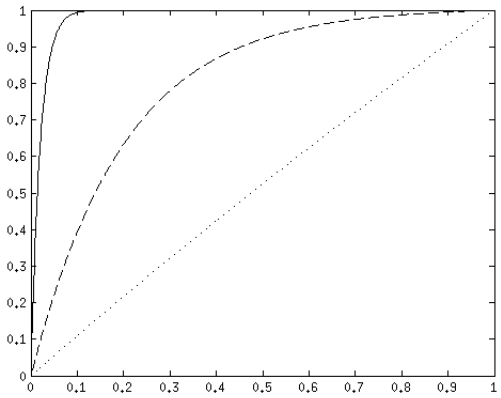


Figure 5.3: The function proposed by K. Guest for different β. Dotted line β=0.5, dashed line β=5 and full line β=50.

For the sensitivity analysis the derivative of (5.8) with respect to $\tilde{\rho}$ becomes

$$\frac{\partial\hat{\rho}}{\partial\tilde{\rho}} = \beta e^{-\beta\tilde{\rho}} + e^{-\beta} \tag{5.9}$$

5.2.2 Min operating scheme

Another similar function was proposed by Sigmund. O [9] who turned the filter to a min-operator-based scheme.

$$\hat{\rho} = e^{-\beta(1-\tilde{\rho})} - (1-\tilde{\rho})e^{-\beta} \tag{5.10}$$

If the density in the node is one then the Heaviside projection becomes one otherwise it becomes zero, see Figure 5.4.

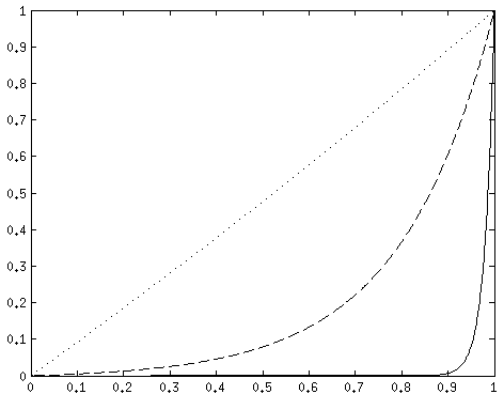


Figure 5.4: The function proposed by O. Sigmund for different β. Dotted line β=0.5, dashed line β=5 and full line β=50.

For the sensitivity analysis the derivative of (5.10) with respect to $\tilde{\rho}$ becomes

$$\frac{\partial \hat{\rho}}{\partial \tilde{\rho}} = \beta e^{\beta(\tilde{\rho}-1)} + e^{-\beta}. \tag{5.11}$$

The difference between the max and the min operating scheme is that the projection removes material from structure in the min scheme as opposed to add material in the max scheme. In the min operating scheme when material is removed, the structure corresponding to the design variable ρ is much thicker, therefore providing more stability during the iterations [9]. Both Heaviside projections above are non-volume preserving, which cause worse convergence towards the optimal solution than a volume preserving projection [15].

5.2.3 Volume preserving Heaviside projection

A volume preserving projection was proposed by S. Xu et al. [15] in which both filters above are combined.

$$\hat{\rho} = \begin{cases} \eta[e^{-\beta(1-\tilde{\rho}/\eta)} - (1-\tilde{\rho}/\eta)e^{-\beta}] & 0 \leq \tilde{\rho} \leq \eta \\ (1-\eta)[1 - e^{-\beta(\tilde{\rho}-\eta)/(1-\eta)} + (\tilde{\rho}-\eta)e^{-\beta}/(1-\eta)] + \eta & \eta < \tilde{\rho} \leq 1 \end{cases} \tag{5.12}$$

Where β determines the slope of the function and η is a volume preserving parameter that move the center of the two graphs towards the corners along the linear line in order for the volume to remain the same before and after filtering. There is a drawback to varying η and keep the volume preserved, when shifting the center of the graph control is lost over the length-scale [2].

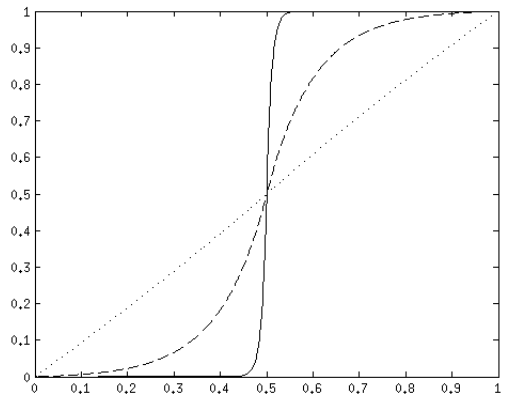


Figure 5.5: The function proposed by S. Xu for different β. Dotted line β=0.5, dashed line β=5 and full line β=50.

For the sensitivity analysis the derivative of (5.12) with respect to $\tilde{\rho}$ becomes

$$\frac{\partial \hat{\rho}}{\partial \tilde{\rho}} = \begin{cases} \beta e^{-\beta(1-\tilde{\rho}/\eta)} + e^{-\beta} & 0 \leq \tilde{\rho} \leq \eta \\ \beta e^{-\beta(\tilde{\rho}-\eta)/(1-\eta)} + e^{-\beta} & \eta < \tilde{\rho} \leq 1 \end{cases} \tag{5.13}$$

Chapter 6

Sensitivities

As mentioned in the previous chapter the derivatives of the objective function and all constraint functions with respect to the design variables are needed when solving the optimization problem. These derivatives are called sensitivities and the solving process is called sensitivity analysis. The two most common ways of calculating the sensitivities are numerically or analytically.

6.1 Numerical Methods

In numerical sensitivity analysis the sensitivities are calculated by finite differences. There exist different ways of doing this, e.g. forward differences or central differences. The forward differences

$$\frac{\partial g_i(\boldsymbol{\phi}^k)}{\partial \phi_j} \approx \mathrm{D}_f = \frac{g_i(\boldsymbol{\phi}^k + h\boldsymbol{e}_j) - g_i(\boldsymbol{\phi}^k)}{h} \tag{6.1}$$

where $e_j = [0, ..., 0, 1, 0, ..., 0]^T$, e_j is 1 on index j and zero otherwise. There is a major issue with the forward differences and that is to find a suitable h. Ideally h should be as small as possible but that leads to numerical errors due to cancellation and if the h is to big it becomes a bad approximation. The numerical methods becomes approximations and are more computationally expensive than the analytical methods, the benefit of using numerical methods are that they are very easy to implement. The only use of numerical differentiation in this thesis has been to check that the analytical method has been implemented correctly.

6.1.1 Analytical Methods

The two most common analytical methods are the direct and adjoint method. Both methods has their benefits, if the problem has more design variables than constraint functions then the adjoint method is more efficient and otherwise the direct method is more efficient. Since the design variables outnumber the constraint functions in topology optimization by far, the adjoint method is used.

6.1.2 Adjoint method

This thesis handles a steady-state coupled non-linear system, since the response from the structure depends on the design variable ϕ_i, the residuals $\boldsymbol{R}$ and $\boldsymbol{C}$ (3.2) are rewritten as

$$\begin{aligned} {}^{n+1}\boldsymbol{R}({}^{n+1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n+1}\boldsymbol{v}(\boldsymbol{\phi}),{}^{n}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}) = \boldsymbol{0} \\ {}^{n+1}\boldsymbol{C}({}^{n+1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n+1}\boldsymbol{v}(\boldsymbol{\phi}),{}^{n}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}) = \boldsymbol{0} \end{aligned} \tag{6.2}$$

where ${}^{i}\boldsymbol{u}(\boldsymbol{\phi})$ and ${}^{i}\boldsymbol{v}(\boldsymbol{\phi})$ are given by the residuals [6]. The response function F can in general be defined as

$$F(\boldsymbol{\phi}) = G({}^{M}\boldsymbol{u}(\boldsymbol{\phi}),{}^{M-1}\boldsymbol{u}(\boldsymbol{\phi}),...,{}^{1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{M}\boldsymbol{v}(\boldsymbol{\phi}),{}^{M-1}\boldsymbol{v}(\boldsymbol{\phi}),...,{}^{0}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}). \tag{6.3}$$

In order to obtain the sensitivity of the system, the objective function is differentiated with respect to the design variables ϕ_i.

$$\frac{DF}{D\phi_i} = \sum_{n=0}^{M}\left(\frac{\partial G}{\partial {}^{n}\boldsymbol{u}}\frac{D^n\boldsymbol{u}}{D\phi_i} + \frac{\partial G}{\partial {}^{n}\boldsymbol{v}}\frac{D^n\boldsymbol{v}}{D\phi_i}\right) + \frac{\partial G}{\partial \phi_i} \tag{6.4}$$

where $\frac{D^n\boldsymbol{u}}{D\phi_i}$ and $\frac{D^n\boldsymbol{v}}{D\phi_i}$ are the implicit derivatives. Since both residuals depend on the design variables the implicit derivatives become costly to compute. The idea by using the adjoint method is to eliminate the implicit derivatives. By introducing the arbitrary adjoint vectors ${}^{n}\boldsymbol{\lambda}$ and ${}^{n}\boldsymbol{\gamma}$ the augmented response functional can be formed [6]

$$\begin{aligned} \hat{F}(\boldsymbol{\phi}) =& G({}^{M}\boldsymbol{u}(\boldsymbol{\phi}),{}^{M}\boldsymbol{v}(\boldsymbol{\phi}),{}^{M-1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{M-1}\boldsymbol{v}(\boldsymbol{\phi}),...,{}^{1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{1}\boldsymbol{v}(\boldsymbol{\phi}),{}^{0}\boldsymbol{u}(\boldsymbol{\phi}),{}^{0}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}) \\ &- \sum_{n=1}^{M}{}^{n}\boldsymbol{\lambda}^{n}\boldsymbol{R}({}^{n}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n-1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n}\boldsymbol{v}(\boldsymbol{\phi}),{}^{n-1}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}) \\ &- \sum_{n=1}^{M}{}^{n}\boldsymbol{\gamma}^{n}\boldsymbol{C}({}^{n}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n-1}\boldsymbol{u}(\boldsymbol{\phi}),{}^{n}\boldsymbol{v}(\boldsymbol{\phi}),{}^{n-1}\boldsymbol{v}(\boldsymbol{\phi}),\boldsymbol{\phi}) \end{aligned} \tag{6.5}$$

The augmented functional $\hat{F}$ is equal to the response functional G since the residuals in equation (6.2) holds. The sensitivity is then obtained by differentiating the augmented response functional from equation (6.5) with respect to the design variables.

$$\frac{D\hat{F}}{D\phi_i} = \frac{D\hat{F}_E}{D\phi_i} + \sum_{n=1}^{M}\left(\frac{D^n\hat{F}_I}{D\phi_i}\right)_{\lambda} + \sum_{n=1}^{M}\left(\frac{D^n\hat{F}_I}{D\phi_i}\right)_{\gamma} \tag{6.6}$$

where the E denotes that the term is explicit and I implicit. The explicit derivative is defined as follows

$$\begin{aligned} \frac{D\hat{F}_E}{D\phi_i} =& \frac{\partial G}{\partial \phi_i} - \sum_{n=1}^{M}{}^{n}\boldsymbol{\lambda}\frac{\partial^n\boldsymbol{R}}{\partial\phi_i} - \sum_{n=1}^{M}{}^{n}\boldsymbol{\gamma}\frac{\partial^n\boldsymbol{C}}{\partial\phi_i} + \frac{\partial G}{\partial^0\boldsymbol{u}}\frac{D^0\boldsymbol{u}}{D\phi_i} + \frac{\partial G}{\partial^0\boldsymbol{v}}\frac{D^0\boldsymbol{v}}{D\phi_i} \\ &- {}^{1}\boldsymbol{\lambda}\frac{\partial^1\boldsymbol{R}}{\partial^0\boldsymbol{u}}\frac{D^0\boldsymbol{u}}{D\phi_i} - {}^{1}\boldsymbol{\lambda}\frac{\partial^1\boldsymbol{R}}{\partial^0\boldsymbol{v}}\frac{D^0\boldsymbol{v}}{D\phi_i} - {}^{1}\boldsymbol{\gamma}\frac{\partial^1\boldsymbol{C}}{\partial^0\boldsymbol{u}}\frac{D^0\boldsymbol{u}}{D\phi_i} - {}^{1}\boldsymbol{\gamma}\frac{\partial^1\boldsymbol{C}}{\partial^0\boldsymbol{v}}\frac{D^0\boldsymbol{v}}{D\phi_i} \end{aligned} \tag{6.7}$$

The implicit derivatives are given by

$$
\left.
\begin{aligned}
\left(\frac{D^1\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\lambda}} &= -\left[{}^1\boldsymbol{\lambda}\frac{\partial^1\boldsymbol{R}}{\partial^1\boldsymbol{u}} + {}^1\boldsymbol{\gamma}\frac{\partial^1\boldsymbol{C}}{\partial^1\boldsymbol{u}} + {}^2\boldsymbol{\lambda}\frac{\partial^2\boldsymbol{R}}{\partial^1\boldsymbol{u}} + {}^2\boldsymbol{\gamma}\frac{\partial^2\boldsymbol{C}}{\partial^1\boldsymbol{u}} - \frac{\partial G}{\partial^1\boldsymbol{u}}\right]\frac{D^1\boldsymbol{u}}{D\phi_i} \\
\left(\frac{D^1\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\gamma}} &= -\left[{}^1\boldsymbol{\lambda}\frac{\partial^1\boldsymbol{R}}{\partial^1\boldsymbol{v}} + {}^1\boldsymbol{\gamma}\frac{\partial^1\boldsymbol{C}}{\partial^1\boldsymbol{v}} + {}^2\boldsymbol{\lambda}\frac{\partial^2\boldsymbol{R}}{\partial^1\boldsymbol{v}} + {}^2\boldsymbol{\gamma}\frac{\partial^2\boldsymbol{C}}{\partial^1\boldsymbol{v}} - \frac{\partial G}{\partial^1\boldsymbol{v}}\right]\frac{D^1\boldsymbol{v}}{D\phi_i} \\
\left(\frac{D^2\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\lambda}} &= -\left[{}^2\boldsymbol{\lambda}\frac{\partial^2\boldsymbol{R}}{\partial^2\boldsymbol{u}} + {}^2\boldsymbol{\gamma}\frac{\partial^2\boldsymbol{C}}{\partial^2\boldsymbol{u}} + {}^3\boldsymbol{\lambda}\frac{\partial^3\boldsymbol{R}}{\partial^2\boldsymbol{u}} + {}^3\boldsymbol{\gamma}\frac{\partial^3\boldsymbol{C}}{\partial^2\boldsymbol{u}} - \frac{\partial G}{\partial^2\boldsymbol{u}}\right]\frac{D^2\boldsymbol{u}}{D\phi_i} \\
\left(\frac{D^2\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\gamma}} &= -\left[{}^2\boldsymbol{\lambda}\frac{\partial^2\boldsymbol{R}}{\partial^2\boldsymbol{v}} + {}^2\boldsymbol{\gamma}\frac{\partial^2\boldsymbol{C}}{\partial^2\boldsymbol{v}} + {}^3\boldsymbol{\lambda}\frac{\partial^3\boldsymbol{R}}{\partial^2\boldsymbol{v}} + {}^3\boldsymbol{\gamma}\frac{\partial^3\boldsymbol{C}}{\partial^2\boldsymbol{v}} - \frac{\partial G}{\partial^2\boldsymbol{v}}\right]\frac{D^2\boldsymbol{v}}{D\phi_i} \\
&\ldots \\
\left(\frac{D^{M-1}\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\lambda}} &= -\left[{}^{M-1}\boldsymbol{\lambda}\frac{\partial^{M-1}\boldsymbol{R}}{\partial^{M-1}\boldsymbol{u}} + {}^{M-1}\boldsymbol{\gamma}\frac{\partial^{M-1}\boldsymbol{C}}{\partial^{M-1}\boldsymbol{u}} + \right. \\
&\qquad \left. {}^{M}\boldsymbol{\lambda}\frac{\partial^{M}\boldsymbol{R}}{\partial^{M-1}\boldsymbol{u}} + {}^{M}\boldsymbol{\gamma}\frac{\partial^{M}\boldsymbol{C}}{\partial^{M-1}\boldsymbol{u}} - \frac{\partial G}{\partial^{M-1}\boldsymbol{u}}\right]\frac{D^{M-1}\boldsymbol{u}}{D\phi_i} \\
\left(\frac{D^{M-1}\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\gamma}} &= -\left[{}^{M-1}\boldsymbol{\lambda}\frac{\partial^{M-1}\boldsymbol{R}}{\partial^{M-1}\boldsymbol{v}} + {}^{M-1}\boldsymbol{\gamma}\frac{\partial^{M-1}\boldsymbol{C}}{\partial^{M-1}\boldsymbol{v}} + \right. \\
&\qquad \left. {}^{M}\boldsymbol{\lambda}\frac{\partial^{M}\boldsymbol{R}}{\partial^{M-1}\boldsymbol{v}} + {}^{M}\boldsymbol{\gamma}\frac{\partial^{M}\boldsymbol{C}}{\partial^{M-1}\boldsymbol{v}} - \frac{\partial G}{\partial^{M-1}\boldsymbol{v}}\right]\frac{D^{M-1}\boldsymbol{v}}{D\phi_i} \\
\left(\frac{D^{M}\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\lambda}} &= -\left[{}^{M}\boldsymbol{\lambda}\frac{\partial^{M}\boldsymbol{R}}{\partial^{M}\boldsymbol{u}} + {}^{M}\boldsymbol{\gamma}\frac{\partial^{M}\boldsymbol{C}}{\partial^{M}\boldsymbol{u}} - \frac{\partial G}{\partial^{M}\boldsymbol{u}}\right]\frac{D^{M}\boldsymbol{u}}{D\phi_i} \\
\left(\frac{D^{M}\hat{F}_I}{D\phi_i}\right)_{\boldsymbol{\gamma}} &= -\left[{}^{M}\boldsymbol{\lambda}\frac{\partial^{M}\boldsymbol{R}}{\partial^{M}\boldsymbol{v}} + {}^{M}\boldsymbol{\gamma}\frac{\partial^{M}\boldsymbol{C}}{\partial^{M}\boldsymbol{v}} - \frac{\partial G}{\partial^{M}\boldsymbol{v}}\right]\frac{D^{M}\boldsymbol{v}}{D\phi_i}
\end{aligned}
\right\} \tag{6.8}
$$

The implicit derivatives $\frac{D^n\boldsymbol{u}}{D\phi_i}$ and $\frac{D^n\boldsymbol{v}}{D\phi_i}$ in equation (6.8) can be eliminated, by choosing the adjoint vectors in a way so the terms within the brackets vanish. It can be seen that the implicit derivatives depends on the residuals and adjoint vectors of its load step and the next for all derivatives except the last step M. Therefore the sensitivity analysis starts with solving the last step M. By treating $\boldsymbol{\gamma}$ as a function of $\boldsymbol{\lambda}$ the equation system (6.9) can be formed. Since the sensitivity analysis starts from the last step it cannot be solved at the same time as the primary structural problem and must be solved after. The equations that needs to be solved are

$$
\begin{aligned}
M:\quad & \left\{\begin{array}{l}
\left[\dfrac{\partial^M \boldsymbol{R}}{\partial^M \boldsymbol{u}} - \dfrac{\partial^M \boldsymbol{R}}{\partial^M \boldsymbol{v}}\left(\dfrac{\partial^M \boldsymbol{C}}{\partial^M \boldsymbol{v}}\right)^{-1}\dfrac{\partial^M \boldsymbol{C}}{\partial^M \boldsymbol{u}}\right]^T {}^M\boldsymbol{\lambda} = \\
\qquad\qquad - \left[\left(\dfrac{\partial^M \boldsymbol{C}}{\partial^M \boldsymbol{v}}\right)^{-1}\dfrac{\partial^M \boldsymbol{C}}{\partial^M \boldsymbol{u}}\right]^T \dfrac{\partial G}{\partial^M \boldsymbol{v}} + \dfrac{\partial G}{\partial^M \boldsymbol{u}} \\
{}^M\boldsymbol{\gamma} = -\left(\dfrac{\partial^M \boldsymbol{C}}{\partial^M \boldsymbol{v}}\right)^{-T}\left[\left(\dfrac{\partial^M \boldsymbol{R}}{\partial^M \boldsymbol{v}}\right)^T {}^M\boldsymbol{\lambda} - \dfrac{\partial G}{\partial^M \boldsymbol{v}}\right]
\end{array}\right. \\
M-1:\quad & \left\{\begin{array}{l}
\left[\dfrac{\partial^{M-1} \boldsymbol{R}}{\partial^{M-1} \boldsymbol{u}} - \dfrac{\partial^{M-1} \boldsymbol{R}}{\partial^{M-1} \boldsymbol{v}}\left(\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^{-1}\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{u}}\right]^T {}^{M-1}\boldsymbol{\lambda} = \\
- \left[\dfrac{\partial^M \boldsymbol{R}}{\partial^{M-1} \boldsymbol{u}} - \dfrac{\partial^M \boldsymbol{R}}{\partial^{M-1} \boldsymbol{v}}\left(\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^{-1}\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{u}}\right]^T {}^M\boldsymbol{\lambda} \\
- \left[\dfrac{\partial^M \boldsymbol{C}}{\partial^{M-1} \boldsymbol{u}} - \dfrac{\partial^M \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\left(\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^{-1}\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{u}}\right]^T {}^M\boldsymbol{\gamma} \\
- \left[\left(\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^{-1}\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{u}}\right]^T \dfrac{\partial G}{\partial^{M-1} \boldsymbol{v}} + \dfrac{\partial G}{\partial^{M-1} \boldsymbol{u}} \\
{}^{M-1}\boldsymbol{\gamma} = -\left(\dfrac{\partial^{M-1} \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^{-T}\left[\left(\dfrac{\partial^{M-1} \boldsymbol{R}}{\partial^{M-1} \boldsymbol{v}}\right)^T {}^{M-1}\boldsymbol{\lambda} + \right. \\
\qquad\qquad \left.\left(\dfrac{\partial^M \boldsymbol{R}}{\partial^{M-1} \boldsymbol{v}}\right)^T {}^M\boldsymbol{\lambda} + \left(\dfrac{\partial^M \boldsymbol{C}}{\partial^{M-1} \boldsymbol{v}}\right)^T {}^M\boldsymbol{\gamma} - \dfrac{\partial G}{\partial^{M-1} \boldsymbol{v}}\right]
\end{array}\right. \\
& \ldots \\
2:\quad & \left\{\begin{array}{l}
\left[\dfrac{\partial^2 \boldsymbol{R}}{\partial^2 \boldsymbol{u}} - \dfrac{\partial^2 \boldsymbol{R}}{\partial^2 \boldsymbol{v}}\left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{u}}\right]^T {}^2\boldsymbol{\lambda} = \\
- \left[\dfrac{\partial^3 \boldsymbol{R}}{\partial^2 \boldsymbol{u}} - \dfrac{\partial^3 \boldsymbol{R}}{\partial^2 \boldsymbol{v}}\left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{u}}\right]^T {}^3\boldsymbol{\lambda} \\
- \left[\dfrac{\partial^3 \boldsymbol{C}}{\partial^2 \boldsymbol{u}} - \dfrac{\partial^3 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{u}}\right]^T {}^3\boldsymbol{\gamma} \\
- \left[\left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{u}}\right]^T \dfrac{\partial G}{\partial^2 \boldsymbol{v}} + \dfrac{\partial G}{\partial^2 \boldsymbol{u}} \\
{}^2\boldsymbol{\gamma} = -\left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^{-T}\left[\left(\dfrac{\partial^2 \boldsymbol{R}}{\partial^2 \boldsymbol{v}}\right)^T {}^2\boldsymbol{\lambda} + \right. \\
\qquad\qquad \left.\left(\dfrac{\partial^3 \boldsymbol{R}}{\partial^2 \boldsymbol{v}}\right)^T {}^3\boldsymbol{\lambda} + \left(\dfrac{\partial^3 \boldsymbol{C}}{\partial^2 \boldsymbol{v}}\right)^T {}^3\boldsymbol{\gamma} - \dfrac{\partial G}{\partial^2 \boldsymbol{v}}\right]
\end{array}\right.
\end{aligned}
\tag{6.9}
$$

$$
1: \quad \left\{ \begin{array}{l}
\left[\dfrac{\partial^1 \boldsymbol{R}}{\partial^1 \boldsymbol{u}} - \dfrac{\partial^1 \boldsymbol{R}}{\partial^1 \boldsymbol{v}}\left(\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{u}}\right]^T {}^1\boldsymbol{\lambda} = \\
-\left[\dfrac{\partial^2 \boldsymbol{R}}{\partial^1 \boldsymbol{u}} - \dfrac{\partial^2 \boldsymbol{R}}{\partial^1 \boldsymbol{v}}\left(\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{u}}\right]^T {}^2\boldsymbol{\lambda} \\
-\left[\dfrac{\partial^2 \boldsymbol{C}}{\partial^1 \boldsymbol{u}} - \dfrac{\partial^2 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\left(\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{u}}\right]^T {}^2\boldsymbol{\gamma} \\
-\left[\left(\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^{-1}\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{u}}\right]^T \dfrac{\partial G}{\partial^1 \boldsymbol{v}} + \dfrac{\partial G}{\partial^1 \boldsymbol{u}} \\
{}^1\boldsymbol{\gamma} = -\left(\dfrac{\partial^1 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^{-T}\left[\left(\dfrac{\partial^1 \boldsymbol{R}}{\partial^1 \boldsymbol{v}}\right)^T {}^1\boldsymbol{\lambda} + \right. \\
\qquad\qquad \left.\left(\dfrac{\partial^2 \boldsymbol{R}}{\partial^1 \boldsymbol{v}}\right)^T {}^2\boldsymbol{\lambda} + \left(\dfrac{\partial^2 \boldsymbol{C}}{\partial^1 \boldsymbol{v}}\right)^T {}^2\boldsymbol{\gamma} - \dfrac{\partial G}{\partial^1 \boldsymbol{v}}\right]
\end{array} \right\}
$$

The equation is solved by first calculating ${}^M\boldsymbol{\lambda}$ and then ${}^M\boldsymbol{\gamma}$. When the last step is known it is possible to calculate ${}^{M-1}\boldsymbol{\lambda}$ and ${}^{M-1}\boldsymbol{\gamma}$, then continue this process until the initial state is solved. Once all equations are solved the sensitivities can be calculated using (6.6), for the complete in-depth method the reader is referred to [6].

6.2 Sensitivity of the volume constraint

The sensitivity of the constraint function or as in this case the volume preserving constraint of the structure needs to be evaluated as well. In the MMA solving process the derivatives of the functions $g_1(\boldsymbol{\phi}^k)$ with respect to the design variable ϕ is needed

$$\frac{\partial g_1(\boldsymbol{\phi}^k)}{\partial \phi_j}$$

where k is the current iteration and j the index of the design variable. The constraint used in this thesis is a mass constraint, where ρ_0 is the element density that is allowed, ρ_e the design variable is the current density of the element and V_e the volume of the element

$$g_1 = \sum_{e=1}^{nelm} \rho_e V_e - \rho_0 \sum_{e=1}^{nelm} V_e \leq 0 \tag{6.10}$$

where $nelm$ stands for the number of elements. Since ρ_0 is a constant the derivative becomes

$$\frac{\partial g_1(\boldsymbol{\rho}^k)}{\partial \rho_e} = V_e. \tag{6.11}$$

With a density filter applied to the design variable the constraint is instead written as

$$g_1 = \int_V \tilde{\rho}_e dV - \rho_0 \sum_{e=1}^{nelm} V_e \leq 0 \tag{6.12}$$

where $\tilde{\rho}$ is the filtered density with nodal values and V is referring to the volume of the body. The derivative becomes

$$\frac{\partial g_1(\tilde{\boldsymbol{\rho}}^k)}{\partial \rho_e} = \int_V \frac{\partial \tilde{\rho}}{\partial \tilde{\rho}_\alpha} dV \frac{\partial \tilde{\rho}^\alpha}{\partial \rho^e} \tag{6.13}$$

where $\tilde{\rho} = \tilde{\boldsymbol{\rho}}\boldsymbol{N}$ and $\boldsymbol{N}$ is the shapefunctions, α refers to the nodal value and e to the element value, if an additional Heaviside projection is applied the constraint is then written as

$$g_1 : \int_V \hat{\rho}_e dV - \rho_0 \sum_{e=1}^{nelm} V_e \leq 0 \tag{6.14}$$

and the derivative is

$$\frac{\partial g_1(\hat{\boldsymbol{\rho}}^k)}{\partial \rho_e} = \int_V \frac{\partial \hat{\rho}}{\partial \tilde{\rho}} \frac{\partial \tilde{\rho}}{\partial \tilde{\rho}_\alpha} dV \frac{\partial \tilde{\rho}^\alpha}{\partial \rho^e} \tag{6.15}$$

Chapter 7

Results

7.1 Maximizing the absorbed plastic work

The objective in this topology optimization problem is to maximize the absorbed plastic work (W^p) in the structure. The load case is presented in fig 7.1, the sides of the structure is fastened so zero displacement occur along the side boundaries. Due to symmetry only the right half of the structure is analyzed in the simulations. The load is displacement controlled i.e. forced displacement is applied and then the internal forces are calculated.

The problem is a plane 2D problem, the thickness of the structure doesn't affect the solution. The mesh consists of 11700 eight node brick elements although 2D elements would have been sufficient. The optimization problem is described in equation (7.1) where ρ_0 is the initial density of each element spread out homogeneously throughout the structure and dictates the mass constraint. In the

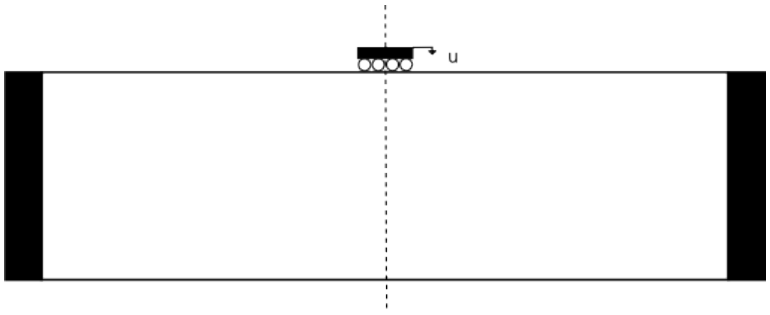


Figure 7.1: Description of the load case used in the simulations. The cantilever beam has a width of 20mm and the height is 5mm. The width where the displacement is applied is 2.4mm.

following simulations the initial density has been set to $\rho_0 = 0.5$

$$\begin{cases} \min\ (-W_p(\boldsymbol{\rho})) \\ \text{s.t.} \begin{cases} \displaystyle\int_V \hat{\rho}_e dV - \rho_0 \sum_{e=1}^{nelm} V_e \leq 0 \\ 0 < \rho_j \leq 1 \qquad\qquad\qquad\qquad j = 1...nelm \end{cases} \end{cases} \tag{7.1}$$

and the material parameters that are used are the following

$$\begin{aligned} K &= 164 \text{ GPa}, \\ G &= 80 \text{ GPa}, \\ \sigma_{y0} &= 400 \text{ MPa}, \\ H &= 18 \text{ GPa}. \end{aligned}$$

The SIMP (Solid isotropic material penalization) applied changes the material properties as

$$\begin{aligned} K &= \hat{\rho_i}^3 K, \\ G &= \hat{\rho_i}^3 G, \\ \sigma_{y0} &= \hat{\rho_i}^2 \sigma_{y0}, \\ H &= \hat{\rho_i}^3 H. \end{aligned}$$

The Heaviside function applied in the simulations below is the one developed by S. Xu. et al. [15] and is shown in equation (5.12). The η in the function is held at a constant value of 0.5. Two different update routines of β has been applied, one linearly and one exponentially increasing update scheme. The linear update is

$$\beta = \beta + 1$$

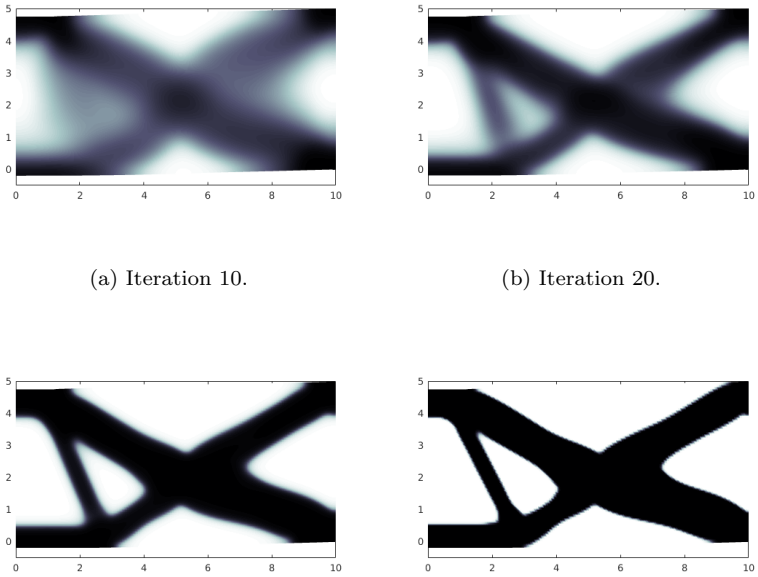
and is increasing every 10th optimization loop. The exponential update is

$$\beta = \beta * 2$$

which is updated every 20th optimization loop. In both cases β has a starting value of 0.1. The length parameter R in the Helmholtz density filter is set as $R = 0.2mm$. The step length depends on the convergence of the problem, the aim is to reach equilibrium in four Newton-Raphson iterations and the step length increase or decrease depending on the previous equilibrium.

7.2 0.25mm displacement with linear β

In this section the maximum displacement is 0.25mm and it's applied in every optimization loop.



(a) Iteration 10.

(b) Iteration 20.

(c) Iteration 50.

(d) Iteration 200.

Figure 7.2: Evolution of the optimization.

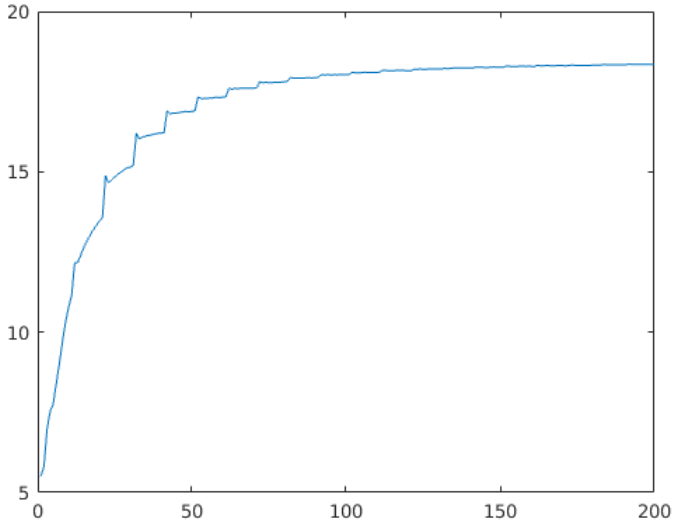


Figure 7.3: Objective function.

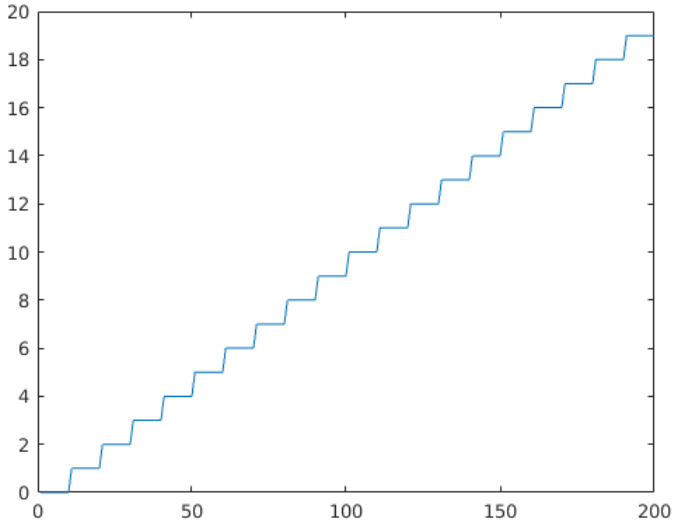
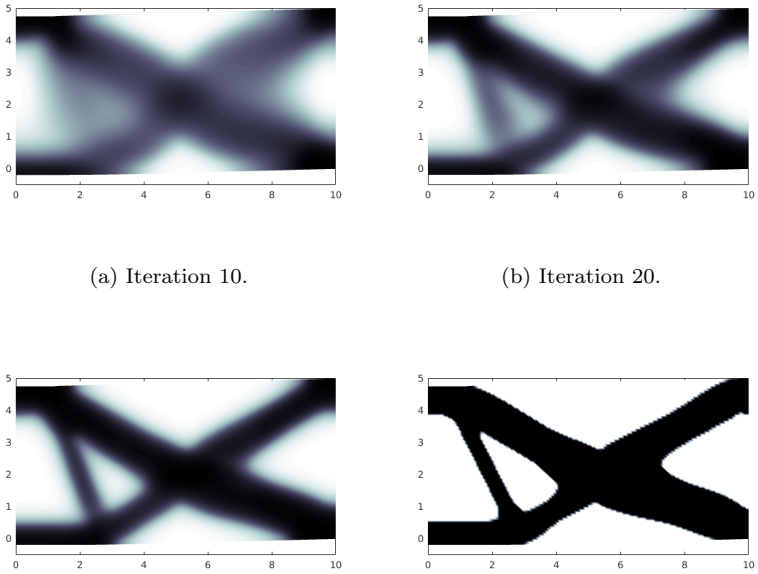


Figure 7.4: Evolution of β.

7.3 0.25mm displacement with exponential β

The load is the same as in the previous case but the exponential updating scheme of β is applied.



(a) Iteration 10.

(b) Iteration 20.

(c) Iteration 50.

(d) Iteration 200.

Figure 7.5: Evolution of the optimization.

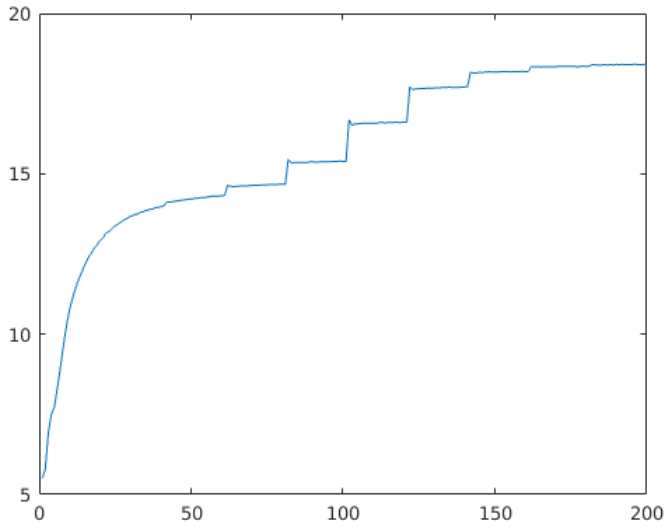


Figure 7.6: objective function.

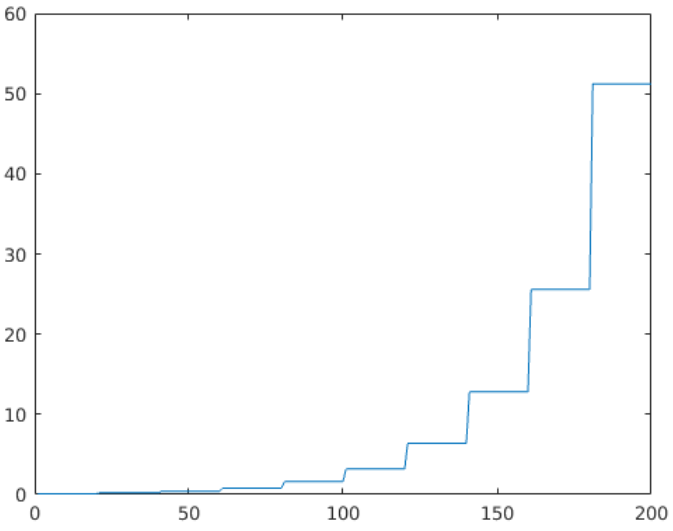


Figure 7.7: Evolution of β.

7.4 0.25mm displacement without Heaviside function

In this section the Heaviside variable β has a constant value of zero, since the Heaviside function becomes linear it doesn't affect the solution at all.

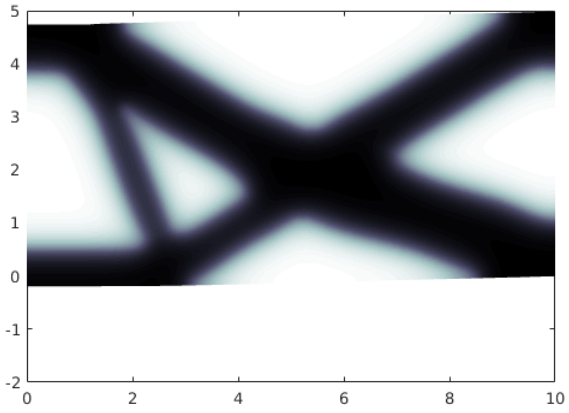


Figure 7.8: Iteration 200.

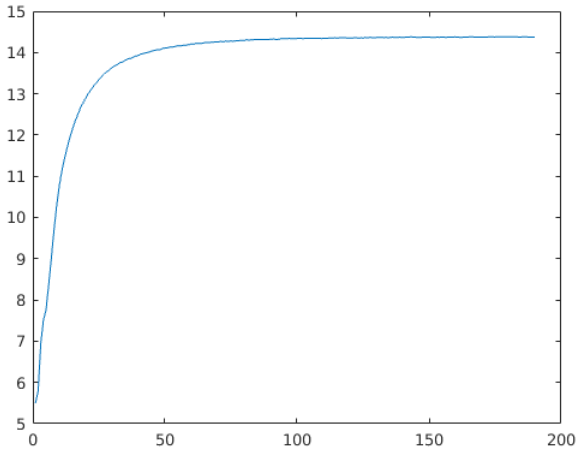
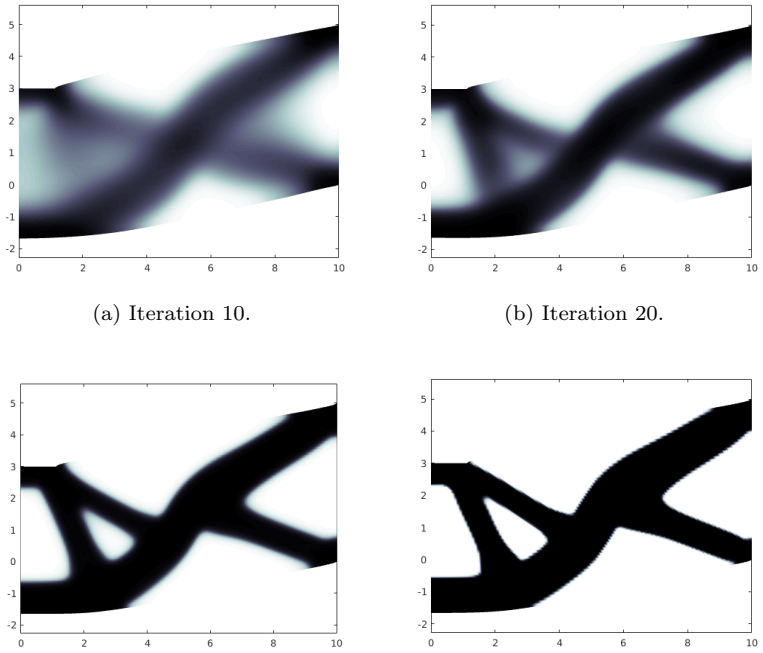


Figure 7.9: Objective function

7.5 2mm displacement with linear β case 1

When the maximum displacement is more than 0.5mm the displacement has to be ramped up or it will take a large number of steps before reaching the final displacement. In this case the displacement reaches 2mm in the fourth loop, increasing 0.5mm in each loop.



(a) Iteration 10.

(b) Iteration 20.

(c) Iteration 50.

(d) Iteration 200.

Figure 7.10: Evolution of the optimization.

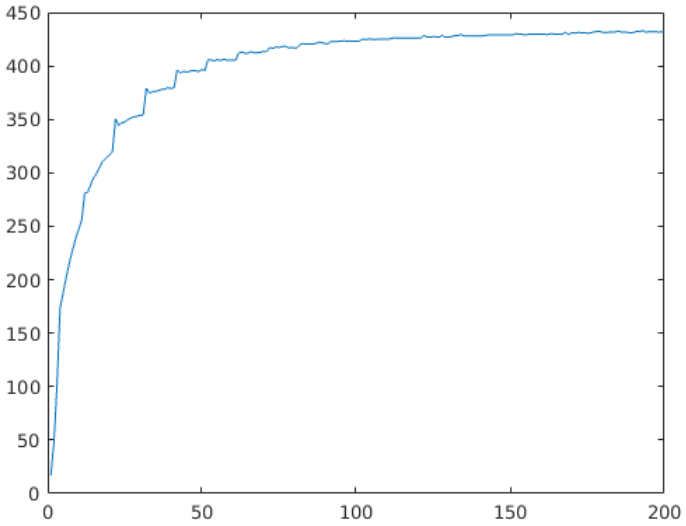


Figure 7.11: Objective function.

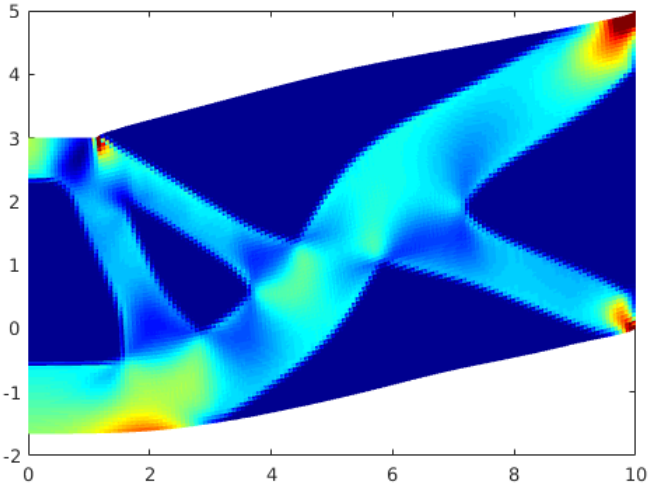


Figure 7.12: Distribution of plastic work in the structure.

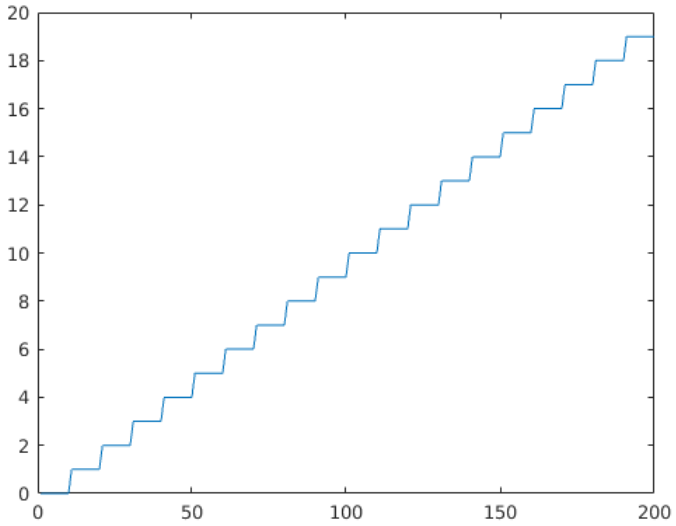
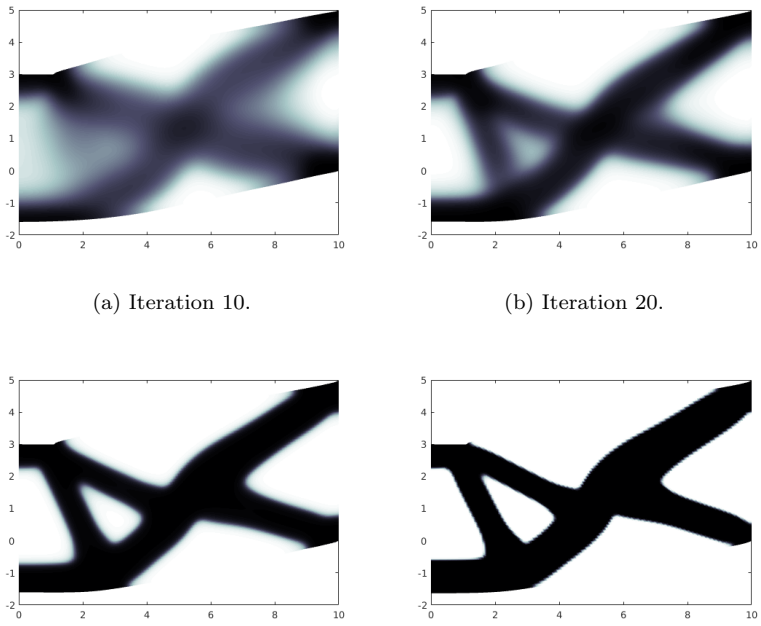


Figure 7.13: evolution of beta.

7.6 2mm displacement with linear β case 2

In this section the displacement has increased during the first eight optimization loops in small steps, reaching 2mm displacement in loop eight.



(a) Iteration 10.

(b) Iteration 20.

(c) Iteration 50.

(d) Iteration 200.

Figure 7.14: Evolution of the optimization.

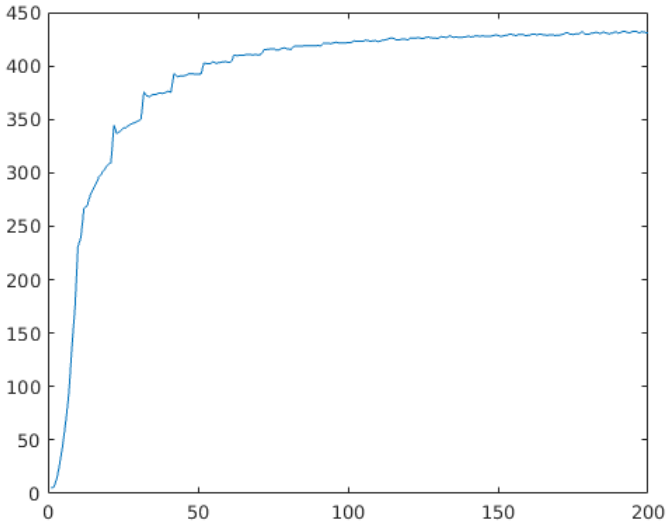


Figure 7.15: Objective function.

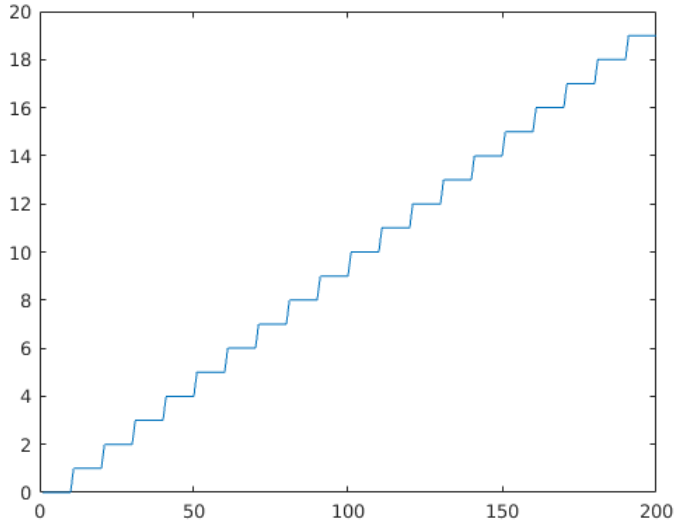


Figure 7.16: Evolution of β.

7.7 2mm displacement without Heaviside function

This section is without the Heaviside function, the displacement is applied in the same manner as in the 2mm displacement case 2 with the Heaviside function over the ten first loops.

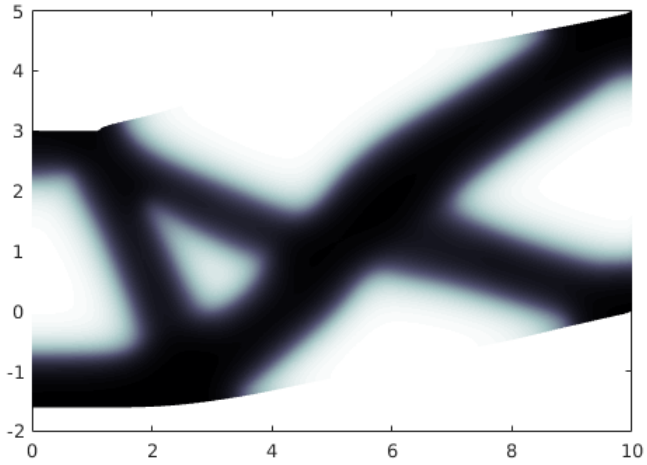


Figure 7.17: Iteration 200.

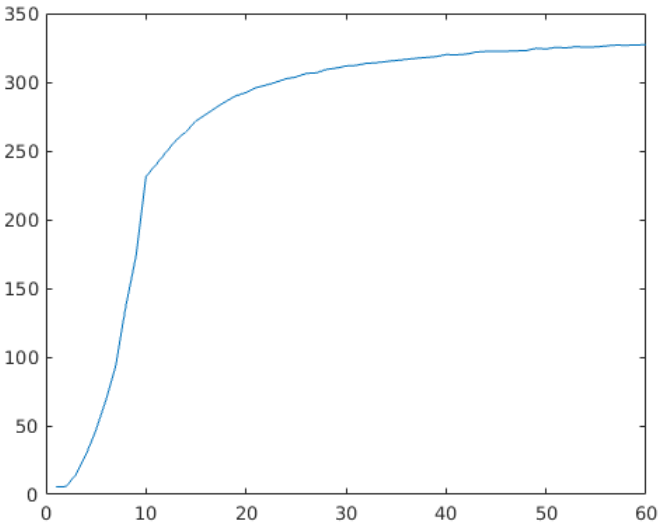


Figure 7.18: Objective function.

Chapter 8

Discussion

When trying different Heaviside functions it was early decided upon the filter by S. Xu et al. [15], the other filters from K. Guest and O. Sigmund didn't converge properly and was therefore abandoned in this thesis. The updating scheme of β has a big impact on the final solution. With the quite aggressive linear β updating scheme the Heaviside function has a dominant role before the structure has settled on the optimal solution. This can cause the Heaviside function to remove the supporting bar to the left in the structure. With the exponential β updating scheme, the Heaviside function doesn't affect the optimization as much in the early stages thus allowing the structure to settle. When comparing the results where the Heaviside function has been implemented to the results with only the Helmholtz filter fig. 7.8, has been applied, a check can be made if the Heaviside function has altered the solution. The solutions does not have to be the same, but if they do differ it should be investigated further. The displacement of 0.25 mm should be regarded as small and the plastic work in the bar is pretty low. When looking at the larger displacement of 2mm the plastic work is larger and the structure moves faster towards the optimal structure. This makes the choice of method in updating β less critical. The choice of updating scheme depends on the individual problem but it should be kept in mind that it can affect the outcome.

When making a comparison to the updating schemes used in the articles of S. Xu, K. Guest and O. Sigmund [15], [2], [9] which all use small linear elastic strains. The Heaviside function is often updated exponentially and very slowly, every 50th iteration or when the problem has converged and the densities doesn't change. This approach is quite conservative and cause the program to run more iterations and increase the computing time a lot, especially for problems with large, non-linear plastic strains that already is relatively slow.

Comparing the 2 mm displacement cases, they don't show the same solution. In figure 7.14 the top bar has not drifted as far from the early cross as in fig 7.10. Since the forced displacement hasn't been applied in the same way it shouldn't give the same results since the deformation is plastic and therefore path dependant. The more realistic case should be when many smaller steps are taken, since a structure doesn't go from unloaded to final displacement without the plastic deformation affecting the displacement during the loading.

The Heaviside function seem to work, achieving discrete solutions. The Heaviside projection made the program very sensitive though, in everything from tolerances to the SIMP penalization and the updating scheme of β.

Bibliography

[1] Lundgren. E Ahadi. A. *Continuum Mechanics, Avdelningen för mekanik, Lund Universitet*. 2011.

[2] K. Guest, A. Asadpoure, and Seung-Hyun Ha. Eliminating beta-continuation from heaviside projection and density filter algorithms. *Structural and multidisciplinary optimization*, 44(4):443–453, 2011.

[3] Gerhard A. Holtzapfel. *Nonlinear Solid Mechanics: A continuum approach for engineering*. Wiley, 2000.

[4] A. Kawamoto, T. Matsumori, S. Yamasaki, et al. Heaviside projection based topology optimization by a pde-filtered scalar function. *Structural and multidisciplinary optimization*, 44(1):19–24, 2011.

[5] B. S. Lazarov and O. Sigmund. Filters in topology optimization based on helmholtz-type differential equations. *International journal for numerical methods in engineering*, 86(6):765–781, 2011.

[6] P. Michaleris, D. A. Tortorelli, and C. 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.

[7] N. S. Ottosen and M. Ristinmaa. *The mechanics of constitutive modeling*. Elsevier, 2005.

[8] L. Grama R. Cazacu. Overview of structural topology optimization methods for plane and solid structures. *Annals of the Oradea university*, 3, 2014.

[9] O. Sigmund. Morphology-based black and white filters for topology optimization. *Structural and multidisciplinary optimization*, 33(4):401–424, 2007.

[10] J.C. Simo and T.J.R. Hughes. *Computational Inelasticity*, volume 7. Springer, 1998.

[11] K. Svanberg. The method of moving asymptotes- a new method for structural optimization. *International journal for numerical methods in engineering*, 24(2):359–373, 1987.

[12] N. P. van Dijk, K. Maute, and F. van Keulen. Level-set methods for structural topology optimization: a review. *Structural and multidisciplinary optimization*, 48(3):437–472, 2013.

[13] M. Wallin, V. Jönsson, and E. Wingren. Topology optimization based on finite strain plasticity. *Structural and multidisciplinary optimization*, 54(4):783–793, 2016.

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

[15] S. Xu, Y. Cai, and G. Cheng. Volume preserving nonlinear density filter based on heaviside functions. *Structural and multidisciplinary optimization*, 41(4):495–505, 2010.