

Department of Construction Sciences
Solid Mechanics

ISRN LUTFD2/TFHF-20/5240-SE(1-63)

Topology optimization with optimal spatially varying length scale

Master's Dissertation by
Stefan Lopar

Supervisors:
Mathias Wallin, Professor. Division of Solid Mechanics

Examiner:
Håkan Hallberg, Docent. Division of Solid Mechanics

Copyright © 2020 by the Division of Solid Mechanics
and Stefan Lopar

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

Acknowledgements

Most of this thesis has been carried out and written at the Division of Solid Mechanics at Lund University.

I am very grateful for my supervisor Mathias Wallin for his support and guidance throughout this project. It would never have existed without his passion for this subject.

Furthermore, I am very thankful to the entire Division of Solid mechanics for providing some much needed great company at the lunch and coffee breaks. Especially the other students writing their master's thesis, all these people made these last months a blast.

The greatest appreciations goes out to my family and close friends, it cannot be understated how their support and care have helped me.

Stefan Lopar
September 2020

Abstract

Density based topology optimization has become an efficient way to design optimal complex structures with different goals, constraints and boundary conditions. To regularize the solution, density filters are utilized and is controlled by the length scale. The length scale controls minimum feature sizes, as well as minimum curvatures in the structure. Usually a constant length scale is used throughout the design domain, but some recent articles have explored the possibility of varying the length scale spatially to control the peak stresses in a compliance minimization problem. The idea of a varying length scale and treating it as a design variable was then further explored in this thesis. The length scale was controlled by penalizing variations from the initial length scale. The idea of penalizing variations can be tied into the cost of producing the structure, where the penalty parameter for the length scale can be seen as a tradeoff between optimal structure and manufacturing cost.

The concept of a varying length scale was then tested on a L-bracket with an objective function of minimizing the compliance of the structure with a volume constraint, and later on with a stress constraint as well. The optimization is solved using the Method of Moving Asymptotes. The final structures were quite similar to what was achieved with a constant length scale, but usually with 2-4% increase in performance but at a cost at computation time.

Finally a variation in the filtering scheme is proposed to achieve consistent feature sizes without the need of padding the structure.

Keywords: Topology optimization, Varying length scale

Sammanfattning

Densitetsbaserad topologioptimering har blivit ett av det vanligaste sätten att generera komplexa optimala strukturer med olika mål, begränsningar och gränsvillkor. För att regularisera strukturerna används densitetsfilter med olika viktfunktioner som styrs av den så kallade längdskalan. Längdskalan påverkar den minsta tillåtna storleken på stänger och minsta tillåtna radien på kurvaturer. Normalt används samma längdskala genom hela designfältet, men i en relativt nypublicerad artikel undersöker der att låta längdskalan variera i rummet för att minimera spänningskoncentrationer i ett styvhetsoptimeringsproblem. Ideén att ha en rumsvarierande längdskala och behandla den som en designvariabel utforskas vidare i denna avhandling. Längdskalan var kontrollerad genom att bestraffa variationer från den initiala längdskalan. Ideén om att bestraffa variationerna kan kopplas till kostnaden för strukturen, där bestraffningsparametern för längdskalan kan ses som en avvägning mellan en optimal struktur och tillverkningskostnader.

Detta concept testades på en L balk med en målfunktion att maximera styvheten i strukturen under en volymbegränsning och sedan också en begränsning på maxspänningarna. Optimeringsproblemet är löst med Method of Moving Asymptotes. De slutgiltiga strukturerna var väldigt lika de strukturer som använde sig av en konstant längdskala. Det var oftast en förbättring på 2-4% med en varierande längdskala, men det kom med en ökning i computations tid.

Tillslut föreslås en variation till filtreringsmetoden för att uppnå konsekvent storlek på minsta tillåtna storlek för stängerna utan att placera extra element runt strukturen.

Contents

1	Introduction	6
2	Finite Element Formulation	9
2.1	Equation of motion	9
2.2	Discretization	10
3	Optimization	13
3.1	General optimization theory	13
3.2	Method of moving asymptotes	14
3.3	Solid Isotropic Material with Penalization	15
3.4	Filters	15
3.5	Heaviside projection	18
3.6	Padding	20
4	Spatially varying length scale	22
5	Implementation	27
5.1	Problem formulation	27
5.2	Sensitivity Analysis	29
5.3	Parameters	35
6	Results	36
6.1	Compliance minimization	36
6.1.1	Constant Length scale	36
6.1.2	Dynamic Length scale	37
6.2	Compliance minimization, with stress constraint	41
6.2.1	Constant Length scale	41
6.2.2	Dynamic Length scale	42
7	Discussion	44
7.1	Optimal length scale	44
7.1.1	Compliance minimization	44
7.1.2	Compliance minimization, with a stress constraint	45
7.2	Computation time and implementation	45
7.3	Locally modified length scale	47
8	Future Work	51
9	References	52

1 Introduction

The purpose of structural optimization is to find and generate the best possible structure with one or multiple goals in mind under certain constraints. This could for example mean to generate the stiffest structure given a certain amount of material, absorb the most amount of energy during an impact. Different methods with different advantages are used within structural optimization to, such as sizing, shape and topology optimization. Each of the different types of optimization schemes use different design variables to control and optimize the structure. Topology optimization is most commonly used and is the subject of interest in this work. Topology optimization works by starting with a predefined design domain that is discretized into smaller regions, each associated with a fictitious density. The densities are then allowed to vary to be either fully void or fully solid (0/1). By adding or removing material at different regions the shape of the structure will be determined and by doing this an optimal structure will emerge. Since the L-bracket will be explored in this work, an example of how the design domain is divided into smaller regions as can be seen in figure 1.

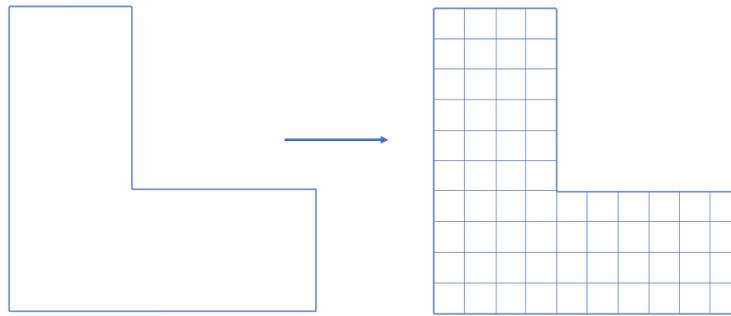


Figure 1: Discretization of the L-bracket

The L-bracket is a common shape used to evaluate stress constrained topology optimization solutions, since the sharp corner introduces many possible problems. To place material in an optimal manner, the densities have to be tied to a parameter of the region, so that the performance of different densities distributions can be evaluated. This is done by associating the densities to the Young's modulus of the region. By then adding other constraints on the shape, such as that only a certain percentage of the design domain is allowed to be filled, the optimization is forced to distribute the material efficiently.

If the optimizer were to be left alone, the generated structure might not be something that is feasible to reliably manufacture. To circumvent this, different type of procedures are used to control its shape. An examples of this is the use of filtering schemes, which are used to regularize the shape. With this procedure, different parameters are introduced which are used to control the shape. An example of these parameter is the so called length scale, which is used during filtering. The length scale

determines the minimum allowed feature size and the minimum allowed curvature in the structure. What this means is if the optimizer would like a bar that is 5 units wide, and a length scale of 8 units is used in the filtering, the optimizer would either have to increase the thickness of that bar to be at least 8 units wide or to remove and distribute the material somewhere else. The same goes for curvature of e.g. internal corners, where if the internal radius of the corner is 5 units, and a length scale of 8 units is used, the radius of the curve have to be increased. This is illustrated in figure 2 where the diameter of the circle is the length scale.

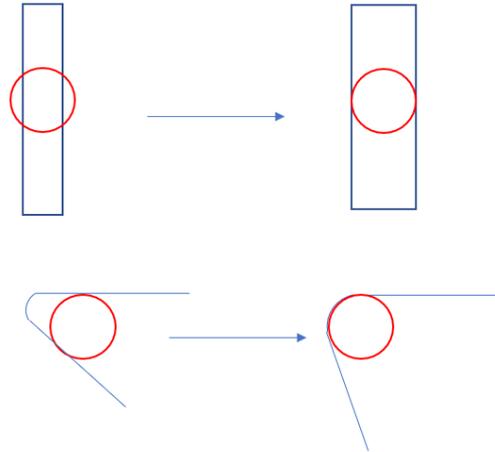


Figure 2: Illustration of how the length scale affects the minimum feature and curvature

Since the length scale controls the design complexity, the length scale can be tied into the cost of the structure, since smoother curves and fewer thicker features are generally associated with less manufacturing costs. However, since the performance of the structure is reduced with increasing length scale, choosing the length scale becomes a tradeoff between performance and cost. Something to note though is that this tradeoff is generally not constant, since the performance of each member varies in the structure. This is illustrated in figure 3 where the performance/cost is different between bar A, B and C.

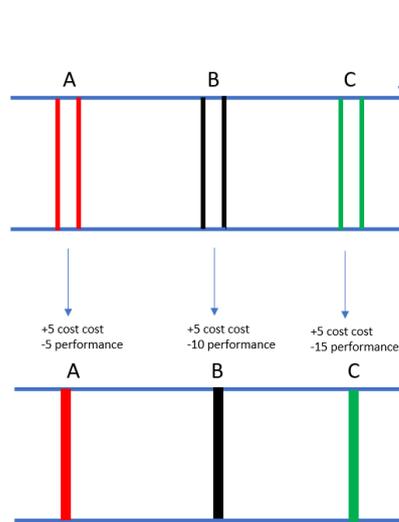


Figure 3: Illustration of how the performance/cost may vary between the bars when increasing the length scale

The performance/cost for each member is usually not considered since the entire structure is filtered with the same length scale, but if it was allowed to vary within the structure, a more accurate comparison could be made. To achieve this, each region is assigned its own length scale to allow the minimum feature and curvature to vary, and is treated as a design variable.

To evaluate the behaviour of the structure, the finite element method is used. The necessary prerequisites to implement the discretization will be described. After that, general optimization theory will be explained, alongside the solution scheme used. Finally the results will be presented and discussed.

2 Finite Element Formulation

To achieve a numerical representation of the structure, a finite element formulation is used. The finite element formulation is based on the weak form of the equations of linear motion. However, the equation is expressed continuously and to get a numerical representation it has to be discretized, essentially what was described in the introduction as dividing into regions. The goal is then to first describe the equation of motion in a suitable manner that can be discretized and then use the discretized field to achieve an approximation of the continuous field.

2.1 Equation of motion

The formulation follows the derivation described by Ottosen and Risinmaa[1]. Using *Newton's Second Law of Motion* on an arbitrary body, see figure 4, with a volume V and surface S the equation can be written as

$$\int_V \rho \ddot{a}_i dV = \int_S t_i dS + \int_V b_i dV \quad (2.1)$$

where ρ is the mass densities, \ddot{a}_i the acceleration vector, a_i the displacement vector, t_i traction vector and b_i the body force vector.

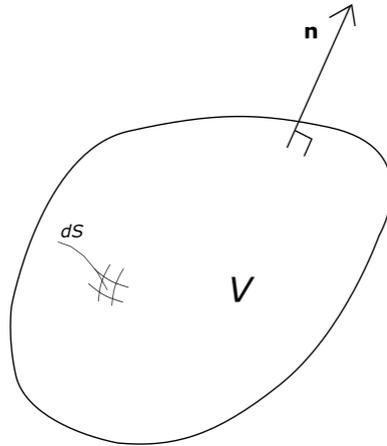


Figure 4: An arbitrary body with a volume V and surface area S

Making use of the divergence theorem and that the traction can be rewritten as

$$t_i = \sigma_{ij} n_j \quad (2.2)$$

where σ_{ij} is the stress and n_j is the normal to the surface, equation 2.1 can be expressed as

$$\int_V (\sigma_{ij,j} + b_i - \rho \ddot{a}_i) dV = 0 \quad (2.3)$$

Since this relation is valid for an arbitrary volume it is concluded that

$$\sigma_{ij,j} + b_i = \rho \ddot{a}_i \quad (2.4)$$

From this, equation 2.4 is multiplied by an arbitrary vector ν_i and integrated over the volume.

$$\int_V [(\sigma_{ij}\nu_i)_{,j} - \sigma_{ij}\nu_{i,j}]dV + \int_V (\nu_i b_i - \rho \nu_i \ddot{a}_i)dV = 0 \quad (2.5)$$

Again from the divergence theorem and equation 2.2 it holds that

$$\int_V (\sigma_{ij}\nu_i)_{,j}dV = \int_S \sigma_{ij}\nu_i n_j dS = \int_S \nu_i t_i dS \quad (2.6)$$

Inserting equation 2.6 into equation 2.5 gives

$$\int_V \rho \nu_i \ddot{a}_i dV + \int_V \nu_{i,j} \sigma_{ij} dV = \int_S \nu_i t_i dS + \int_V \nu_i b_i dV \quad (2.7)$$

By introducing the relation

$$\epsilon_{ij}^\nu = \frac{1}{2}(\nu_{i,j} + \nu_{j,i}) \quad (2.8)$$

and the symmetry of σ_{ij} yields

$$\nu_{i,j} \sigma_{ij} = \epsilon_{ij}^\nu \sigma_{ij} \quad (2.9)$$

Inserting this in equation 2.7 and rewriting it in Voigt notation gives the principal of virtual work or the weak form of the balance equation

$$\int_V \rho \boldsymbol{\nu}^T \ddot{\mathbf{a}} dV + \int_V \boldsymbol{\epsilon}^{\nu T} \boldsymbol{\sigma} dV = \int_S \boldsymbol{\nu}^T \mathbf{t} dS + \int_V \boldsymbol{\nu}^T \mathbf{b} dV \quad (2.10)$$

2.2 Discretization

To solve the balance equation it is discretized, meaning that the continuous domain is divided into finite amount of elements where the nodal displacements of the elements can be calculated. By introducing the global shape function $\mathbf{N} = \mathbf{N}(\mathbf{X})$, where \mathbf{X} are the undeformed coordinates of the element, intermediate values can be interpolated from the nodal values. The relation between the nodal displacements u and the displacement vector \mathbf{a} is then described as $\mathbf{a} = \mathbf{N}u$

By also introducing the partial derivative of the shape function it follows that

$$\begin{aligned} \boldsymbol{\nu} &= \mathbf{N} \mathbf{c} \\ \ddot{\mathbf{a}} &= \mathbf{N} \ddot{\mathbf{u}} \\ \mathbf{B} &= \tilde{\nabla}^s \mathbf{N} \\ \tilde{\nabla}^s &= \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \\ \boldsymbol{\epsilon} &= \mathbf{B} u \end{aligned} \quad (2.11)$$

where \mathbf{c} is an arbitrary vector. By assuming static conditions, the acceleration $\ddot{\mathbf{u}}$ will be equal to zero. By also introducing the constitutive matrix \mathbf{D} , a relation between the stress tensor $\boldsymbol{\sigma}$ and the strain tensor $\boldsymbol{\epsilon}$ can be made for linear elastic materials as $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon}$

For isotropic materials with plane stress conditions, the constitutive matrix \mathbf{D} is described as

$$\mathbf{D} = \frac{E}{1 - \nu^2} = \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & 1 - \nu \end{bmatrix} \quad (2.12)$$

Where E is the elasticity modulus and ν is Poisson's ratio of the material. With \mathbf{c} also being arbitrary, inserting equation 2.11 and equation 2.12 into equation 2.10 will simplify it to

$$\left(\int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \right) \mathbf{u} = \int_S \mathbf{N}^T \mathbf{t} dS + \int_V \mathbf{N}^T \mathbf{b} dV \quad (2.13)$$

The stiffness matrix \mathbf{K} and the force vector \mathbf{F} are introduced as

$$\begin{aligned} \mathbf{K} &= \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \\ \mathbf{F} &= \int_S \mathbf{N}^T \mathbf{t} dS + \int_V \mathbf{N}^T \mathbf{b} dV \end{aligned} \quad (2.14)$$

Simplifying equation 2.13 to

$$\mathbf{K} \mathbf{u} = \mathbf{F} \quad (2.15)$$

Depending on what type of element is used the shape function will look different. A 4 node quadratic element was used in this thesis and is illustrated in figure 5.

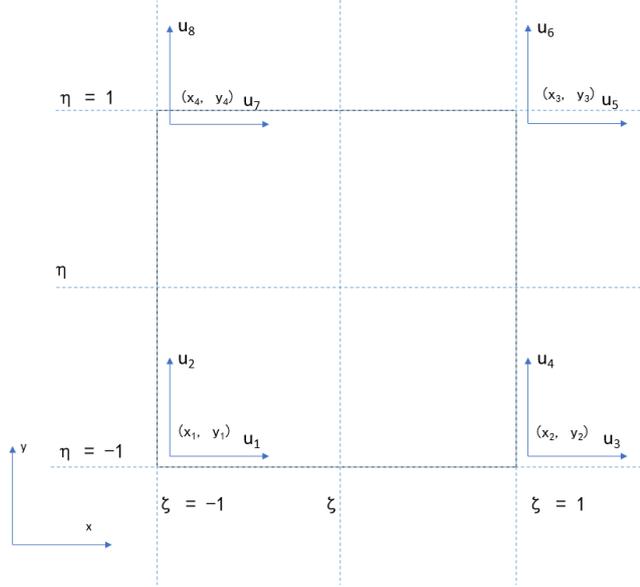


Figure 5: 4 node quadratic element with nodal displacements at the degrees of freedom

The 4 node element shape function is defined as

$$\xi = \xi(x) \quad \eta = \eta(y) \quad a = \frac{1}{2}(x_3 - x_1) \quad b = \frac{1}{2}(y_3 - y_1)$$

$$N_1 = \frac{1}{4ab}(\xi - x_2)(\eta - y_4) \quad N_2 = \frac{1}{4ab}(\xi - x_1)(\eta - y_3)$$

$$N_3 = \frac{1}{4ab}(\xi - x_4)(\eta - y_2) \quad N_4 = \frac{1}{4ab}(\xi - x_3)(\eta - y_1)$$

$$\mathbf{N}^e = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix}$$

The property of the shape function guarantees that the values on the nodes are preserved since, e.g. N_1 is equal to 1 if $\xi = -1$, $\eta = -1$ are inserted, while N_2 , N_3 and N_4 will be 0.

3 Optimization

Once we have the numerical representation of the structure, we can optimize the structure using gradient based methods. Based on the gradient, the design is updated iteratively. In this chapter we define the problems, how the iterative approximation is used and what tools we use to control the behaviour of the structure.

3.1 General optimization theory

A general structural optimization problem can be defined as

$$\mathbb{SO} \begin{cases} \text{minimize } g_0(\mathbf{x}, \mathbf{y}) \text{ with respect to } \mathbf{x} \text{ and } \mathbf{y} \\ \text{subjected to } \begin{cases} \text{behavioral constraint on } \mathbf{y} \\ \text{design constraint on } \mathbf{x} \\ \text{equilibrium constraint} \end{cases} \end{cases} \quad (3.1)$$

where \mathbf{x} are the design variables and \mathbf{y} are the state variables representing the response of the structure. For a mechanical structure the response could mean e.g. displacement or stress. The formulation 3.1 is called a *simultaneous formulation*, since the optimization problem and equilibrium are solved simultaneously. It is however common that the state problem is uniquely defined by the design variables. An example can be made for state equation 2.15 where if $\mathbf{K}(\mathbf{x})$ is invertible for all \mathbf{x} , the displacements $\mathbf{u} = \mathbf{u}(\mathbf{x})$ can be treated as a given function and the equilibrium constraint can be left out of \mathbb{SO} . This leads to the so called nested formulation

$$\mathbb{SO}_{nf} \begin{cases} \text{minimize } g_0(\mathbf{x}, \mathbf{u}(\mathbf{x})) \\ \text{subjected to } g_i(\mathbf{x}, \mathbf{u}(\mathbf{x})) < 0, \quad i = 1, 2, \dots, N_{constraints} \end{cases} \quad (3.2)$$

where g_0 is the objective function and g_i are the constraint functions. To find an optimal solution, a Lagrangian formulation is used, which is a function defined by the target and constraint functions. If a structural optimization problem \mathbb{P} is convex, and Slater's constraint qualification is satisfied, then it can be expressed as a Lagrangian Duality and be equivalent to each other[2]. The dual problem is then defined as

$$\mathbb{D} \begin{cases} \max \varphi(\boldsymbol{\lambda}) \\ \text{s.t. } \boldsymbol{\lambda} \geq 0 \end{cases} \quad (3.3)$$

where the dual objective function φ is

$$\varphi = \min \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) \quad (3.4)$$

and the Lagrangian \mathcal{L} of \mathbb{P} is defined as

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = g_0(\mathbf{x}) + \sum_{i=1}^l \lambda_i g_i(\mathbf{x}) \quad (3.5)$$

where l are the number of constraints and λ_i is the Lagrangian multiplier for the i -th constraint. An optimal solution to an optimization problem, should satisfy the so called *KKT-conditions*. The set of points that satisfies the *KKT-conditions* are called *KKT-points*. For a point x^* to be qualified as a *KKT-point* the following conditions must be met:

$$\begin{aligned}
\frac{\partial \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_j} &\leq 0 \quad \text{if } x_j = x_j^{max} \\
\frac{\partial \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_j} &= 0 \quad \text{if } x_j^{min} < x_j < x_j^{max} \\
\frac{\partial \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_j} &\geq 0 \quad \text{if } x_j = x_j^{min} \\
\lambda_i g_i(\mathbf{x}) &= 0 \quad g_i(\mathbf{x}) \leq 0 \quad \lambda_i \geq 0
\end{aligned} \tag{3.6}$$

for all $j = 1, 2, \dots, n$ and $i = 1, 2, \dots, l$ where n are the number of design variables.

3.2 Method of moving asymptotes

In the previous section it was mentioned that if a problem was convex it could be expressed as a Lagrangian duality and they would be equivalent to each other. However, many of the problems that are solved in structural optimization are non-convex. This leads to that the *KKT-points* that are found, are not guaranteed to be a local optima. If the problem is non-convex there may exist *KKT-points* which are not optimal solutions. To circumvent this, convex approximations are made of the optimization problem. One common method is the *Method of Moving Asymptotes* (MMA), derived by Svanberg[3]. MMA linearizes the problem by replacing the design variables with the intervening variables

$$y_j(x_j) = \begin{cases} \frac{1}{x_j - L_j} & \text{if } \frac{\partial g_i(\mathbf{x}^k)}{\partial x_j} < 0 \\ \frac{1}{U_j - x_j} & \text{if } \frac{\partial g_i(\mathbf{x}^k)}{\partial x_j} > 0 \end{cases} \tag{3.7}$$

The intervening variables are in turn controlled by the moving asymptotes L_j and U_j during the optimization loop to control the speed of convergence as necessary. The MMA approximation is then obtained by expressing the functions with the intermediate terms and taking a Taylor series expansion and disregarding the higher terms to get

$$\begin{aligned}
g_i^{M,k} &\approx g_i(\mathbf{x})^k + \sum_{j=1}^n \frac{\partial g_i^k}{\partial x_j} \left(\frac{(U_j^k - x_j^k)^2}{U_j^k - x_j^k} - (U_j^k - x_j^k) \right) - \\
&\quad \sum_{j=1}^n \frac{\partial g_i^k}{\partial x_j} \left(\frac{(x_j^k - L_j^k)^2}{x_j^k - L_j^k} - (x_j^k - L_j^k) \right)
\end{aligned} \tag{3.8}$$

where k denotes the current iteration step. This is done for both the target and constraint function to iteratively solve the approximated version of the dual problem. The

derivatives used in solving structural optimization problems are called sensitivities. The asymptotes \mathbf{U} and \mathbf{L} are dependent on the values of the current design values \mathbf{x}^k and the two previous design values \mathbf{x}^{k-1} and \mathbf{x}^{k-2} . If the signs from $x_j^k - x_j^{k-1}$ are different from $x_j^{k-1} - x_j^{k-2}$ then the asymptote values are too far away from the current design and should be brought closer. If the signs are the same the asymptotes are moved further away to increase the speed of convergence.

3.3 Solid Isotropic Material with Penalization

With the Solid Isotropic Material with Penalization method(SIMP) the Young's modulus of the element of linear elastic materials and the densities are directly tied together as

$$E_e = \rho_e^q E \quad (3.9)$$

where E_e is the effective Young's modulus of the element and q is a penalty factor and E is the Young's modulus of the material. By choosing $q > 1$ intermediate values will be penalized in the optimization, as shown in figure 6. However, depending on the optimization problem, different penalization values are used. A common practice is to set q_k to 3, where k denotes the penalization used the stiffness calculation, and q_s to 0.5, where s denotes the penalization used used in the stress calculation, which is what will be used throughout this thesis.

Values where $\rho_e \approx 0$ numerical problems may occur with the penalization and therefore the effective Young's modulus is rewritten as

$$E_e = (\rho_e^q(1 - \delta) + \delta)E \quad (3.10)$$

where δ is a smaller number in the order of 10^{-7} . This guarantees E_e to at least be equal to δE .

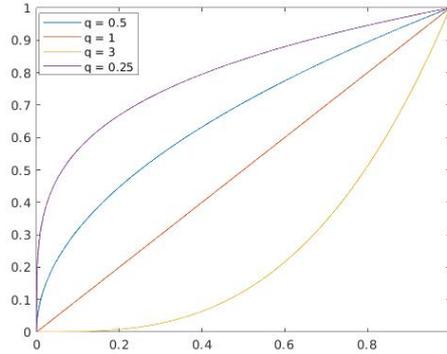


Figure 6: The effective density as a function of ρ and different penalization factors.

3.4 Filters

A common problem that is associated with the SIMP method is the occurrence of a checkerboard pattern in the design. The parameter q penalizes the intermediate

values and pushes the design to either solid or void, but at an alternating fashion, as illustrated in figure 7.

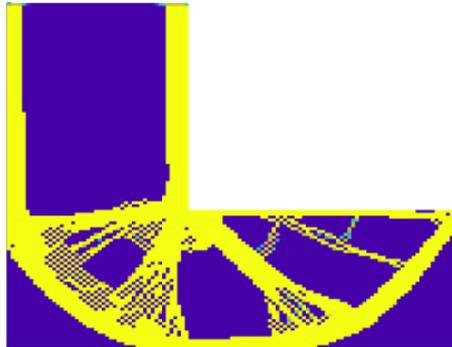


Figure 7: A checkerboard pattern that appears in optimization problems.

Fully solid regions are yellow and fully void regions are blue, which also applies for all figures in this work with a density distribution. This type of structure is not desired, since it becomes essentially impossible to manufacture and is mesh dependent. A commonly employed remedy is the introduction of filters. The filters gives elements an influence on surrounding elements that are within a certain length of each other, called the length scale. The values will become a weighted average of the elements within the length scale, where elements that are closer will have a higher effect on the weight. This is done to get a better representation if a element is surrounded by solid or void elements, and is achieved with the weight functions. An illustration of this effect can be seen in figure 8.

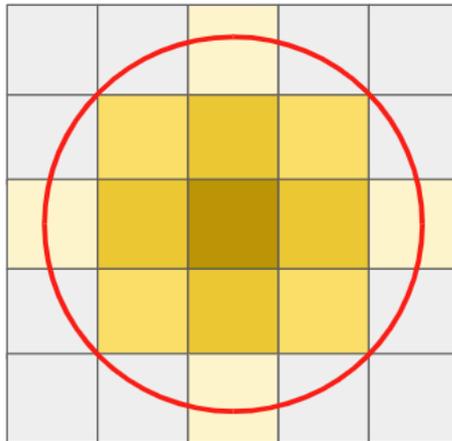


Figure 8: Illustration of how a weight function might work, with elements further away have a less impact which is illustrated with a lighter colour. The radius of the circle is the length scale.

The effect of one such filter has on the checkerboard is illustrated in figure 9.

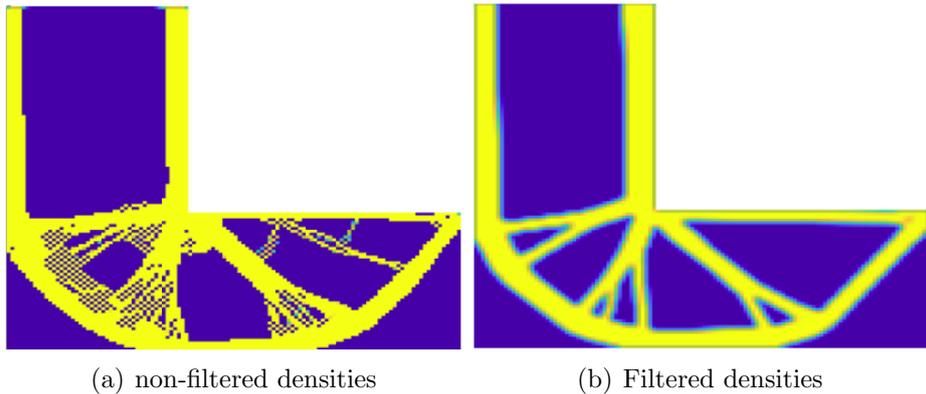


Figure 9: The effect the filter equation 3.15 has on the densities

A common filter used in topology optimizations is the density filter proposed by Bruns and Tortorelli [4], where the filtered densities are calculated as

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e} w(x_i) v_i \rho_i}{\sum_{i \in N_e} w(x_i) v_i} \quad (3.11)$$

where N_e is the set of elements within the length scale r of e and $w(x_i)$ is the weight function and v_i is the volume of element i . One of the most common weighting function is the linear decaying function, also called cone-shape function, defined as

$$w(x_i) = r - |x_i - x_e| \quad (3.12)$$

where $|x_i - x_e|$ is the distance between the centers of elements i and e . The function 3.12 is easy to implement, but a drawback is that it is not continuous with respect to r since elements will be added or removed from the summation in equation 3.11 with increasing/decreasing r . One solution to this is to use a smooth weighting function such as the Gaussian distribution function

$$w(x_i) = e^{-\frac{1}{2} \left(\frac{|x_i - x_e|}{R} \right)^2} \quad (3.13)$$

where N_e instead includes all elements in the design. However, to have the same influential length scale as the cone-shaped function, the radius R is modified to

$$R = \frac{r}{3} \quad (3.14)$$

This helps the bell-shaped function to truncate close to the same length scale as in the cone-shaped weight function.

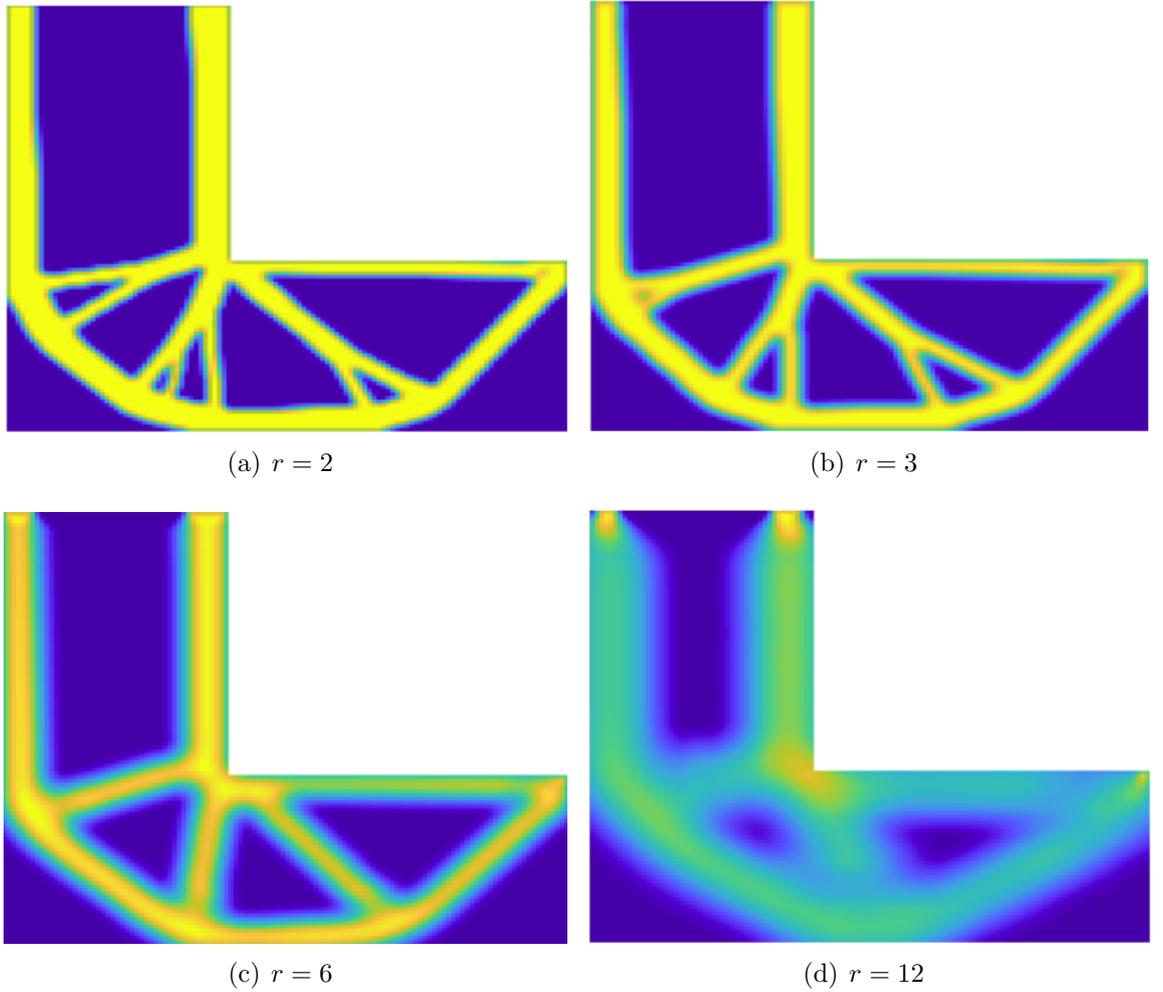


Figure 10: Figures showing the impact the length scale has on the structure

3.5 Heaviside projection

A drawback of using filters to regularize the structure is the introduction of gray zones, design variables with intermediate values that can make the structure undefined with increasing values of the length scale, as can be seen in the figures in 10. This is usually solved by using a Heaviside projection on the filtered densities to push the elements to either be solid or void. Since the function is used in the sensitivity analysis it is preferable if the function is continuously differentiable, therefore an approximation of the step is used such as

$$\bar{\rho} = \frac{\tanh(B_H z) + \tanh(B_H(\tilde{\rho} - z))}{\tanh(B_H z) + \tanh(B_H(1 - z))} \quad (3.15)$$

where z is the cutoff point of the truncation and B_H is a constant used to control the slope of the truncation[8]. For small values of B_H the function is almost linear and is

used early in the optimization procedure to ensure stable convergence and is increased throughout to approximate a step.

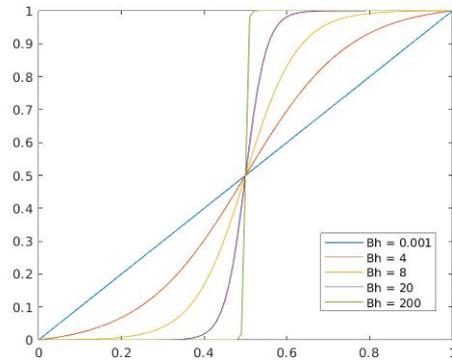


Figure 11: Plots of equation 3.15 with different B_H values and z equal to 0.5

Different projection schemes have been proposed (see [13] and [7]) with different goals in mind, but equation 3.15 is the only one used in this thesis.

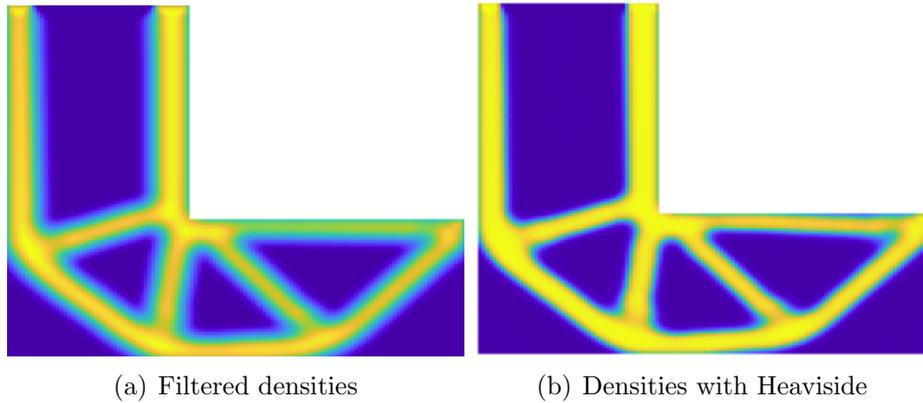


Figure 12: The effect the Heaviside function 3.15 on the densities

3.6 Padding

The length scale becomes a parameter that determines the minimum feature size of both the solid and void elements. The effect is shown in figure 13.

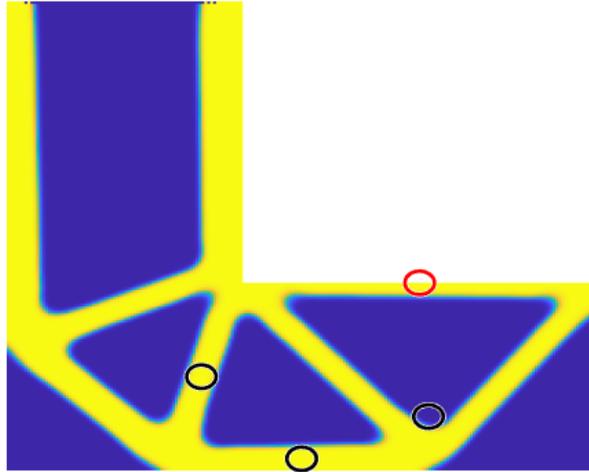


Figure 13: A problem that is filtered with a length scale of 10 element widths, the size of the circles. Both the thickness of the bars and the curvature of the void is affected by the length scale. Since no padding is added, the effective length scale on the edges is halved as can be seen by the red circle

A problem that may arise when using this type of filter is the non-consistency that arises at the boundaries. The elements at, or close to, the boundaries will have fewer elements to compare to when calculating the “average” if anything outside the design domain is not part of the evaluation. This allows smaller features to exist at the edge since effectively the length scale is halved, which can also be seen by the red circle in figure 13. A common solution is to extend the domain the weights are calculated with the introduction of padded void elements along the edges of the design domain. With the padded void elements the minimum feature size is applied over the entire structure as can be seen in figure 14. However with the introduction of void elements, elements close to the boundary may lose their connection with the edge, see figure 14 again, potentially losing contact with loads which may become a problem for the finite element analysis as discussed in [4]. This problem is illustrated in figures 14 and 15, and shows the need of local modification in these areas with padded solid elements to ensure contact.

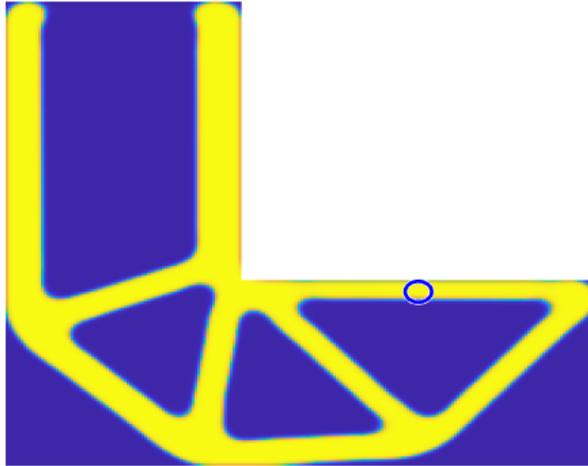


Figure 14: The same minimization problem as in figure 13, but with padded elements around the design region, which allows the length scale to influence the bar elements equally.

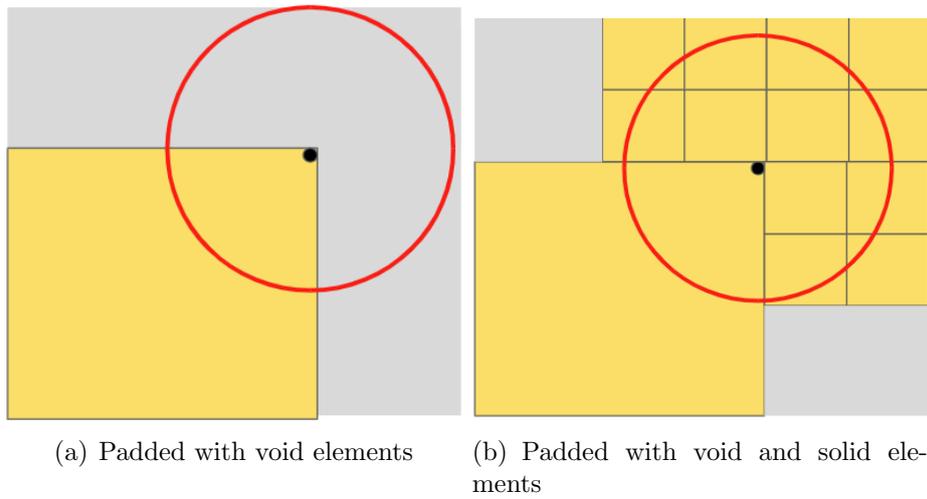


Figure 15: A close up of the rightmost corner of the L-bracket, showing the padding with solid elements. The gray indicates void, and yellow indicates solid

4 Spatially varying length scale

Choosing the correct length scale r is usually done by testing different values until a suitable tradeoff has been made between optimal and suitable design. Since large values for r may make the structure easier to produce with larger feature sizes and smoother transitions, but in general a worse design in terms of mechanical response. This is because smaller features may be advantageous for the design but will be filtered out during the regularization scheme.

In [9] both small and large length scales could exist simultaneously. The strategy was applied to compliance minimization of an L-bracket.

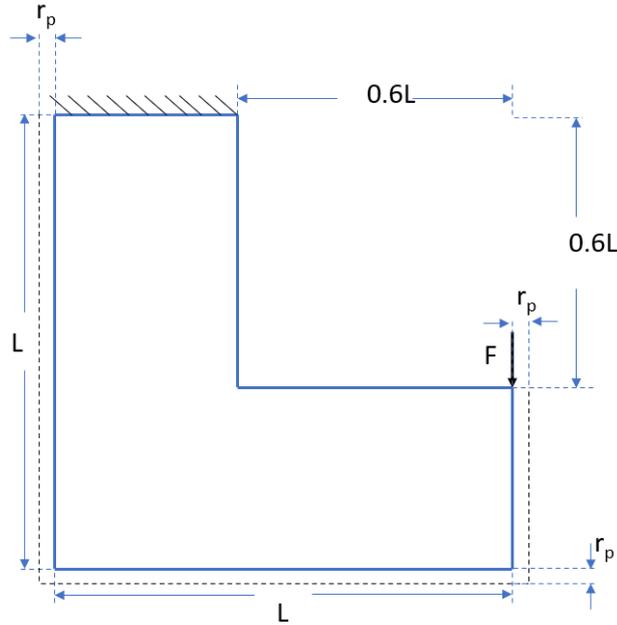


Figure 16: Design domain of a L-bracket, with fixed support at the top

In the considered example, the domain is padded with void elements at the edge, where r_p is the size of the length scale and the force \mathbf{F} is distributed to 9 neighbouring nodes to minimize stress concentrations as is illustrated in figure 17. The corner is also padded with solid elements where the load is applied to ensure the load is connected to material after the regularization scheme and Heaviside projection. The top of the L-bracket is fixed, which can be seen in figure 16. The upper parts of the L-bracket is also padded with solid elements to not lose contact with the top edge.

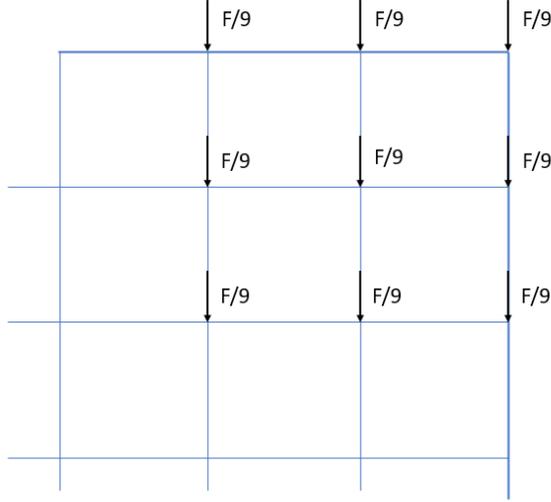


Figure 17: Close up of the corner of the L-bracket where the load is applied. The load is distributed among 9 neighboring nodes

In [9] it is shown that the compliance increases when the length scale increased. The peak stresses in the structure was also dependent on the length scale where they decreased with increasing values of r up to a certain point. This is because having smaller length scale allows the structure to have thinner features, which is good for the stiffness, but at the same time sharper corners where peak stresses may occur. Increasing the length scale smoothens the sharp corners, decreasing the peak stresses up to a certain point where deformation in structures becomes too large due to the structure being weaker. In [9] this shift happened when r was larger than 13 element widths. This shift is however dependent on how coarse or fine the mesh is.

With this in mind, [9] then introduced another function

$$\phi_{outer} = |\sigma_{max}^{VM}(\bar{\rho}(r)) - \sigma_{max}^*| \quad (4.1)$$

where σ_{max}^{VM} is the max von mises stress in the structure and σ_{max}^* is the max allowed stress. The von mises stress is an equivalent stress used to approximate the total stress each element is experiencing and is defined as

$$\sigma_e^{VM} = \sqrt{\sigma_{e,xx}^2 + \sigma_{e,yy}^2 - \sigma_{e,xx}\sigma_{e,yy} + 3\tau_{e,xy}^2} \quad (4.2)$$

for plane stress conditions and $\sigma_{e,xx}$, $\sigma_{e,yy}$ and $\tau_{e,xy}$ are components of the Cauchy stress tensor. The function was not considered in the main optimization loop, and the goal was to minimize the difference in max stress and allowed stress with the length scale as a variable, since the length scale had an impact on the max stresses, as was shown in [9]. To get the sensitivity of equation 4.1 with respect to r a numerical difference was used to get the derivative according to

$$\frac{d\sigma_{max}^{VM}}{dr} \approx \frac{\sigma_{max}^{VM}(\bar{\rho}(r_1 + \Delta r)) - \sigma_{max}^{VM}(\bar{\rho}(r_i))}{\Delta r} \quad (4.3)$$

where r_i is the current length scale and was updated as

$$r_{i+1} = r_i + \operatorname{sgn} \left(\frac{\sigma_{max}^* - \sigma_{max}^{VM}(\bar{\rho}(r_i))}{\frac{d\sigma_{max}^{VM}}{dr}} \right) \left| \frac{\sigma_{max}^* - \sigma_{max}^{VM}(\bar{\rho}(r_i))^{0.5}}{\frac{d\sigma_{max}^{VM}}{dr}} \right| \quad (4.4)$$

The outer loop was performed every 10^{th} iteration and [9] called this procedure “An inner-outer optimization procedure”

Alongside this, another procedure called “A procedure with spatially varying filter radius” was presented in [9]. From their first simulations it was also noted that the lowest compliance was achieved with larger number of thinner bars and lower stresses was achieved with smoother transitions at re-entrant corners. Instead of using the same filter radius throughout the structure, a length scale depending on its position in space was introduced. This was done to be able to have both smooth transitions and thinner bars at the same time.

The points where peak stresses occurred, such as at re-entrant corners, was seen as a “stress-attractor points” and became a starting point for updating the length scale field. The function ψ was introduced as

$$\psi(x, y) = \exp \left(- \left| \frac{d(x, y)}{D} \right|^\theta \right) \quad (4.5)$$

where $d(x, y)$ is the distance from the stress attractor point, D is the characteristic influenced distance and θ determines the sharpness of the function. With equation 4.5 the new filters can be computed with the distance from the stress attractor point in mind as

$$\hat{r}(x, y) = (1 + \zeta\psi(x, y))r_{min} \quad (4.6)$$

where r_{min} is the initial length scale and ζ is a parameter that determines the increase in filter radius. The effect these different parameters have on \hat{r} are shown in figure 18. With this, different parts of the structure will be filtered differently and allow for differently sized minimum features.

This new radii are updated throughout the optimization process and follows the structure:

1. Determine all points that violate the max allowed stress σ_{max}^* .
2. Sort the stress points according to their magnitude.
3. Remove duplicates i.e. points close to each other that belong to the same stress concentration. They are considered the same stress concentration if they are within D of the point with the highest stresses.
4. Generate a set of ψ functions related to each stress point and update $\hat{r}(x, y)$, if any sets are overlapping, the highest value is chosen.

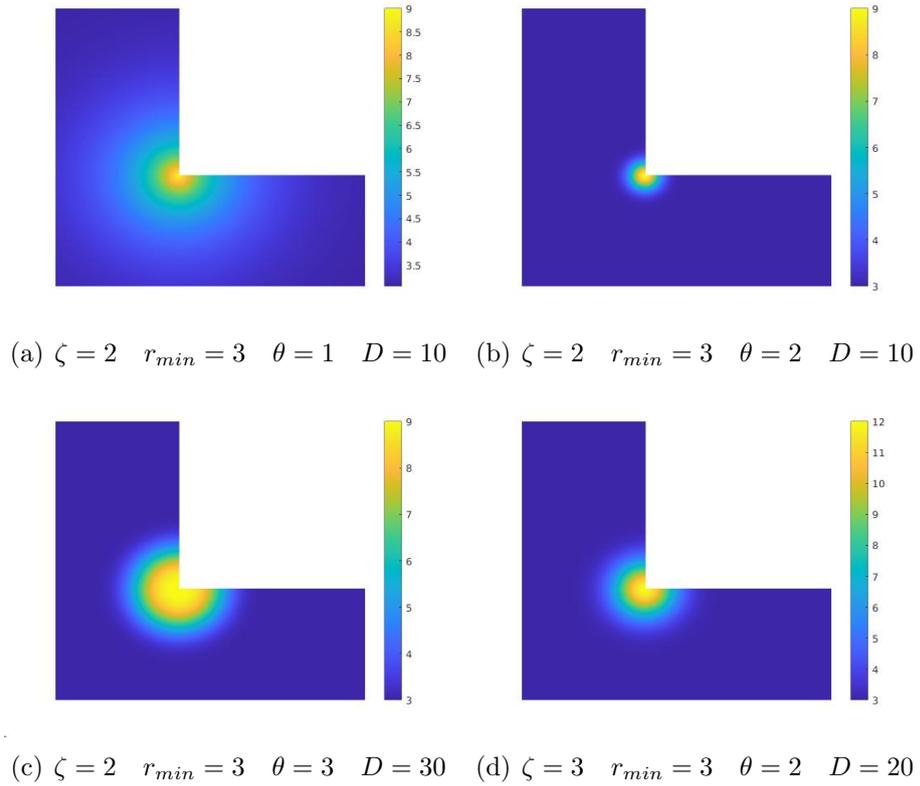


Figure 18: The impact the parameters ζ , r_{min} , θ and D have on the spatial radius \hat{r}

Finally both methods were added together where the “*An inner-outer optimization procedure*” was used to update r_{min} according to equation 4.4 and then using the stress points the spatial radii was calculated using equation 4.5 and 4.6.

A few of the parameters used in [9] were tested to show the influence a varying length scale can have on the final structure and the result can be seen in figure 19. The same trends noted in their article were also observed, mainly that more material was placed at the inner bend of the L-bracket with the bigger local radius. However, the results differed a lot from the the results in the paper. This is because the simulations done in this thesis did not include the elements outside the design domain in the FEM analysis, only during the filtration. This had quite a big impact on the peak stresses since the larger radii did not allow the corner to smoothen properly outside the design domain, as are shown in figure 20. It should also be noted that the simulations in [9] used the robust approach [11] to generate a dilated and eroded design during the optimization procedure, which was not utilized in the figures shown in figure 19.

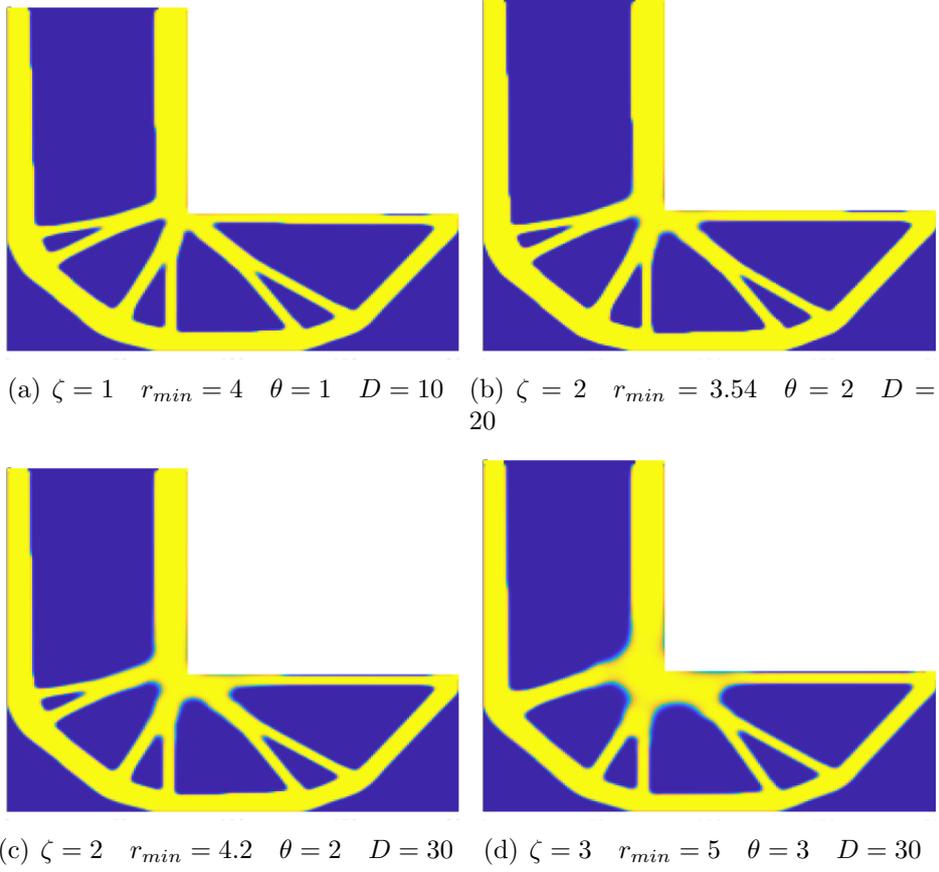


Figure 19: Different structures generated using the “An inner-outer optimization procedure”

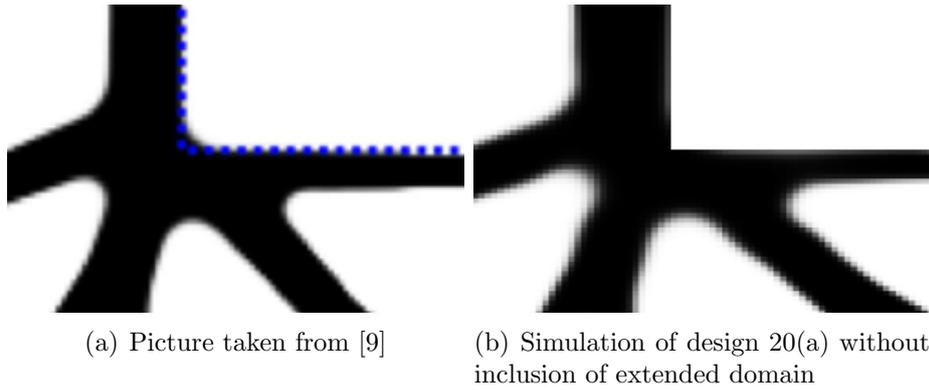


Figure 20: A highlight between the difference in the corner of the L-bracket.

5 Implementation

In the previous section, the influence of the length scale had on the stiffness stiffness was discussed. It was noted that the length scale had an impact on the peak stresses since it had the ability to control the curvature of the design. In conclusion, smoother curvature can reduce the peak stresses and the ease of manufacturing. The effect of 2 different length scales is shown in figure 21.

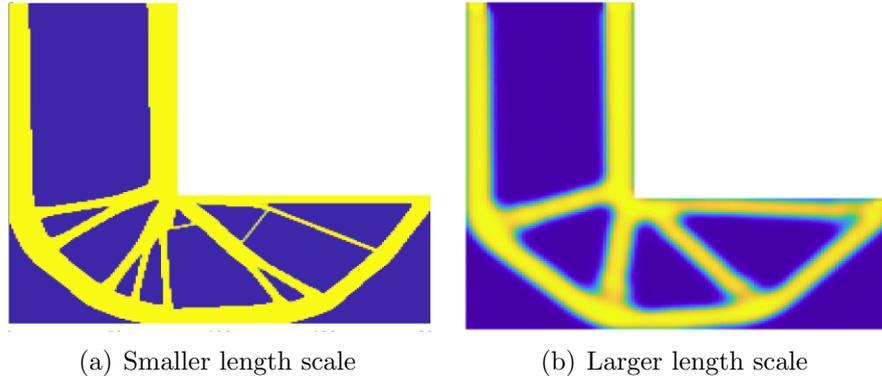


Figure 21: Structures used to compare the impact of the length scale

Figure 21(a) is stiffer from the simulations but depending on the situation figure 21(b) might be stiff enough. If the goal was then to manufacture the structures in figure 21 using for a example CNC mill, the cost of producing the structure with smaller length scale would be higher. This is because thinner end mills would have to be used to achieve the tighter curvature that is produced by the smaller length scale, increasing the runtime of the machine since smaller cuts have to be made each pass or add the need to change the tooling used throughout the cutting process. This is usually where the user has to find a balance of appropriate length scale to achieve a manufacturable structure but also optimal at its task. However, similarly to what was done in [9] it might be advantageous to have spatially varying length scales in the structure to allow tighter curvature if a big advantage can be made in the performance of the structure, while increasing its cost. How to modify the length scale to achieve this balance is not trivial, and is therefore left to the optimizer.

5.1 Problem formulation

The optimization problem that has been studied is defined as a compliance minimization problem of an L-bracket with a constraint on the volume. The same L-bracket described in section 4 was used. The difference from the standard compliance minimization problem is that both the densities and the length scales are treated as a design variable. Each element is assigned a length scale and a density value, similar to what was done in [9], which allows the length scale to vary spatially and also be

controlled by the optimizer. The problem \mathbb{P}_1 is then defined as

$$\mathbb{P}_1 \left\{ \begin{array}{l} \min g_0(\bar{\mathbf{x}}, \mathbf{u}(\bar{\mathbf{x}})) = \mathbf{F}^T \mathbf{u} \\ \text{s. t.} \left\{ \begin{array}{l} g_1 = \sum \bar{\rho}_i v_e - V_{max} \leq 0 \\ \bar{\mathbf{x}} = [\bar{\boldsymbol{\rho}}, \bar{\mathbf{r}}] \\ 0 \leq \bar{\rho}_i \leq 1, \quad r_{min,i} \leq \bar{r}_i \leq r_{max,i}, \\ i = 1, 2, \dots, Nel m \end{array} \right. \end{array} \right. \quad (5.1)$$

where the objective function g_0 is the standard compliance minimization, g_1 the volume constraint with a max volume of V_{max} , $Nelm$ is the number of elements, the Heaviside projected densities $\bar{\rho}$ and the maximum and minimum allowed length scale r_{min} and r_{max} . V_{max} is set to 35% of the design domain. The densities can vary from 0 to 1 where 1 is solid material and 0 being void. To control the length scale, g_0 is modified to

$$g_0(\bar{\mathbf{x}}, \mathbf{u}(\bar{\mathbf{x}})) = \mathbf{F}^T \mathbf{u} + \gamma \sum_{i=1}^{Nelm} (r_c - \bar{r}_i)^2 \quad (5.2)$$

where r_c is the initial length scale of the structure, \bar{r}_i is the length scale of each element. The parameter γ is introduced to penalize the compliance if the length scale deviates from the initial length scale. By discouraging variations in length scales, unless a clear advantage in compliance could be made, γ can be seen as a balancing parameter between manufacturing costs and compliance.

It should be mentioned that to make γ mesh independent it should be scaled by the number of elements, but since the number of elements stays the same throughout the computation in this work it is not necessary.

This approach was also tested on problem \mathbb{P}_2 where a constraint on the maximum allowed peak stress σ_{max} was introduced.

$$\mathbb{P}_2 \left\{ \begin{array}{l} g_0(\bar{\mathbf{x}}, \mathbf{u}(\bar{\mathbf{x}})) = \mathbf{F}^T \mathbf{u} + \gamma \sum_{i=1}^{Nelm} (r_c - \bar{r}_i)^2 \\ \text{s. t.} \left\{ \begin{array}{l} g_1 = \sum \bar{\rho}_i v_e - V_{max} \leq 0 \\ g_2 = \sigma^{PN} - \sigma_{max} \leq 0 \\ \bar{\mathbf{x}} = [\bar{\boldsymbol{\rho}}, \bar{\mathbf{r}}] \\ 0 \leq \bar{\rho}_i \leq 1, \quad r_{min} \leq \bar{r}_i \leq r_{max}, \\ i = 1, 2, \dots, Nel m \end{array} \right. \end{array} \right. \quad (5.3)$$

To evaluate the peak stress in the structure a P-norm was used, defined as

$$\sigma^{PN} = \left(\sum_{i=1}^{Nelm} (\sigma_i^{VM})^p \right)^{\frac{1}{p}} \quad (5.4)$$

where σ_i^{VM} is the von Mises stress of each element. The exponent p is an ‘‘exaggeration’’ constant that heavily increases the effect stress concentrations have on σ^{PN} for large

values of p . In the limit case of when p goes to infinity equation 5.4 is equal to the max von Mises stress in the structure. Minimizing σ^{PN} will then become equivalent of minimizing the max stresses in the structure. For extremely large values of p numerical problems occur, which leads to a practical limit for how big p can be chosen. $p = 14$ is used throughout this thesis, but lower values of p have been used with successful results[10].

5.2 Sensitivity Analysis

Since the sensitivities are necessary to evaluate the response of the structure they have to be calculated. Calculating the sensitivities of the compliance can be quite costly to compute because of the implicit derivatives that appear. A common solution to this is the introduction of an adjoint vector. With the adjoint vector, g_0 is modified to be

$$\bar{g}_0 = \mathbf{F}^T \mathbf{u} + \gamma \sum (r_c - r_i)^2 - \boldsymbol{\lambda}^T (\mathbf{K} \mathbf{u} - \mathbf{F}) \quad (5.5)$$

where $\boldsymbol{\lambda}$ is the adjoint vector. This does not change the original function since the parenthesis $\mathbf{K} \mathbf{u} - \mathbf{F}$ is 0. A total derivative of \bar{g}_0 is calculated as

$$\frac{D\bar{g}_0(\mathbf{u}(\bar{\mathbf{x}}), \mathbf{x})}{D\mathbf{x}} = \frac{\partial \bar{g}_0}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \bar{\mathbf{x}}} \frac{\partial \bar{\mathbf{x}}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_0}{\partial \bar{\mathbf{x}}} \frac{\partial \bar{\mathbf{x}}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_0}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_0}{\partial \mathbf{x}} \quad (5.6)$$

$$\bar{\mathbf{x}} = [\bar{\boldsymbol{\rho}}, \bar{\mathbf{r}}], \quad \tilde{\mathbf{x}} = [\tilde{\boldsymbol{\rho}}, \tilde{\mathbf{r}}], \quad \mathbf{x} = [\boldsymbol{\rho}, \mathbf{r}], \quad \bar{\mathbf{r}} = \tilde{\mathbf{r}} = \mathbf{r}$$

where $\bar{\mathbf{x}}$ are the variables after Heaviside thresholding, $\tilde{\mathbf{x}}$ are the filtered design variables and \mathbf{x} are the original design variables. The sensitivities are calculated with respect to original design variables \mathbf{x} .

With the adjoint the vector $\partial \bar{g}_0 / \partial \mathbf{u}$ becomes

$$\frac{\partial \bar{g}_0}{\partial \mathbf{u}} = (\mathbf{F} - \boldsymbol{\lambda}^T \mathbf{K}) \quad (5.7)$$

By setting $\boldsymbol{\lambda}$ to be

$$\boldsymbol{\lambda} = (\mathbf{K}^{-1} \mathbf{F})^T = \mathbf{u} \quad (5.8)$$

Equation 5.7 is assigned to 0, such that entire first term in equation 5.6 can be disregarded. The partial derivative with respect to the densities becomes instead

$$\frac{\partial \bar{g}_0}{\partial \bar{\boldsymbol{\rho}}} = \mathbf{F} \frac{\partial \mathbf{u}}{\partial \bar{\boldsymbol{\rho}}} - \frac{\partial \boldsymbol{\lambda}^T \mathbf{K} \mathbf{u}}{\partial \bar{\boldsymbol{\rho}}} = \mathbf{u}^T \mathbf{K} \frac{\partial \mathbf{u}}{\partial \bar{\boldsymbol{\rho}}} - \boldsymbol{\lambda}^T \frac{\partial \mathbf{K}}{\partial \bar{\boldsymbol{\rho}}} \mathbf{u} - \boldsymbol{\lambda}^T \mathbf{K} \frac{\partial \mathbf{u}}{\partial \bar{\boldsymbol{\rho}}} \quad (5.9)$$

Since $\boldsymbol{\lambda} = \mathbf{u}$, equation 5.9 is simplified to

$$\frac{\partial \bar{g}_0}{\partial \bar{\boldsymbol{\rho}}} = -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial \bar{\boldsymbol{\rho}}} \mathbf{u} \quad (5.10)$$

With the SIMP penalization the stiffness matrix for an element then becomes

$$\mathbf{K}_e = \mathbf{B}_e \mathbf{D}_e^k \mathbf{B}_e \quad (5.11)$$

$$\mathbf{D}_e^k = (\bar{\rho}_e^{q_k}(1 - \delta) + \delta)\mathbf{D}_0 \quad (5.12)$$

$$\frac{\partial \mathbf{D}_e^k}{\partial \bar{\rho}_e} = q_k \bar{\rho}_e^{q_k-1} (1 - \delta) \mathbf{D}_0 \quad (5.13)$$

\mathbf{D}_0 is the constitutive matrix for plane stress without any penalization. All the element stiffness matrices \mathbf{K}_e are then assembled into the global stiffness matrix \mathbf{K} . With this, the partial derivative of the stiffness matrix becomes

$$\frac{\partial \mathbf{K}}{\partial \bar{\rho}_e} = \mathbf{B}_e \frac{\partial \mathbf{D}_e^k}{\partial \bar{\rho}_e} \mathbf{B}_e \quad (5.14)$$

The other partial derivatives in equation 5.6 are

$$\begin{aligned} \frac{\partial \bar{g}_0}{\partial \bar{\mathbf{r}}} &= 0 & \frac{\partial \bar{g}_0}{\partial \tilde{\mathbf{x}}} &= 0 \\ \frac{\partial \bar{g}_0}{\partial \boldsymbol{\rho}} &= 0 & \frac{\partial \bar{g}_0}{\partial r_i} &= -2\gamma(r_c - r_i) \end{aligned} \quad (5.15)$$

which simplifies equation 5.6 to

$$\frac{D\bar{g}_0}{D\mathbf{x}} = \frac{\partial \bar{g}_0}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_0}{\partial \mathbf{x}} \quad (5.16)$$

The same density filter described in section 3.4 is used with some small changes since each element has their own length scale.

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e} w(x_i, r_e) v_i \rho_i}{\sum_{i \in N_e} w(x_i, r_e) v_e} \quad (5.17)$$

where N_e are all the elements within a distance lower then the length scale r_e of element e . The partial derivative of equation 5.17 respect to the densities becomes

$$\frac{\partial \tilde{\rho}_i}{\partial \rho_e} = \frac{w(x_e, r_e) v_e}{\sum_{j \in N_e} w(x_j, r_e) v_j} = W_\rho \quad (5.18)$$

and the partial derivative with respect to the length scale

$$\frac{\partial \tilde{\rho}_i}{\partial r_e} = \frac{\sum_{j \in N_e} \frac{\partial w}{\partial r_e} \rho_j v_j \sum_{j \in N_e} w(x_j, r_e) v_j - \sum_{j \in N_e} \frac{\partial w}{\partial r_e} v_j \sum_{j \in N_e} w(x_j, r_e) v_j \rho_j}{\left(\sum_{j \in N_e} w(x_j, r_e) v_j \right)^2} = W_r \quad (5.19)$$

Where $\partial w / \partial r_e$ is the partial derivative of the weight function. Since the length scale does not affect other elements except its own, equation 5.19 will be 0 for all values except for when $e = i$. Depending on which weight function is used, different derivatives are obtained. For a linear decaying function the derivative becomes

$$w(x_i, r_e) = r_e - |x_i - x_e|$$

$$\frac{\partial w}{\partial r_e} = 1 \quad (5.20)$$

and for a Gaussian

$$w(x_i, r_e) = e^{-\frac{1}{2}\left(\frac{|x_i - x_e|}{r_e/c}\right)^2}$$

$$\frac{\partial w}{\partial r_e} = e^{-\frac{1}{2}\left(\frac{|x_i - x_e|}{r_e/c}\right)^2} \frac{c^2 |x_i - x_e|^2}{r_e^3} \quad (5.21)$$

where c is a scaling factor that helps the Gaussian function to truncate at similar length scales as the cone shaped function as discussed in section 3.4.

The final partial derivatives of $\partial\tilde{\mathbf{x}}/\partial\mathbf{x}$ are calculated as

$$\frac{\partial\tilde{\mathbf{r}}}{\partial\rho} = 0, \quad \frac{\partial\tilde{\mathbf{r}}}{\partial\mathbf{r}} = \mathbf{I} \quad (5.22)$$

where \mathbf{I} is the identity matrix. With this, the partial derivative $\partial\tilde{\mathbf{x}}/\partial\mathbf{x}$ can be assembled into

$$\frac{\partial\tilde{\mathbf{x}}}{\partial\mathbf{x}} = \begin{bmatrix} \mathbf{W}_\rho & \mathbf{W}_r \\ 0 & \mathbf{I} \end{bmatrix} \quad (5.23)$$

The same procedure is used for $\partial\bar{x}/\partial\tilde{x}$. The approximated Heaviside step function is used to truncate $\bar{\rho}$ according to

$$\bar{\rho} = \frac{\tanh(B_H z) + \tanh(B_H(\bar{\rho} - z))}{\tanh(B_h z) + \tanh(B_h(1 - z))} \quad (5.24)$$

The partial derivative then becomes

$$\frac{\partial\bar{\rho}}{\partial\tilde{\rho}} = \frac{B_H(1 - \tanh(B_H(\bar{\rho}) - z))^2}{\tanh(B_H z) + \tanh(B_H(1 - z))} = H_\rho \quad (5.25)$$

and

$$\frac{\partial\bar{\mathbf{r}}}{\partial\tilde{\rho}} = \frac{\partial\bar{\rho}}{\partial\tilde{\rho}} = 0, \quad \frac{\partial\bar{\mathbf{r}}}{\partial\tilde{\mathbf{r}}} = \mathbf{I} \quad (5.26)$$

The partial derivative $\partial\bar{\mathbf{x}}/\partial\tilde{\mathbf{x}}$ can then be assembled into

$$\frac{\partial\bar{\mathbf{x}}}{\partial\tilde{\mathbf{x}}} = \begin{bmatrix} \mathbf{H}_\rho & 0 \\ 0 & \mathbf{I} \end{bmatrix} \quad (5.27)$$

The same procedure is done for the volume constraint g_1 which is defined as

$$g_1 = \sum \bar{\rho}_i v_e - V_{max} \quad (5.28)$$

Since the constraint is not dependent on the displacements, it is not necessary to add an adjoint vector. The total derivative of the volume constraint becomes

$$\frac{Dg_1}{D\mathbf{x}} = \frac{\partial g_1}{\partial\bar{\mathbf{x}}} \frac{\partial\bar{\mathbf{x}}}{\partial\tilde{\mathbf{x}}} \frac{\partial\tilde{\mathbf{x}}}{\partial\mathbf{x}} + \frac{\partial g_1}{\partial\tilde{\mathbf{x}}} \frac{\partial\tilde{\mathbf{x}}}{\partial\mathbf{x}} + \frac{\partial g_1}{\partial\mathbf{x}} = \frac{\partial g_1}{\partial\bar{\mathbf{x}}} \frac{\partial\bar{\mathbf{x}}}{\partial\tilde{\mathbf{x}}} \frac{\partial\tilde{\mathbf{x}}}{\partial\mathbf{x}} \quad (5.29)$$

since

$$\frac{\partial g_1}{\partial \tilde{\mathbf{x}}} = 0 \quad \frac{\partial g_1}{\partial \mathbf{x}} = 0 \quad (5.30)$$

The derivative with respect to $\bar{\rho}$ and $\bar{\mathbf{r}}$ becomes

$$\frac{\partial g_1}{\partial \bar{\rho}_e} = v_e, \quad \frac{\partial g_1}{\partial \bar{\mathbf{r}}} = 0 \quad (5.31)$$

$\partial \bar{\mathbf{x}}/\partial \tilde{\mathbf{x}}$ and $\partial \tilde{\mathbf{x}}/\partial \mathbf{x}$ are the same as equation 5.23 and 5.27

Calculating the sensitivities of the stresses can be quite costly because of the implicit derivatives that appear. To circumvent this another adjoint vector is added to g_2 in a similar fashion as in g_0 . The stress constraint is then expressed as

$$\bar{g}_2 = \sigma^{PN} - \sigma_{max} - \mathbf{\Lambda}^T(\mathbf{K}\mathbf{u} - \mathbf{F}) \quad (5.32)$$

where $\mathbf{\Lambda}$ is the adjoint vector for the stress constraint. The sensitivity of the constraint g_2 is calculated as

$$\frac{D\bar{g}_2(\mathbf{u}(\mathbf{x}), \mathbf{x})}{D\mathbf{x}} = \frac{\partial \bar{g}_2}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \bar{\mathbf{x}}} \frac{\partial \bar{\mathbf{x}}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_2}{\partial \bar{\mathbf{x}}} \frac{\partial \bar{\mathbf{x}}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_2}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} + \frac{\partial \bar{g}_2}{\partial \mathbf{x}} \quad (5.33)$$

Where

$$\frac{\partial \bar{g}_2}{\partial \tilde{\mathbf{x}}} = \frac{\partial \bar{g}_2}{\partial \mathbf{x}} = 0 \quad (5.34)$$

Since there is no direct relation between the length scale and the stresses, meaning that

$$\frac{\partial \bar{g}_2}{\partial \bar{\mathbf{r}}} = \frac{\partial \bar{g}_2}{\partial \tilde{\mathbf{r}}} = \frac{\partial \bar{g}_2}{\partial \mathbf{r}} = 0$$

The design variable $\tilde{\mathbf{x}}$ will be replaced with $\bar{\rho}$ throughout the derivation when calculating the stress sensitivities.

Since many transformations are used to get the P-norm stress, a few implicit derivatives have to be calculated to get the derivative of g_2 with respect to the densities. It will be defined as

$$\frac{\partial \bar{g}_2}{\partial \bar{\rho}_e} = \sum_{i=1}^{Nelm} \left(\frac{\partial \sigma^{PN}}{\partial \sigma_i^{VM}} \left(\frac{\partial \sigma_i^{VM}}{\partial \sigma_i} \right)^T \frac{\partial \sigma_i}{\partial \bar{\rho}_e} \right) - \mathbf{\Lambda}_e^T \frac{\partial \mathbf{K}}{\partial \bar{\rho}_e} \mathbf{u}_e \quad (5.35)$$

Where each partial derivative, along with $\mathbf{\Lambda}$ have to be calculated. The strains and stresses are evaluated by the nodal displacements, the shape functions and the constitutive matrix as

$$\begin{aligned} \boldsymbol{\epsilon}_e^T &= \mathbf{B}_e \mathbf{u}_e \\ \boldsymbol{\sigma}_e &= \mathbf{D}_e^s \boldsymbol{\epsilon}_e \end{aligned} \quad (5.36)$$

Where \mathbf{D}_e^s is calculated in the same manner as equation 5.12 but with a different exponent. Intermediate values are penalized by the factor $q_s = 0.5$ instead, as discussed in section 3.3. The stresses are evaluated at 4 Gauss points in each element. Since the

nodal displacements are also dependent on the densities, the derivative of the stresses becomes

$$\frac{\partial \sigma_i}{\partial \bar{\rho}_e} = \frac{\partial \mathbf{D}_e^s}{\partial \bar{\rho}_e} \mathbf{B}_i \mathbf{u} + \mathbf{D}_e^s \mathbf{B}_i \frac{\partial \mathbf{u}}{\partial \bar{\rho}_e} \quad (5.37)$$

The partial derivative of the nodal displacement with regards to the densities is calculated from the global state equation.

$$\begin{aligned} \mathbf{K} \mathbf{u} - \mathbf{F} = 0, \quad \frac{\partial \mathbf{K}}{\partial \bar{\rho}} \mathbf{u} - \mathbf{K} \frac{\partial \mathbf{u}}{\partial \bar{\rho}} = 0 \\ \frac{\partial \mathbf{u}}{\partial \bar{\rho}} = \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \bar{\rho}} \mathbf{u} \end{aligned} \quad (5.38)$$

Where $\partial \mathbf{K} / \partial \bar{\rho}$ is calculated according to equation 5.14

For plane stress conditions the von Mises stresses are calculated as

$$\sigma_e^{VM} = \sqrt{\sigma_{e,xx}^2 + \sigma_{e,yy}^2 - \sigma_{e,xx} \sigma_{e,yy} + 3\tau_{e,xy}^2} \quad (5.39)$$

Where the partial derivatives of the von Mises stresses with regards to the different stress components becomes

$$\begin{aligned} \frac{\partial \sigma_e^{VM}}{\partial \sigma_{e,xx}} &= \frac{1}{2\sigma_e^{VM}} (2\sigma_{e,xx} - \sigma_{e,yy}) \\ \frac{\partial \sigma_e^{VM}}{\partial \sigma_{e,yy}} &= \frac{1}{2\sigma_e^{VM}} (2\sigma_{e,yy} - \sigma_{e,xx}) \\ \frac{\partial \sigma_e^{VM}}{\partial \tau_{e,xy}} &= \frac{3}{\sigma_e^{VM}} (\tau_{e,xy}) \end{aligned} \quad (5.40)$$

The final partial derivative to calculate in equation 5.35 is the derivative of the P-norm stress with respect to the von Mises, and is calculated as

$$\frac{\partial \sigma^{PN}}{\partial \sigma_e^{VM}} = \left(\sum_{i=1}^{Nelm} (\sigma_i^{VM})^p \right)^{\frac{1}{p}-1} (\sigma_e^{VM})^{p-1} \quad (5.41)$$

With the introduction of the adjoint vector, the partial derivative with respect to the nodal displacements becomes

$$\frac{\partial \bar{g}_2}{\partial \mathbf{u}} = \sum_{i=1}^{Nelm} \frac{\partial \sigma^{PN}}{\partial \sigma_i^{VM}} \left(\frac{\partial \sigma_i^{VM}}{\partial \sigma_i} \right)^T \mathbf{D}_i^s \mathbf{B}_i - \Lambda^T \mathbf{K} = 0 \quad (5.42)$$

$$\Lambda = \left(\sum_{i=1}^{Nelm} \frac{\partial \sigma^{PN}}{\partial \sigma_i^{VM}} \mathbf{B}_i^T (\mathbf{D}_i^s)^T \frac{\partial \sigma_i^{VM}}{\partial \sigma_i} \right) \mathbf{K}^{-1} \quad (5.43)$$

By setting lambda to be equal to equation 5.43, equation 5.42 and the first term in equation 5.33, becomes zero. This alleviates the need to calculate $\partial \mathbf{u} / \partial \bar{\rho}$ and simplifies equation 5.33 to

$$\frac{D\bar{g}_2(\mathbf{u}(\mathbf{x}), \mathbf{x})}{D\mathbf{x}} = \frac{\partial \bar{g}_2}{\partial \bar{\rho}} \frac{\partial \bar{\mathbf{x}}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial \mathbf{x}} \quad (5.44)$$

Equation 5.37, 5.40, 5.41 together with $\mathbf{\Lambda}$ can be used to calculate $\partial \bar{g}_2 / \partial \bar{\rho}$ in the equation above. $\partial \bar{\mathbf{x}} / \partial \tilde{\mathbf{x}}$ and $\partial \tilde{\mathbf{x}} / \partial \mathbf{x}$ are the same as equations 5.23 and 5.27

5.3 Parameters

To compare the results of the simulations where a constant length scale was used, an average length scale was calculated for the elements with solid material in the case of the dynamic length scale. The material parameters $E = 23000Pa$, $\nu = 0.3$ and a load of $F = 100N$ was used. A move limit of 1 and 0.5 was also introduced on the densities and the length scale respectively, since the optimizer would not converge properly. Only a few of the figures can be seen in the next chapter since a lot of the structure look similar. The ones chosen show some distinct differences in the length scale field. All figures can be seen in Appendix A and B. In all the figures showing the density distribution, yellow represent solid material and blue represents void. A mesh of 25600 elements was used in all the simulations in this section and in the Appendix (200 elements along the side L in figure 16)

The linear weight function and the Gaussian weight function were tested on \mathbb{P}_1 to see if any difference could be noted.

The Heaviside function was updated every 10^{th} iteration according to the vector

$$\mathbb{B} = [0.001, 0.1, 0.5, 1.2, 2, 3, 4, 5, 6, 7.5, 8, 8.3, 9, 10, 10.5, 11]$$

and the truncation always occurred at $z = 0.5$ in the thesis. The vector was generated empirically, when the code was tested with a small amount of elements, and the numbers were tweaked if the simulations did not converge at a certain value for B_H .

6 Results

The results shown are split into 2 subsections, one for each of the different problems. To show how the values and structures differ from the standard topology optimization approach, meaning only the densities are treated as a design variable, each subsection is divided into 2 parts. A constant length scale, which is the standard approach with no variation in space and is not updated between iterations, and a dynamic length scale, which uses the implementation discussed in section 5.1 where the length scale is treated as a design variable.

6.1 Compliance minimization

6.1.1 Constant Length scale

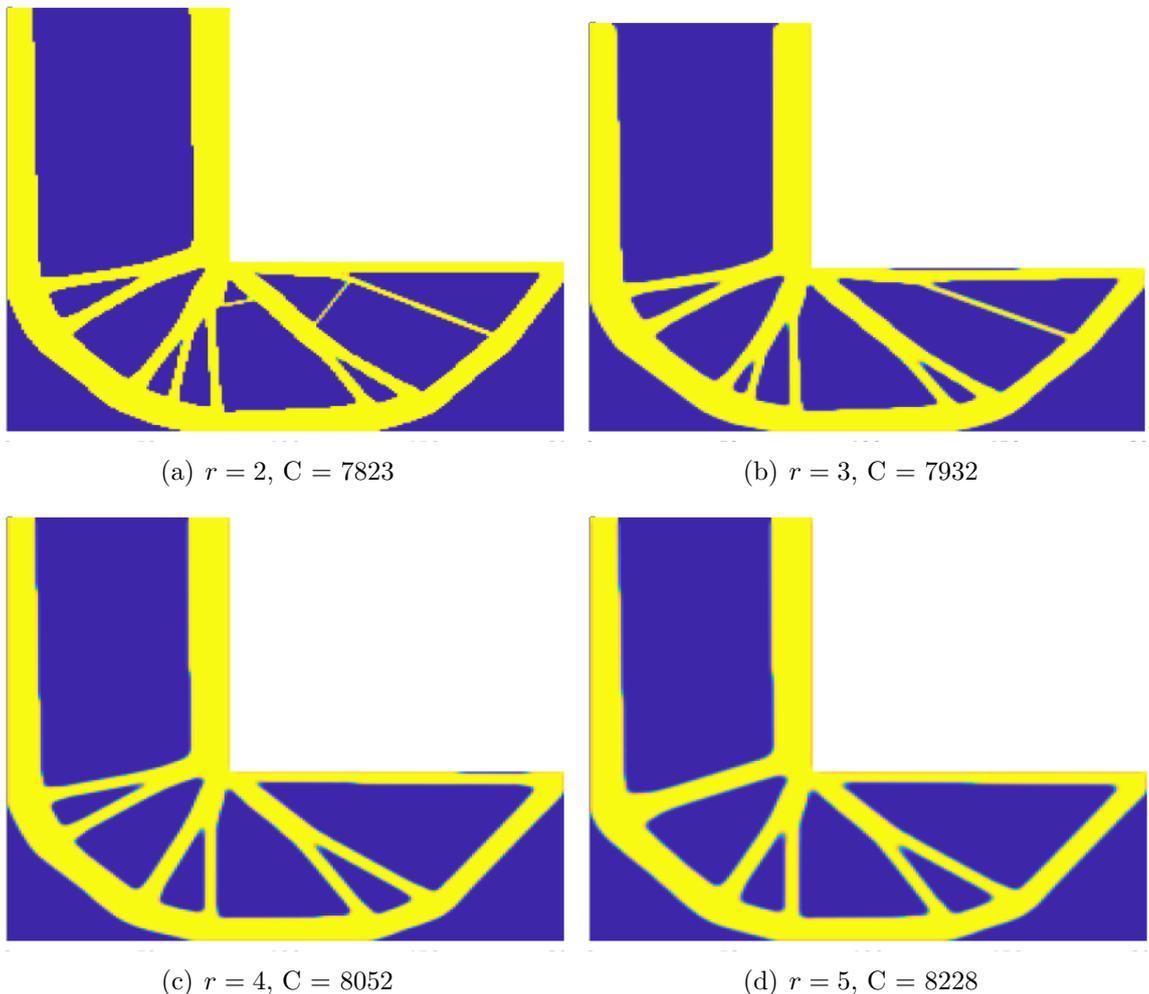


Figure 22: Compliance minimization with a volume constraint, with different constant length scales and a linear cone shaped weighting function.

6.1.2 Dynamic Length scale

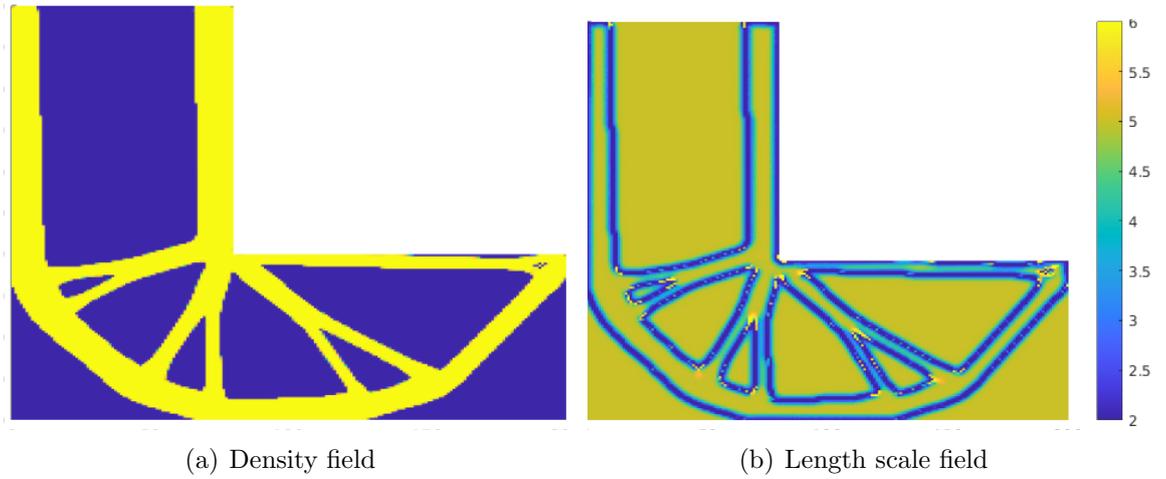


Figure 23: Optimization of \mathbb{P}_1 with $\gamma = 0.0001$, $C = 7868$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$ and a linear cone shaped weighting function.

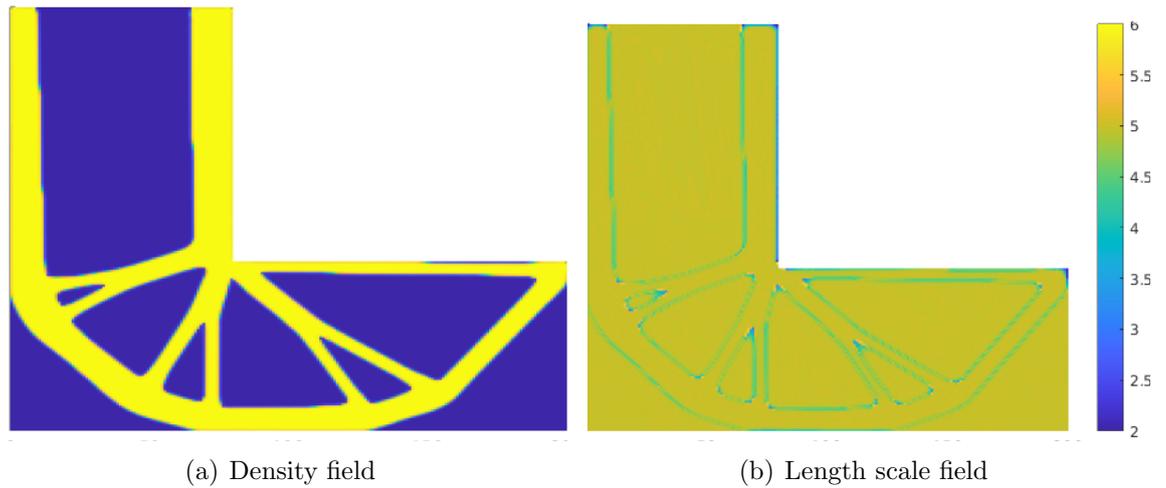


Figure 24: Optimization of \mathbb{P}_1 with $\gamma = 0.05$, $C = 8072$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$ and a linear cone shaped weighting function.

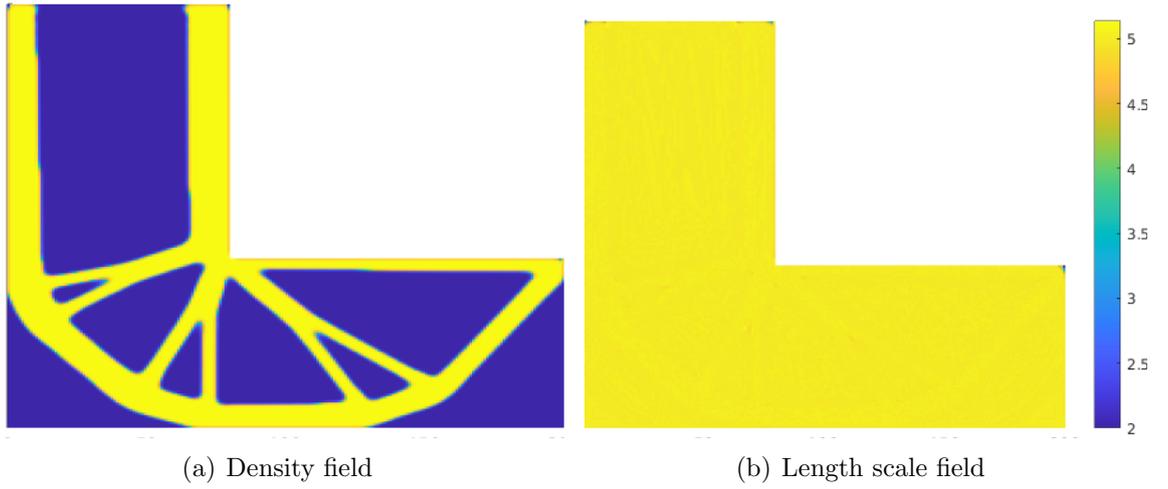


Figure 25: Optimization of \mathbb{P}_1 with $\gamma = 1$, $C = 8192$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$ and a linear cone shaped weighting function.

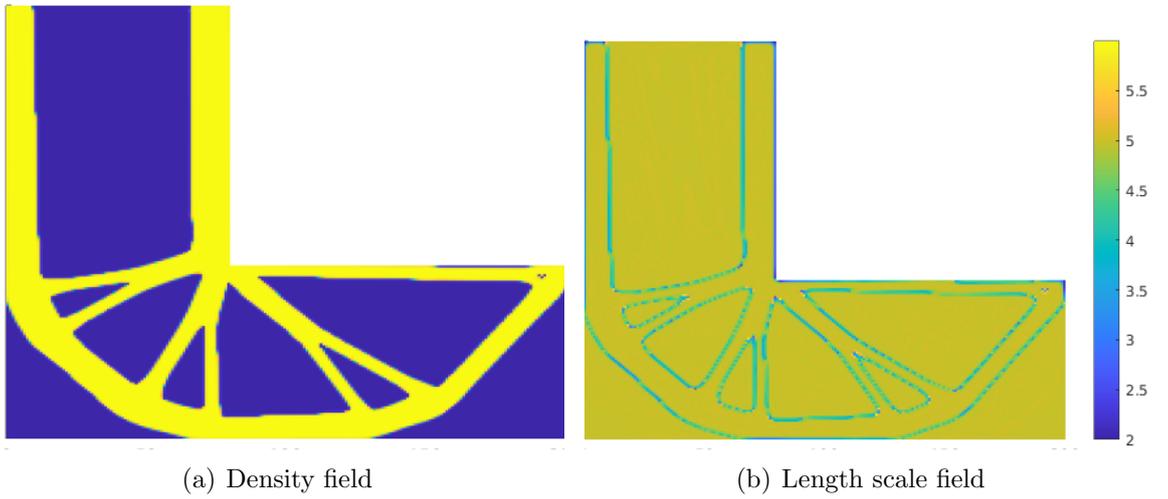


Figure 26: Optimization of \mathbb{P}_1 with $\gamma = 0.01$, $C = 7918$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$ and a Gaussian bell-shaped weighting function.

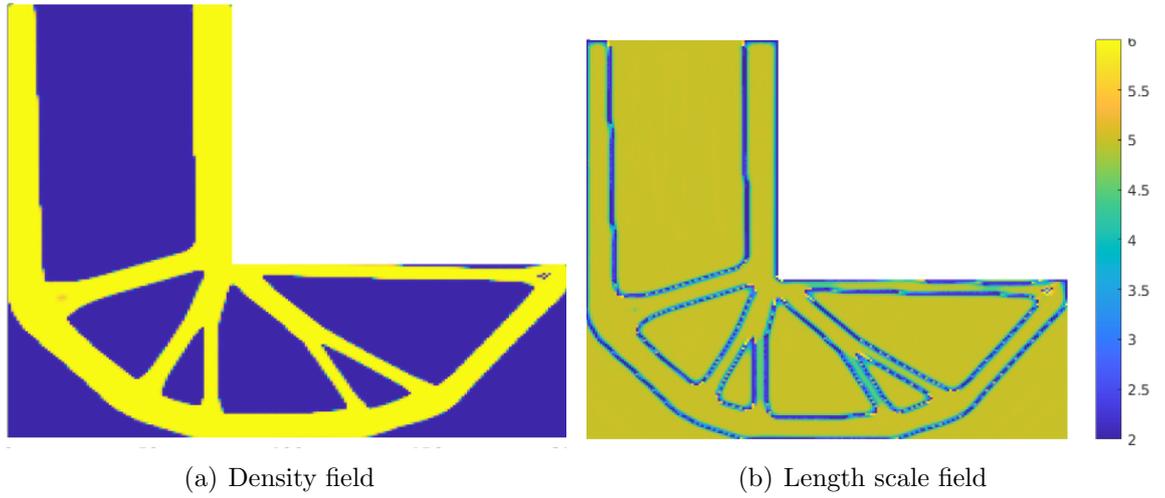


Figure 27: Optimization of \mathbb{P}_1 with $\gamma = 0.005$, $C = 7909$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$ and a Gaussian bell-shaped weighting function.

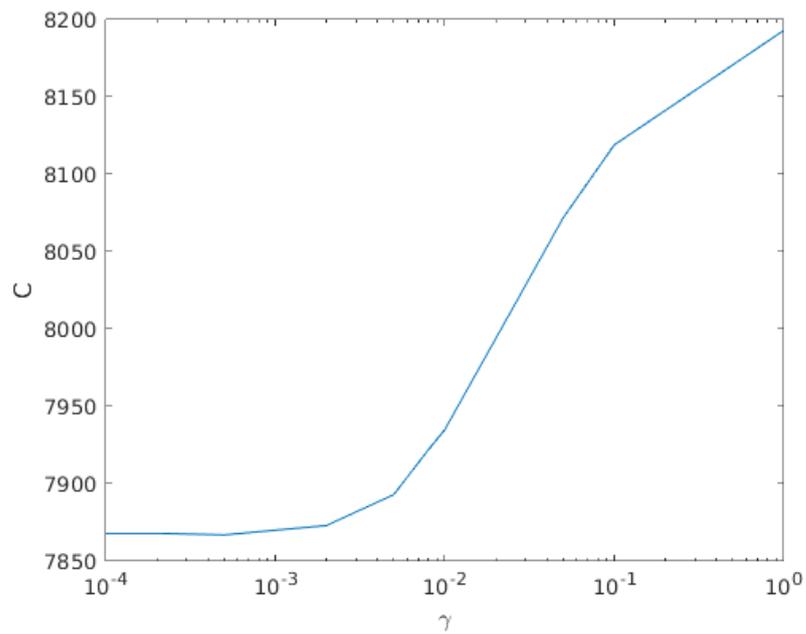


Figure 28: Plot showing how the compliance of structure changes with respect to γ , with linear cone shaped weighting function. The penalty factor γ is scaled logarithmically. $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

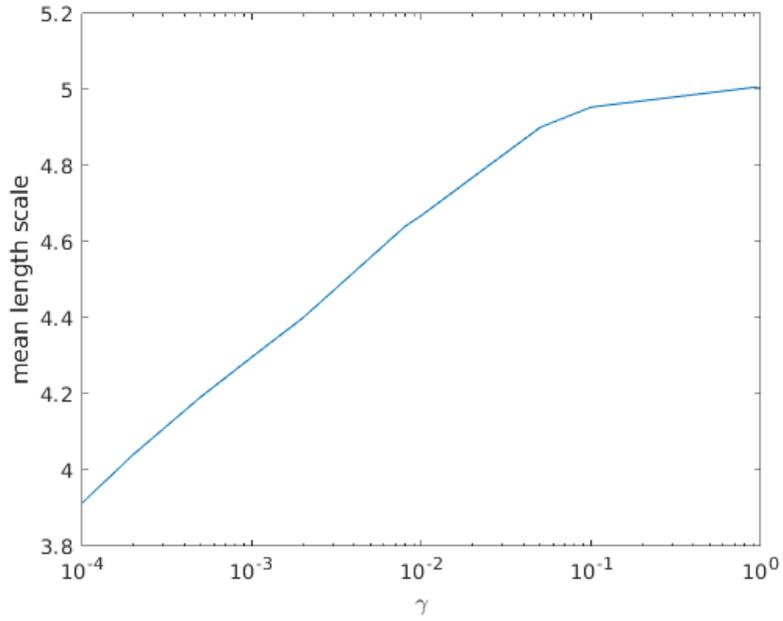


Figure 29: Plot showing how the mean radius of structure changes with respect to γ , with linear cone shaped weighting function. The penalty factor γ is scaled logarithmically. $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

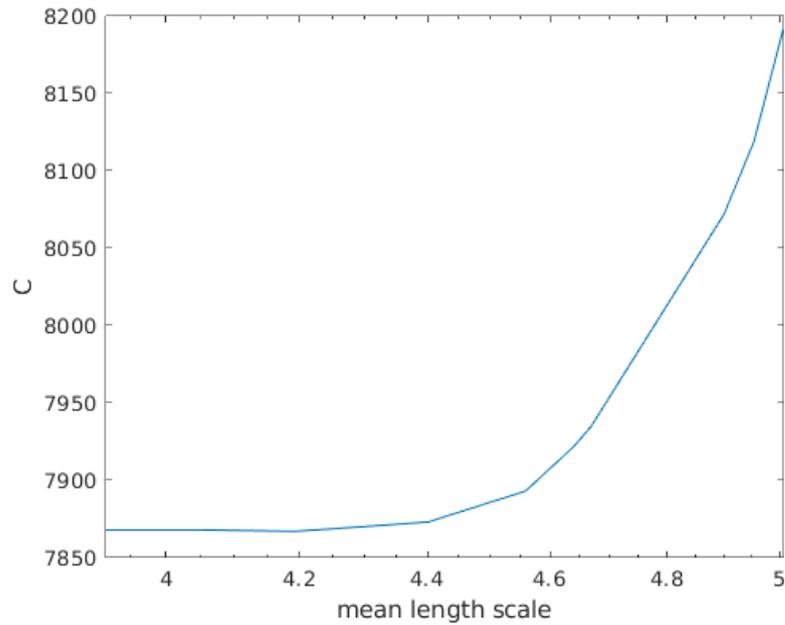


Figure 30: Plot showing how the compliance C in structure changes with respect to the mean radius, with linear cone shaped weighting function. $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

6.2 Compliance minimization, with stress constraint

Comparing the mean length scale to γ was not done for \mathbb{P}_2 , because of the increase in parameters.

6.2.1 Constant Length scale

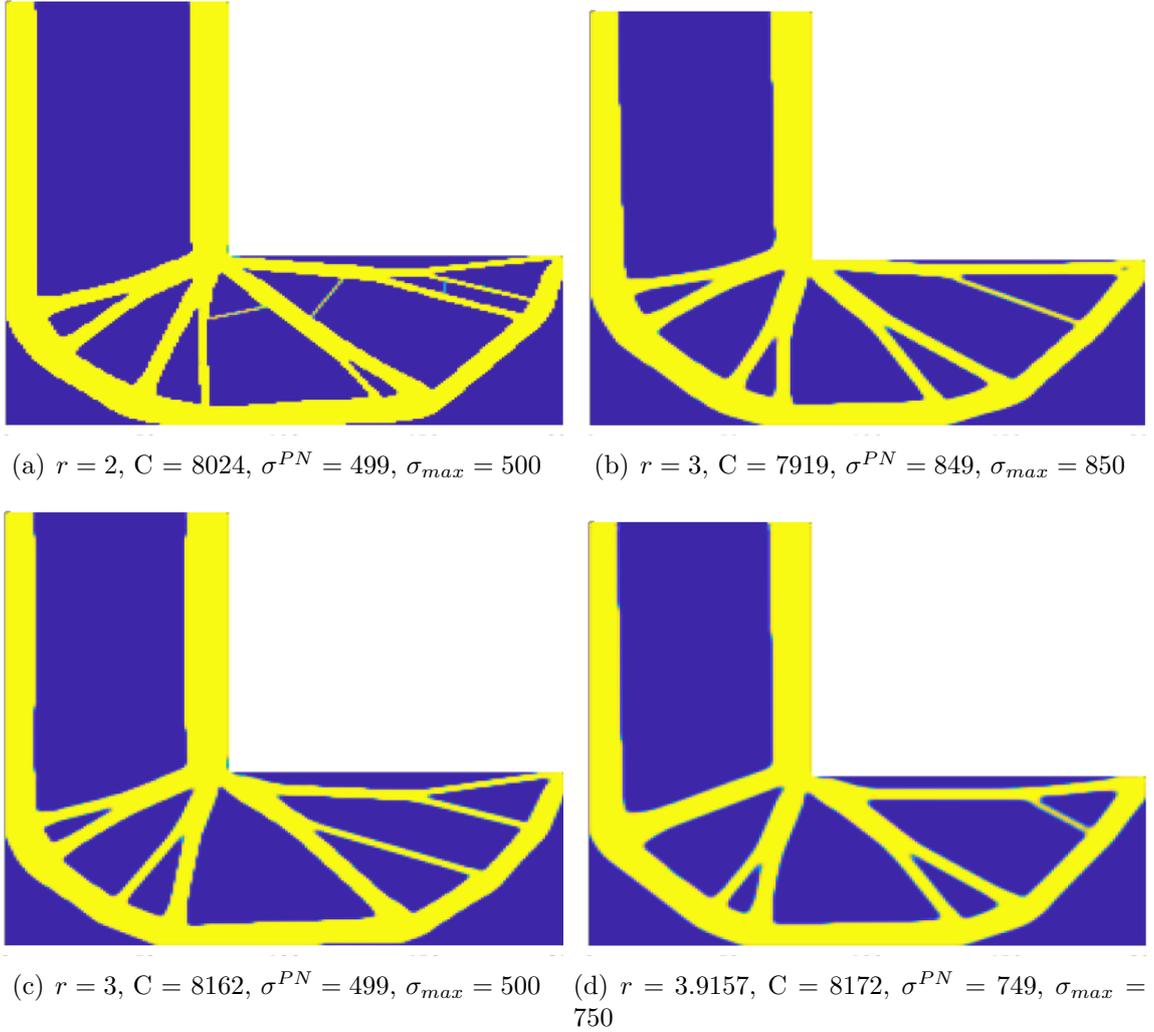


Figure 31: Compliance minimization with a volume constraint of 35% and a stress constraint, with different constant length scales and a linear cone shaped weighting function.

6.2.2 Dynamic Length scale

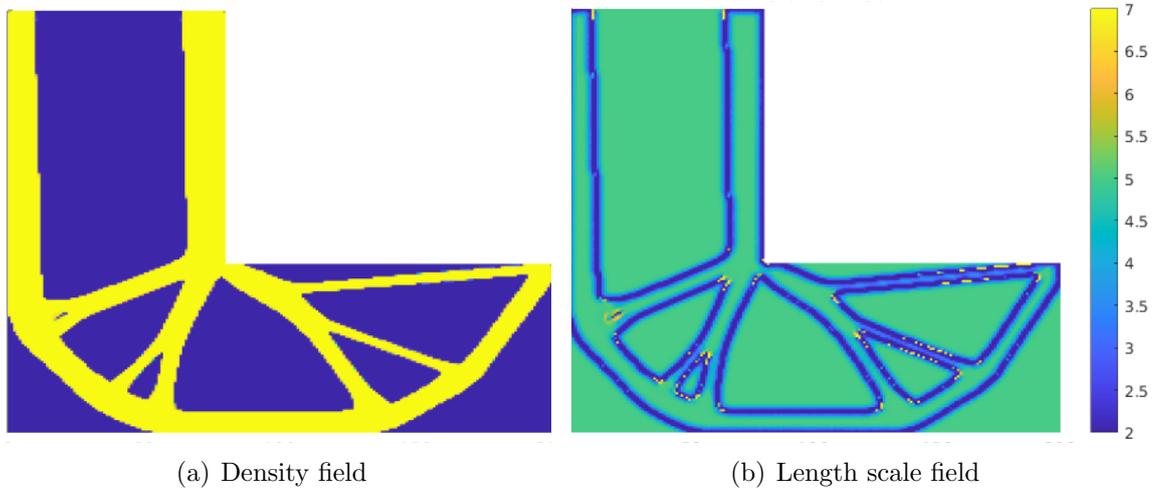


Figure 32: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8018$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 849$, $\sigma_{max} = 850$, $r_{mean} = 3.9103$

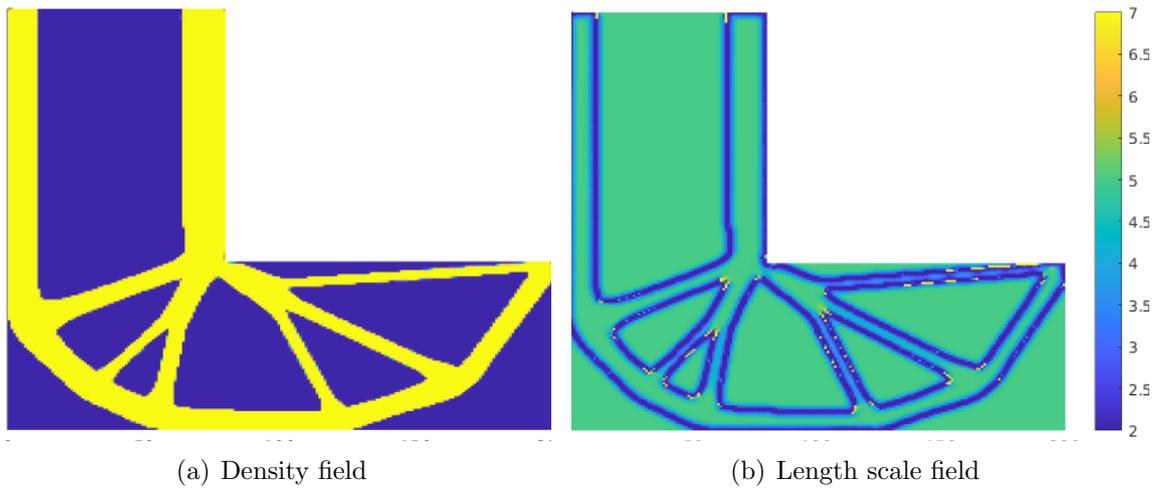


Figure 33: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8047$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 600$, $\sigma_{max} = 600$, $r_{mean} = 3.8719$

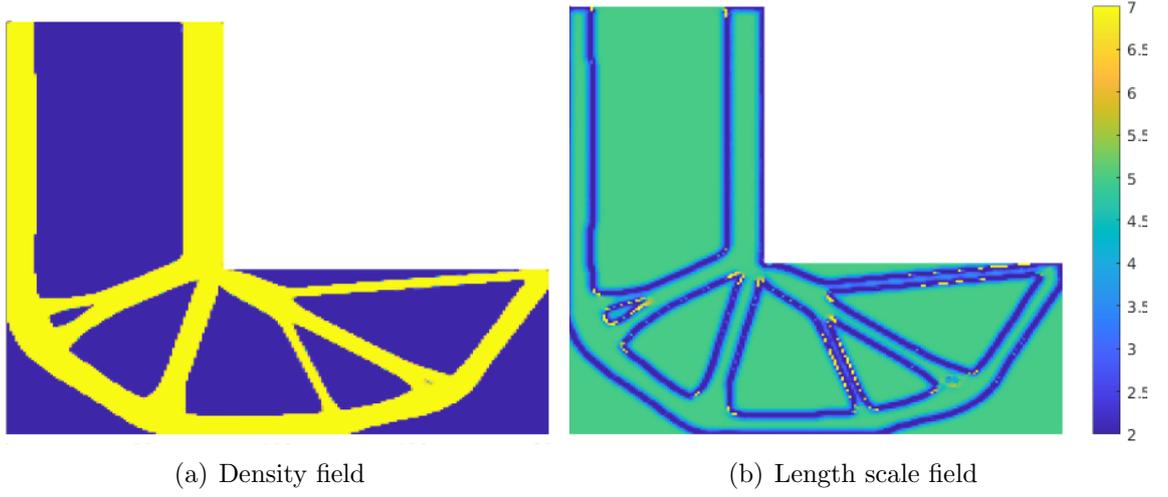


Figure 34: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8028$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 749$, $\sigma_{max} = 750$, $r_{mean} = 3.9157$

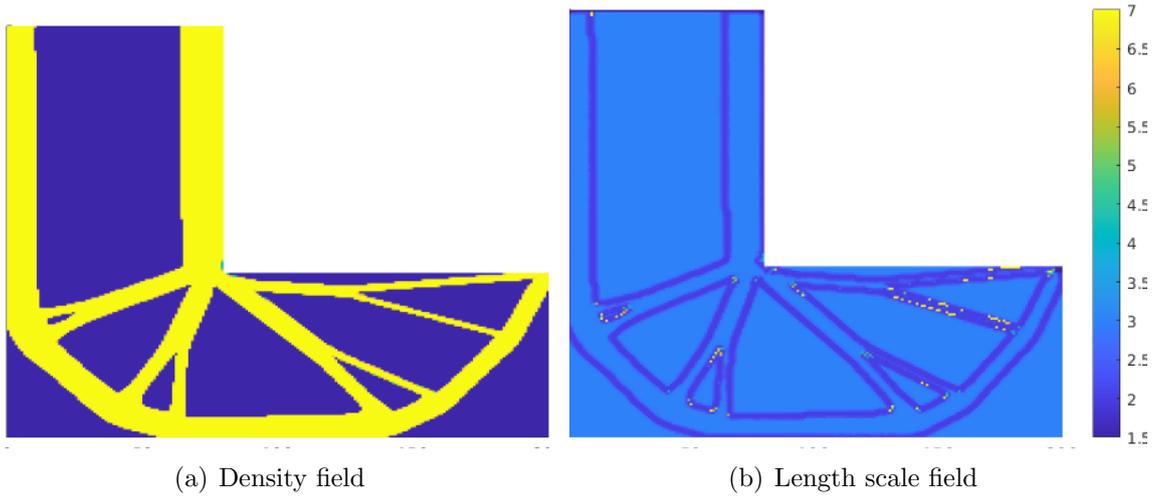


Figure 35: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 7979$, $r_c = 3$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 499$, $\sigma_{max} = 500$, $r_{mean} = 2.6619$

7 Discussion

From the results, some general trends could be noted for the different parameters which will be discussed for problem \mathbb{P}_1 and \mathbb{P}_2 . The problems and benefits of implementing this type of optimizations is then discussed, where in that section, the linear and Gaussian weight function are examined to understand why they have similar behaviour.

7.1 Optimal length scale

7.1.1 Compliance minimization

Comparing the density field and the length scale field between figures 23, 24 and 25 it can be noted that the final structures do look similar for problem \mathbb{P}_1 . The parameter γ essentially controls the extent the optimizer is allowed to place r_{min} from the inner bars in the structure. It can be seen in figure 29 there is a quite narrow regime where γ is very sensitive. This is also where some distinction can be made between the different structures. Mainly that the optimizer almost wants to place what appears to be checkerboard patterned holes where the load is applied, and that the bar from the inner corner of the bracket to the corner where the load is applied becomes more and more horizontal, probably because of the harsher constraints on length scale. These changes are illustrated in figure 36 and 37. It can also be seen that the thickness of the bars are all similar to each other, suggesting that thickness of each feature is heavily controlled by the initial length scale r_c .

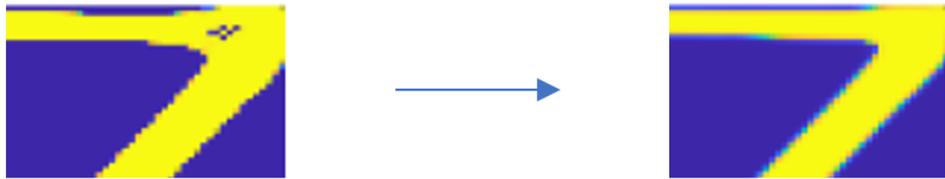


Figure 36: The removal of checkerboard with increasing γ



Figure 37: The bar from the inner corner to where load is applied becoming straight with increasing γ

7.1.2 Compliance minimization, with a stress constraint

Different parameter for \mathbb{P}_2 were tested to see how the final structure was affected by them. Again, the mean radius was calculated to compare it to the constant length scale. With the parameters used in figure 34, a mean radius of 3.9157 was achieved and a compliance of $C = 8028$. A constant length scale of 3.9157 was then tested where a compliance of $C = 8172$ was achieved, an increase of about 1.8% as can be seen in figure 31(d). Again, some increase in performance but unclear if the “mean radius” is a fair comparison between them. A quite hard stress constraint was applied and the initial length scale was changed from 5 to 3, to help the optimizer a bit. The result of this can be seen in figure 38(c). This was then compared to the values from figure 38(a) where the same stress constraint was applied with a constant length scale of 2. It is quite clear that the structure from figure 38(a), which used the constant length scale of 2, is harder to manufacture due to the extremely thin bars that appear. Interestingly the dynamic length scale achieved a compliance of $C = 7979$ and the constant achieved $C = 8024$, having a more realistic structure and being stiffer at the same time. Instead comparing it to the structure from figure 38(b) similar curvature is achieved but a less stiff structure ($C = 8162$).

When it came to the shape of the structure, the initial length scale seemed to be the determining factor, where γ basically refined the edges depending on its size, similar to what happened in \mathbb{P}_1 .

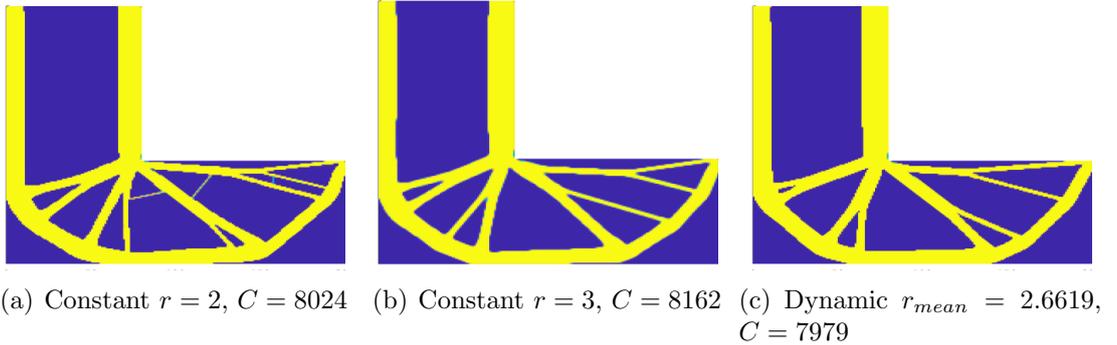


Figure 38: 3 different optimizations of \mathbb{P}_2 $\sigma_{max} = 500$

7.2 Computation time and implementation

What is quite nice with this approach is the ease of implementation since it rarely affects the sensitivities directly, unless modified such as equation 5.2, it becomes an implicit matrix that is calculated at the same time as the weight matrix. It is unclear though if optimizing the length scale is such a benefit for problem \mathbb{P}_1 and \mathbb{P}_2 , in the manner done in this thesis, since it is hard to evaluate if the manufacturing costs are lowered. The mean radius used in the graphs to compare the results may not be an accurate representation of manufacturing costs since it can be seen from the solver it almost always places the minimum length scale along the edges of the structure, allowing sharp corners. The solver is also somewhat slow since the increase

in computation comes from updating the length scale requires equation 5.23 to be recalculated every iteration. Perhaps it is not necessary to update the length scale every iteration. If it were updated every 10th iteration instead, at least at later stages in the optimization, it would lower the computational time quite significantly.

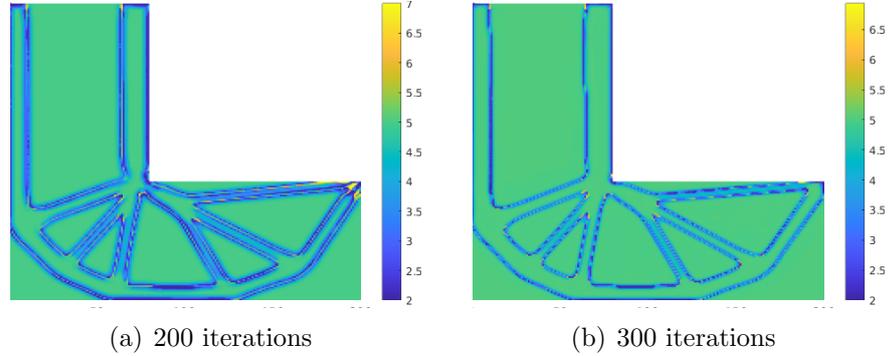
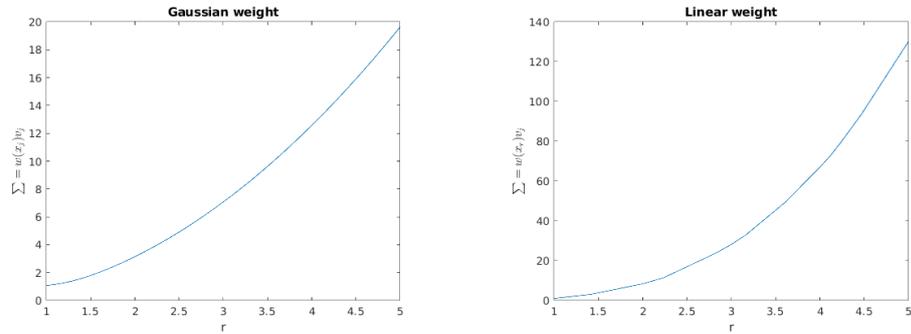


Figure 39: Difference in length scale field over 100 iterations

The continuous Gaussian weighting function may in theory be more accurate than the cone shaped weighting function since the bell shaped function is continuously differentiable. Only a few simulations were done with the continuous bell shaped function, since no large benefit was seen with it and the computational costs were quite significant since the length scaled varied with each iteration, meaning that the weight matrix had to be recalculated each time. The cone shaped function is noncontinuous since increasing the length scale introduces new elements in the sum that were not accounted for before. This can be seen in figure 40



(a) Plot of the function 7.1 with the bell shaped weighting function according to 7.2

(b) Plot of the function 7.1 with cone shaped weighting function according to 7.3

Figure 40: Comparison of how the sum 7.1 varies with increasing length scale from 1 to 5 element widths. The element used for the calculation was in the middle of a square with a side length of 101 elements widths

$$\sum_{j \in N_i} w(x_j) v_j \tag{7.1}$$

$$w(x_j) = e^{-\frac{1}{2}\left(\frac{|x_j - x_e|}{r}\right)^2} \quad (7.2)$$

$$w(x_j) = r - |x_j - x_e| \quad (7.3)$$

However the cone shaped function does appear to become continuous for larger values of the length scale. By also limiting the change in length scale between each iteration, the drawbacks of being noncontinuous are counteracted. This is believed to be the reason why they behaved similarly.

7.3 Locally modified length scale

It was noted from one of the simulations done in section 6.1.2 that even though the penalty factor γ was relatively large, the optimizer still chose to place the lowest allowed length scale on the corner where the load was applied, see figure 25(b). With that observation, the following is a suggestion of how to remove the need of padding while still having a consistent minimum feature size.

As discussed in section 3.4 padding the structure with void elements will help the algorithm not to favor edges and ensure uniform feature sizes, since each element will be compared to the same number of other elements during the regularization scheme. With this also comes the need of padding with solid elements to ensure contact at critical places as was seen in figure 14.

However, a quite simple alternative method to this is to locally change the length scale at these points to not include the elements outside the design domain in the calculation. This allows the optimizer to place material and support at these locations without them being removed after the filtering scheme and without the need for solid elements. An example of this can be seen in figure 41.



Figure 41: Length scale field of an L-bracket with locally smaller length scale at the tip where the load is applied, and at the top where there are boundary conditions.

The same load and boundary conditions are used as in the problem formulation. The corner is then locally modified with a length scale of 1.5 element widths to not filter the connection with the load. Length of 2 element widths has also been used with success. If the length is decreased to 1, checkerboard patterns start to appear. The same modified length scale is used on the top of the L-bracket because of the boundary conditions in the structure and again to not filter away the connection with the corners as in figure 14. Calculating the weighted densities would then go as follows.

1. Make a search box around the current element e that is the width of $2r_e$ as can be seen in figure 42
2. The distance from the center of e to the center of the neighboring elements are calculated.
3. A check is made to see if the element is within the design domain and if the distance is smaller than r_e .
4. If the element is within the design domain, it is treated as normal, if the element is outside only the bottom sum in equation 3.11 is updated.

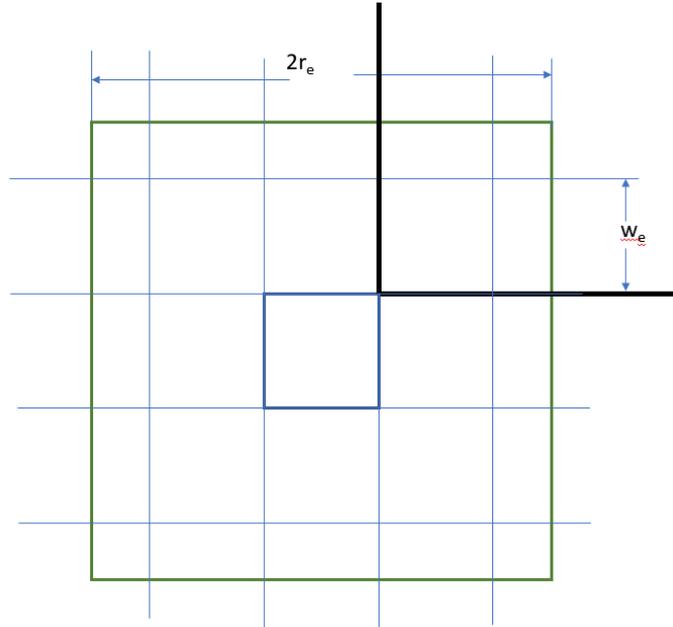


Figure 42: The search box around the element at the inner corner of the L-bracket. The black lines are the edges of the design domain, thinner blue squares are neighboring nodes, thicker blue square is the element e and w_e is the element width

Mathematically this can be described as

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e^{inner}} w(x_i) v_i \rho_i}{\sum_{i \in N_e} w(x_i) v_i} \quad (7.4)$$

Where N_e^{inner} are all the elements within r_e and inside the design domain and N_e are all elements within r_e . This is essentially the same as the elements outside the design domain being void but does not make it necessary to create the elements. Instead the elements are treated as geometric objects during the filter calculations with coordinates used to calculate the weighted densities. This was quite effective at calculating the neighboring nodes since the length scale was updated. This might be used with some effect with the Gaussian weight function as well. If a box is used with a side length slightly bigger than double the maximum allowed length scale, the same elements in the box are used every iteration. The influence then comes from where the truncation occurs (the size of r_e). An added benefit is the lower computational time since not every element is used in the filter calculation and that the number of elements stay the same.

Some problems were noted when there is a large difference between the locally modified length scale and length scale in the structure as can be seen in figure 43.

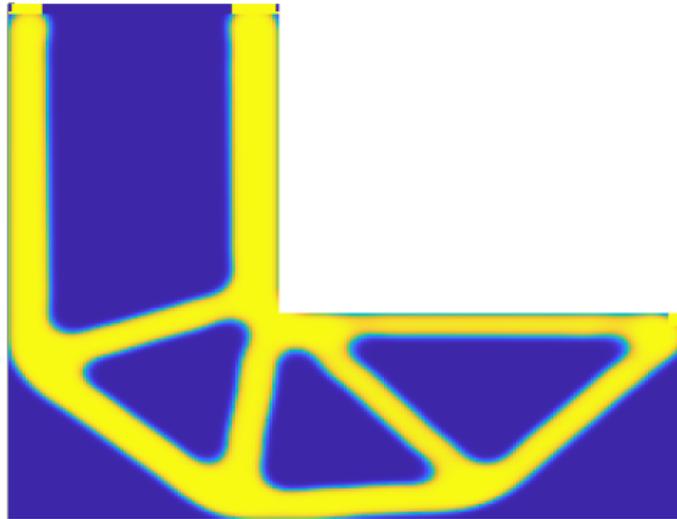


Figure 43: Compliance minimization with locally modified length scale filed according to figure 41, with a minimum length scale of 1.5 element widths and 12 element widths in the rest of the structure.

This probably occurs due to the sharp transition between the different length scales. This is solved by changing the transition to gradually change either linearly or with a bell shape, from the modified edges and corners. The effect of this can be seen in figure 44 and 45. An added benefit of this is since the void is not an actual element, but the space is just treated as one, the finite element analysis can be somewhat sped up since the padded elements does not have to be accounted for. In the case of the L-bracket with 200 elements vertically and 200 elements horizontally with the same proportions as figure 16 there exists 25600 elements. Adding 5 element as padding around the structure add 3650 elements to the FEM analysis, an increase of about 14%. The L-bracket does have a big ratio between the design area and the edge length. The number of elements needed for padding other structure such as the MBB beam, which also a standard shape used to evaluate structural optimization solutions,

is smaller.



Figure 44: Modified length scale field which increase linearly with the distance from the corner and the edge up to 12 element widths

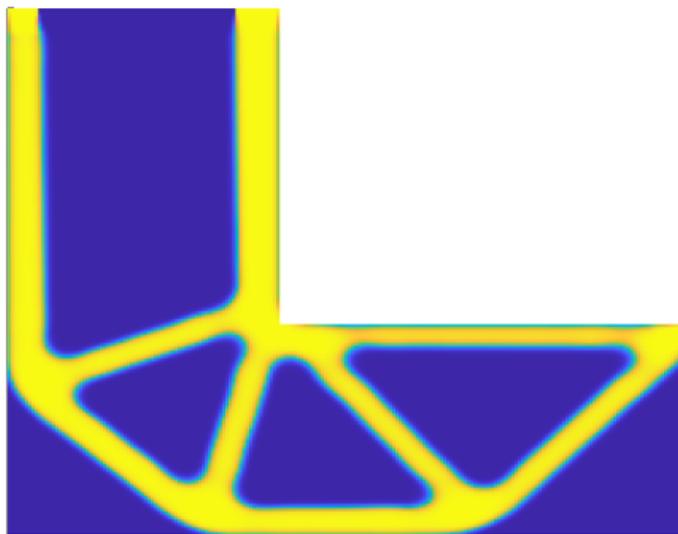


Figure 45: Compliance minimization with the length scale field in figure 44 with the filtered densities calculated according the method described in this chapter.

8 Future Work

The idea of optimizing the length scale could also be extended to other filters such as the Helmholtz PDE filter or the modified PDE filter cf. e.g. [12].

Something that was also realised quite late in the process was that the geometrical elements used outside the design domain to get consistent length scale don't have to be void elements. Instead they could be solid or some intermediate value. The weighting function 5.17 is then altered to

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e^{inner}} w(x_i)v_i\rho_i}{\sum_{i \in N_e} w(x_i)v_i} + \frac{\sum_{i \in N_e^{outer}} w(x_i)v_i\rho_i}{\sum_{i \in N_e} w(x_i)v_i} \quad (8.1)$$

Where N_e^{outer} are all the element within r_e and outside the design domain. The values from the outer function could then be added into a vector \mathbf{T} , which would change equation 5.17 into

$$\tilde{\rho} = \mathbf{W}_\rho \tilde{\rho} + \mathbf{T} \quad (8.2)$$

If all the densities outside the design domain are 0, \mathbf{T} will also be 0 and the same expression as in equation 7.4 is achieved. Perhaps even negative values could be used for the densities outside the design domain, to discourage the structure sticking to the edge.

9 References

References

- [1] N Saabye Ottosen and M Ristinmaa, *The mechanics of constitutive modeling*, Elsevier Amsterdam, 2005
- [2] P. Christensen and A. Klarbring, *An introduction to structural optimization*, Solid Mechanics and Its Applications, Springer Netherlands, 2008.
- [3] K. Svanberg, *The method of moving asymptotes – a new method for structural optimization*, International Journal for Numerical Methods in Engineering, 1987, 24:359-373.
- [4] E. Bruns and D Tortorelli *Topology optimization of non-linear elastic structures and compliant mechanisms.*, Computer Methods in Applied Mechanics and Engineering, 2001 190(26-27):3443–3459
- [5] M. Wang and S. Wang *Bilateral filtering for structural topology optimization*, Numerical Methods for Engineering, 2005, 63:1911-1938
Bilateral filtering for structural topology optimization
- [6] A. Clausen and E. Andreassen *Topology optimization of non-linear elastic structures and compliant mechanisms*, Structural and Multidisciplinary Optimization, 2001, 190:3443-3459.
- [7] O. Sigmund, *Morphology-based black and white filters for topology optimization*, Structural and Multidisciplinary Optimization, 2007, 33:401–424.
- [8] F. Wang, B. Lazarov, O. Sigmund and J. Jensen *Interpolation scheme for fictitious domain techniques and topology optimization of finite strain elastic problems*, Computer Methods in Applied Mechanics and Engineering, 2014, 276:453–424
- [9] O. Amir and B. Lazarov *Achieving stress-constrained topological design via length scale control*, Structural and Multidisciplinary Optimization, 2018, 58:2053–2071
- [10] E. Holberg, B. Torstenfelt and A. Klarbring *Stress constrained topology optimization.*, Structural and Multidisciplinary Optimization, 2013, 48:33-47
- [11] F. Wang, B. Lazarov and O. Sigmund *On projection methods, convergence and robust formulations in topology optimization.*, Structural and Multidisciplinary Optimization, 2011, 43:767-784
- [12] M. Wallin, N. Ivarsson, O. Amir, D. Tortorelli *Consistent boundary conditions for PDE filter regularization in topology optimization*, Structural and Multidisciplinary Optimization, 2020, 62:1299–1311

- [13] J.K. Guest, J.H. Prévost and T. Belytschko *Achieving minimum length scale in topology optimization using nodal design variables and projection functions*, Numerical Methods in Engineering, 2004, 61(2):238-254
- [14] N. Pollini and O. Amir *Mixed projection- and density-based topology optimization with applications to structural assemblies*, Structural and Multidisciplinary Optimization, 2020, 61:687-710

Appendix

A

Compliance minimization plots with a volume constraint and dynamic length scale

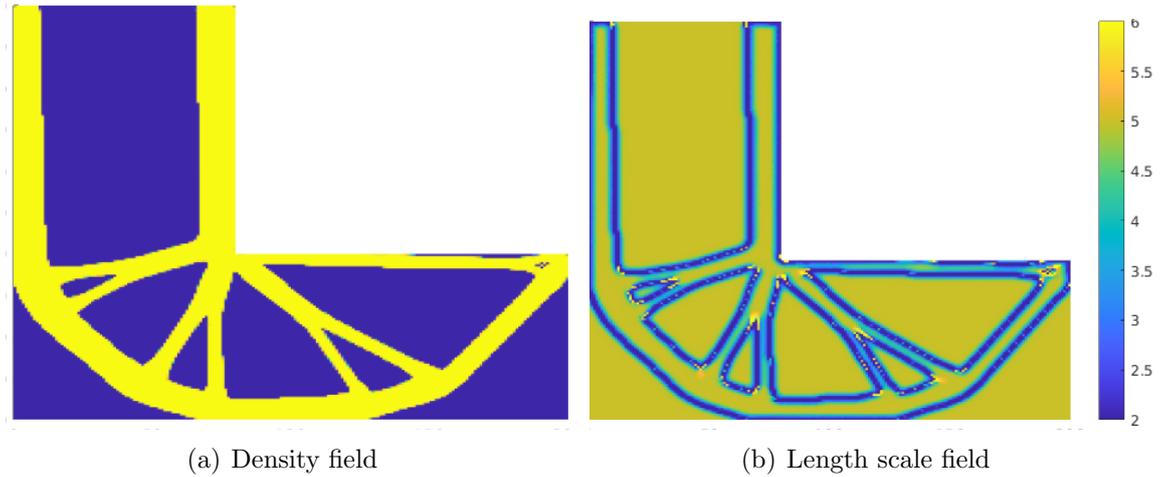


Figure 46: Optimization of \mathbb{P}_1 with $\gamma = 0.0001$, $C = 7868$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

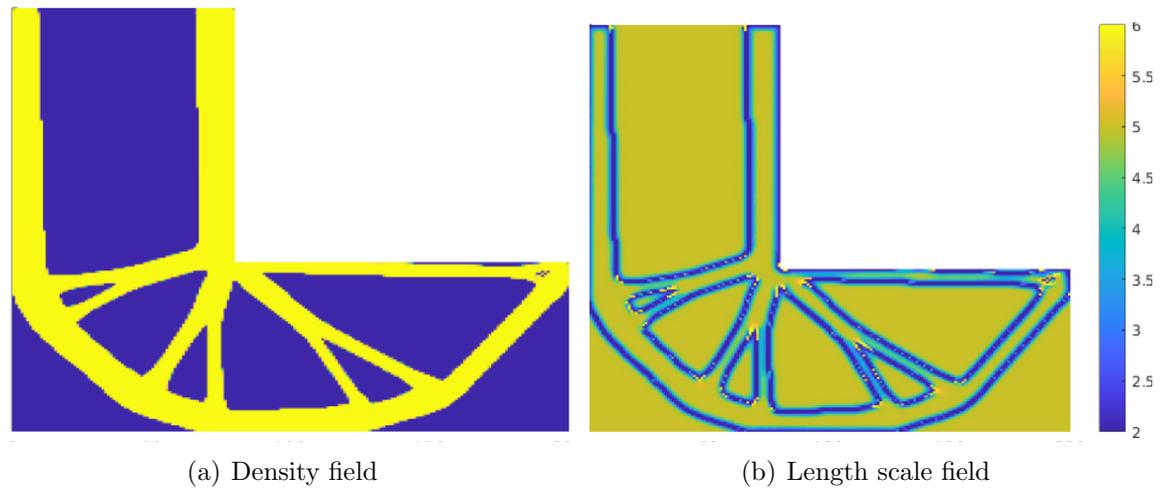


Figure 47: Optimization of \mathbb{P}_1 with $\gamma = 0.0002$, $C = 7868$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

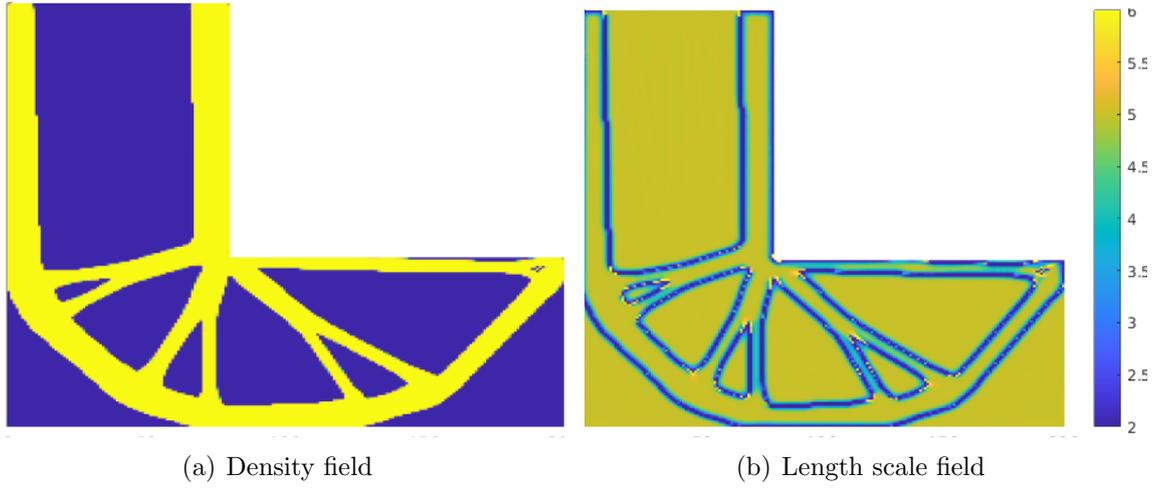


Figure 48: Optimization of \mathbb{P}_1 with $\gamma = 0.0005$, $C = 7867$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

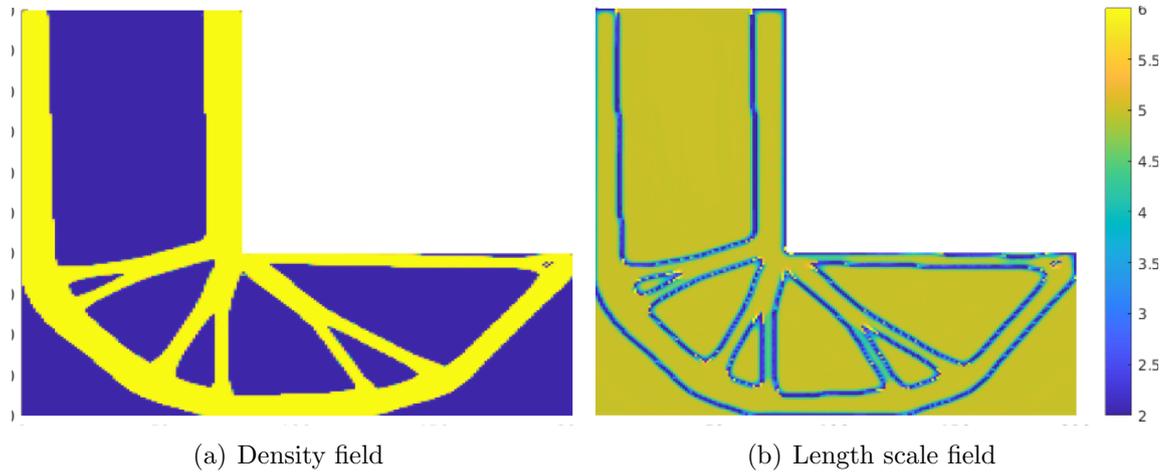


Figure 49: Optimization of \mathbb{P}_1 with $\gamma = 0.002$, $C = 7873$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

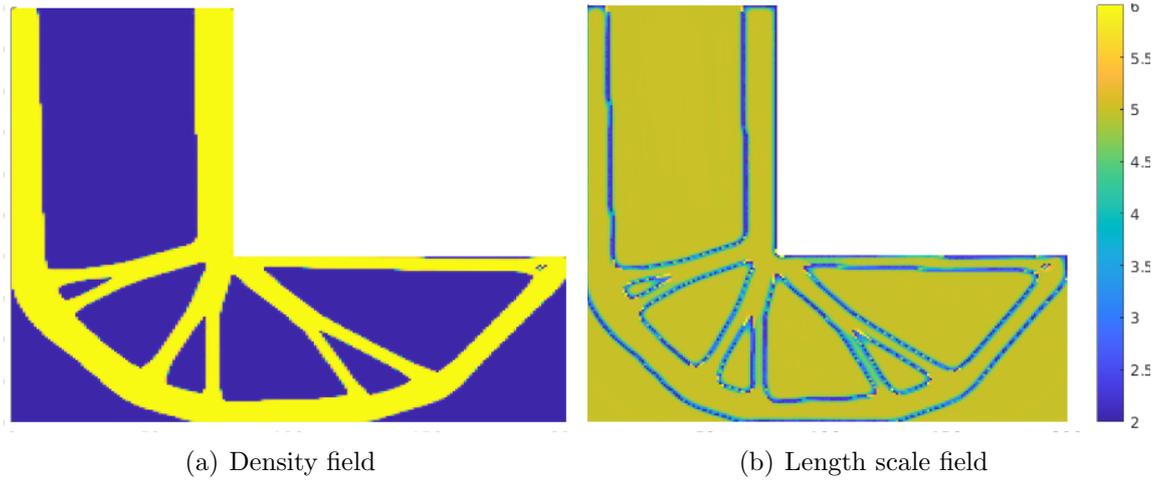


Figure 50: Optimization of \mathbb{P}_1 with $\gamma = 0.005$, $C = 7893$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

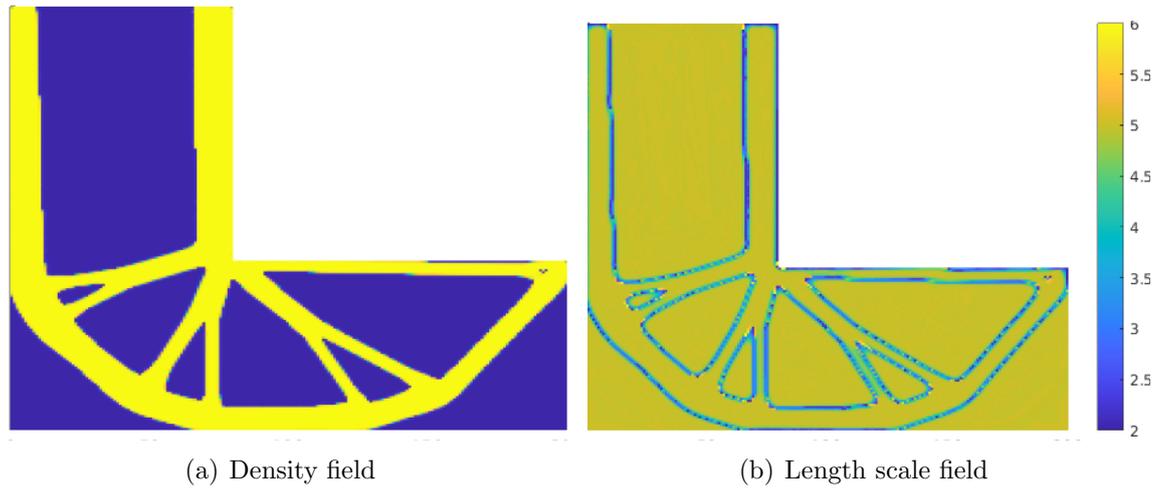


Figure 51: Optimization of \mathbb{P}_1 with $\gamma = 0.008$, $C = 7922$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

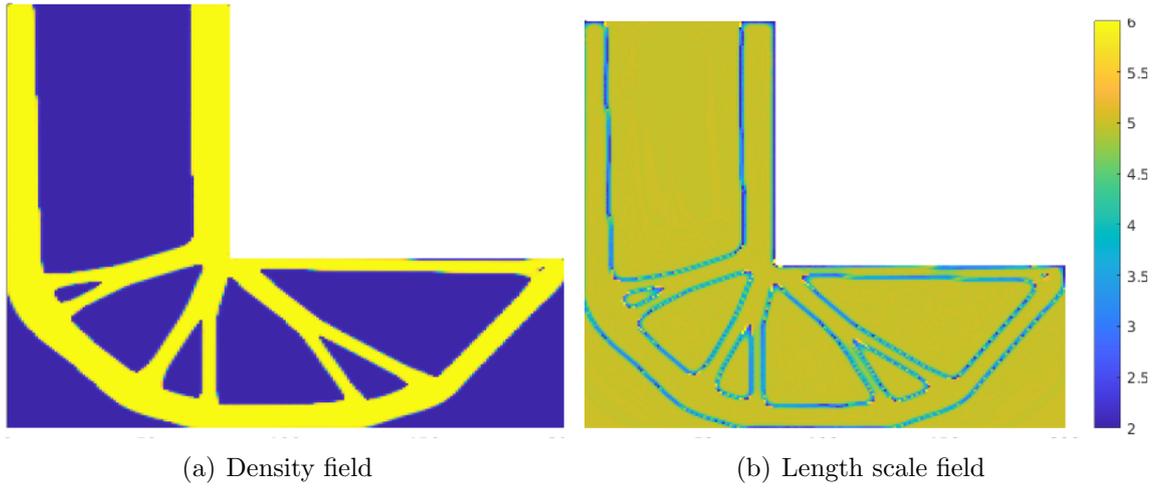


Figure 52: Optimization of \mathbb{P}_1 with $\gamma = 0.01$, $C = 7935$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

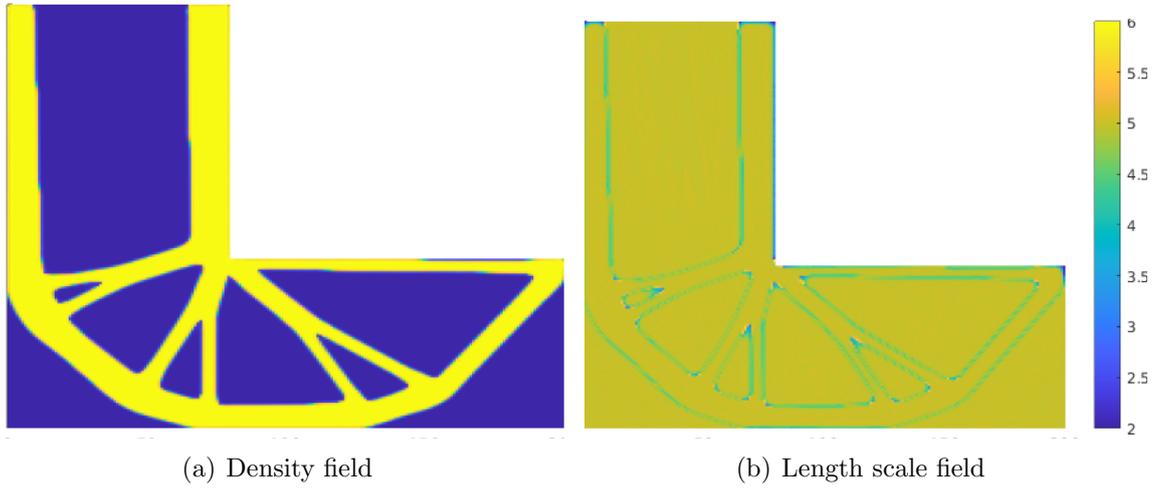


Figure 53: Optimization of \mathbb{P}_1 with $\gamma = 0.05$, $C = 8072$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

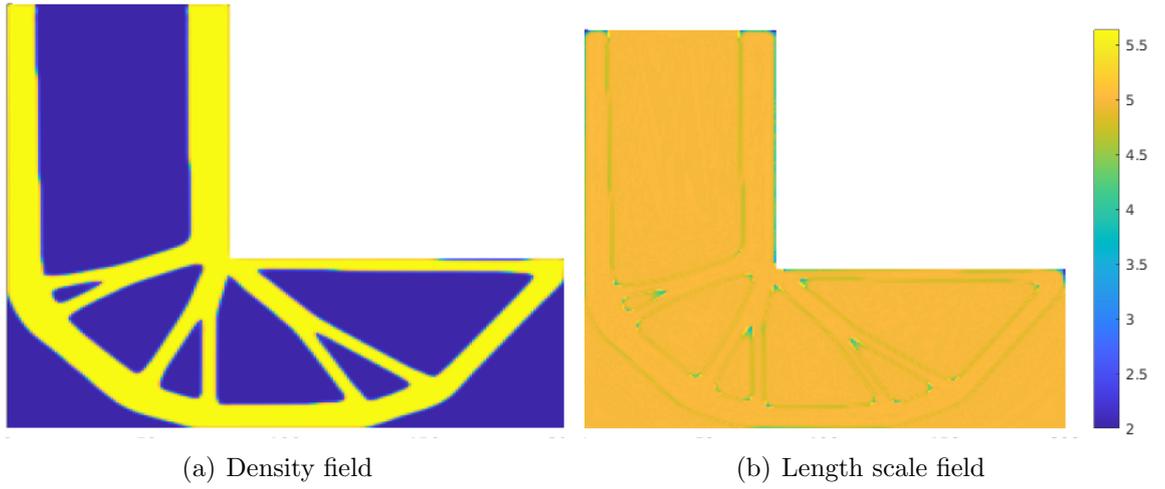


Figure 54: Optimization of \mathbb{P}_1 with $\gamma = 0.1$, $C = 8119$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

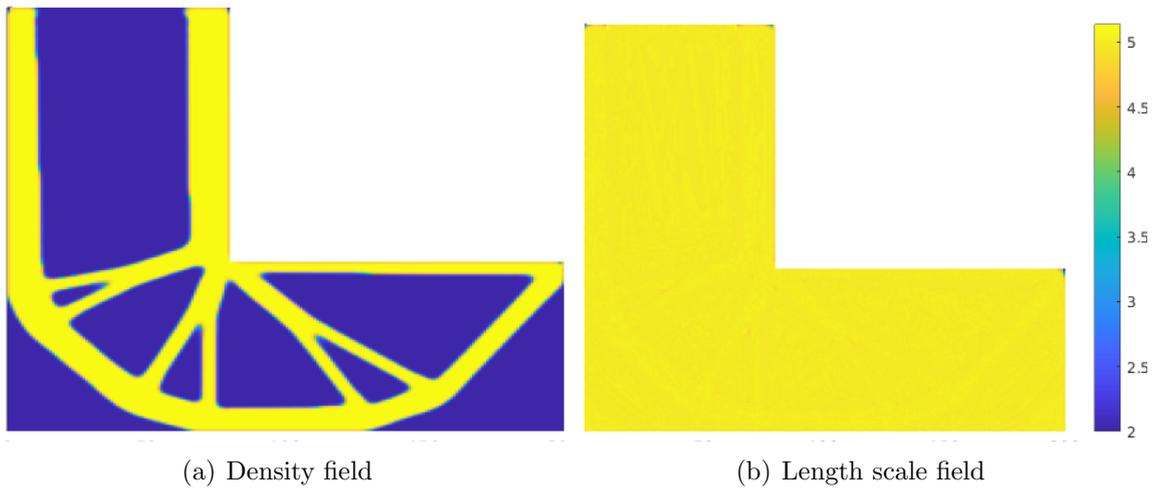


Figure 55: Optimization of \mathbb{P}_1 with $\gamma = 1$, $C = 8192$, $r_c = 5$, $r_{max} = 6$, $r_{min} = 2$

B

Compliance minimization plots with a volume constraint and a stress constraint and a dynamic length scale.

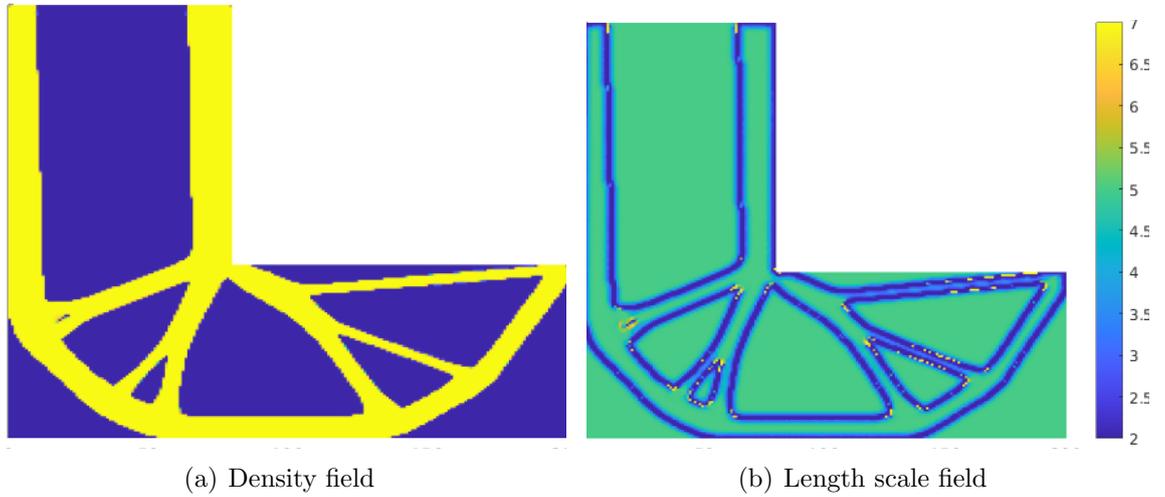


Figure 56: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8018$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 849$, $\sigma_{max} = 850$, $r_{mean} = 3.9103$

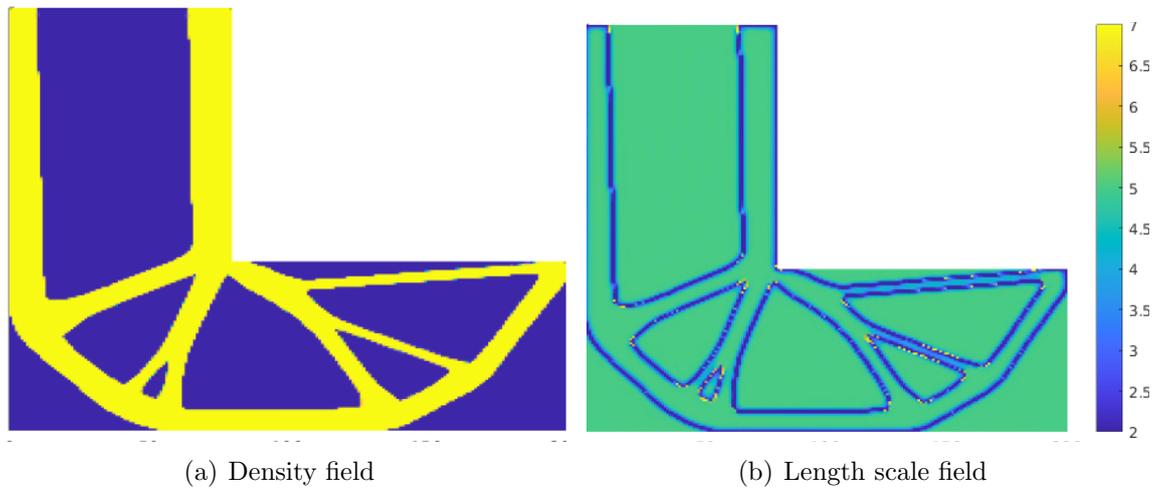


Figure 57: Optimization of \mathbb{P}_2 with $\gamma = 0.001$, $C = 8001$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 849$, $\sigma_{max} = 850$, $r_{mean} = 4.2804$

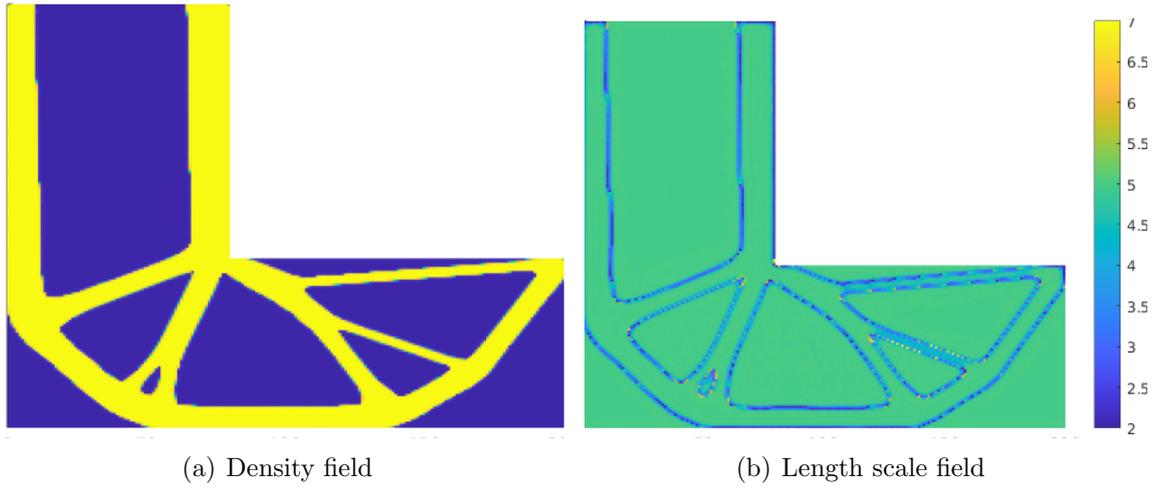


Figure 58: Optimization of \mathbb{P}_2 with $\gamma = 0.01$, $C = 8095$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 849$, $\sigma_{max} = 850$, $r_{mean} = 4.6518$

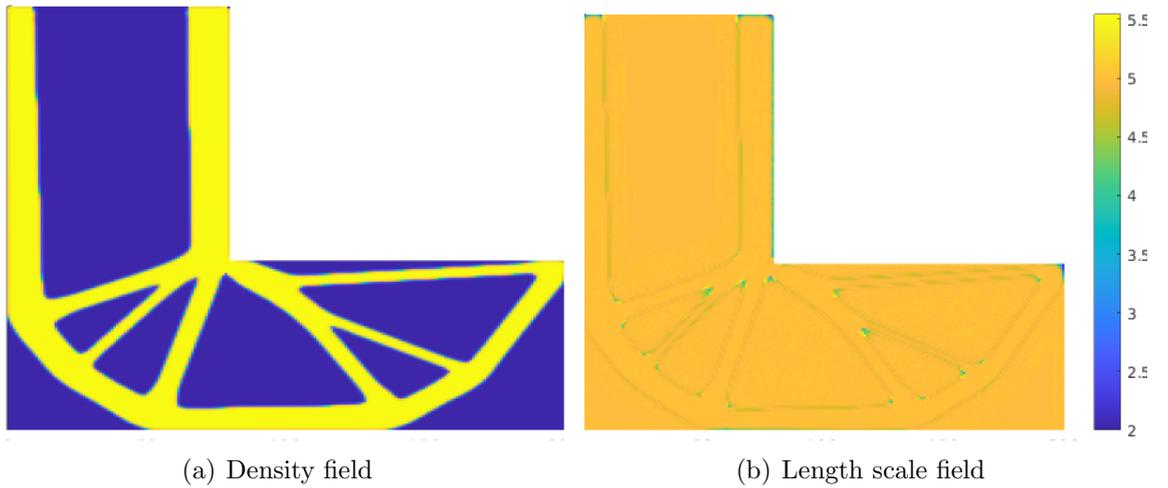


Figure 59: Optimization of \mathbb{P}_2 with $\gamma = 0.1$, $C = 8268$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 849$, $\sigma_{max} = 850$, $r_{mean} = 4.9483$

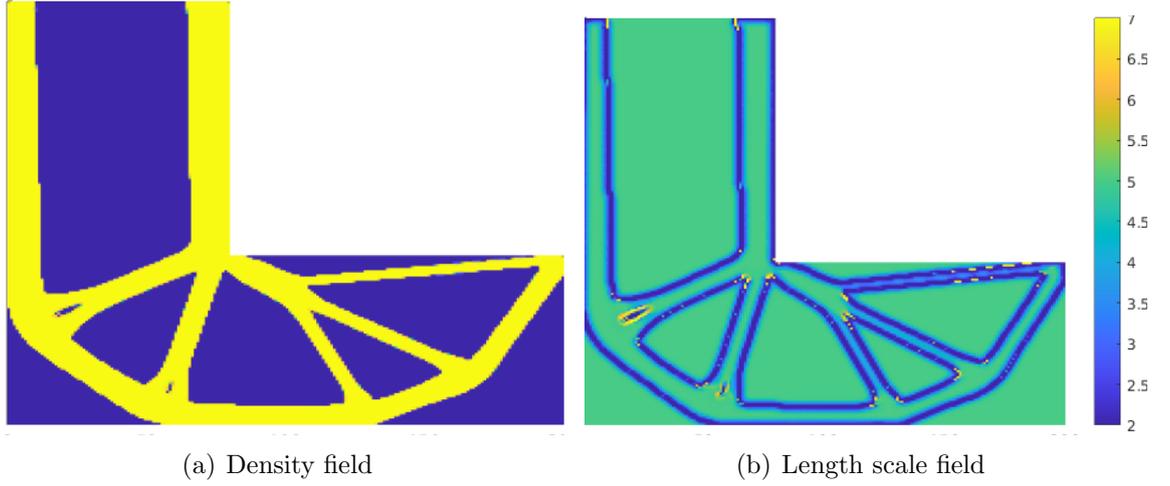


Figure 60: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8015$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 799$, $\sigma_{max} = 800$, $r_{mean} = 3.9275$

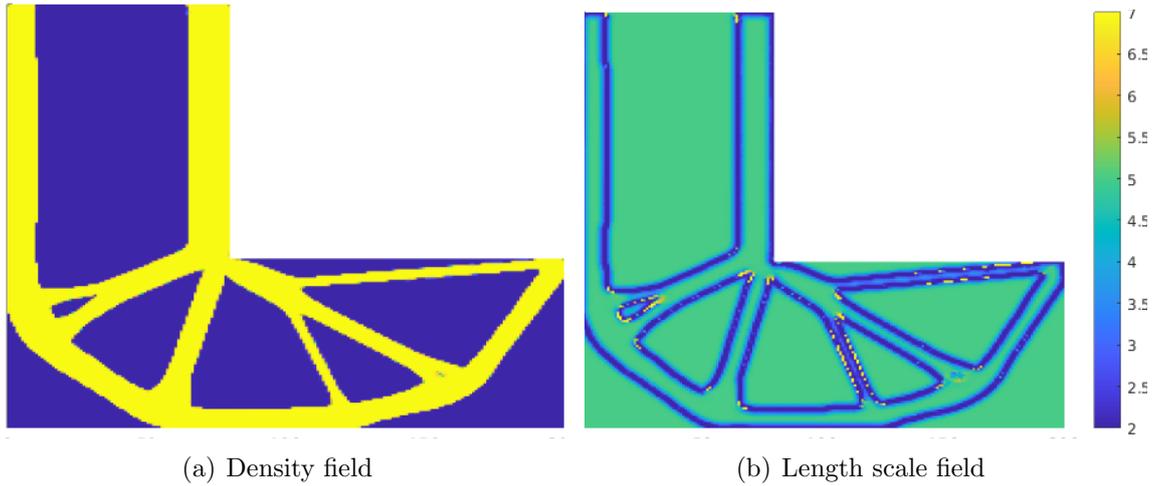


Figure 61: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8028$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 749$, $\sigma_{max} = 750$, $r_{mean} = 3.9157$

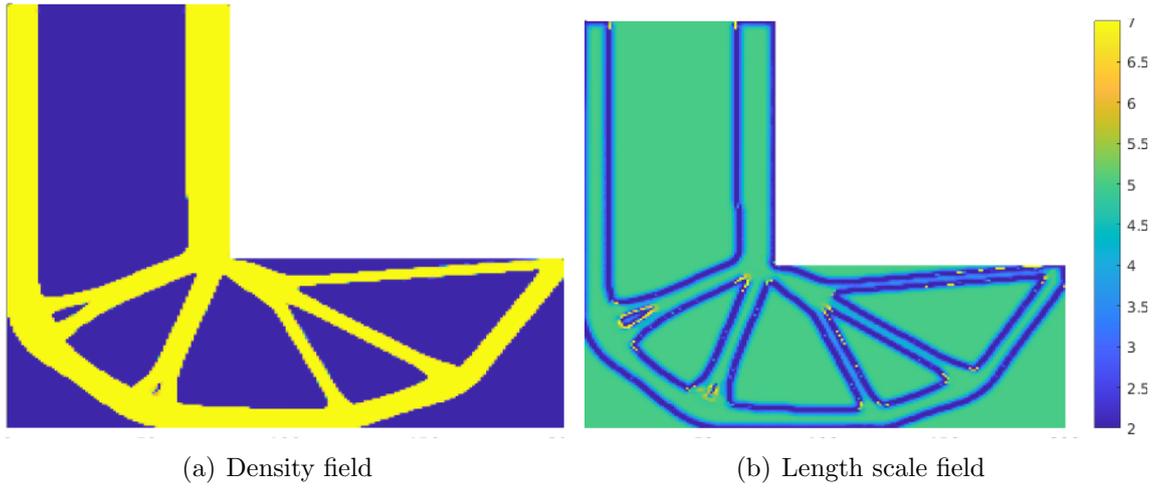


Figure 62: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8019$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 650$, $\sigma_{max} = 650$, $r_{mean} = 3.8980$

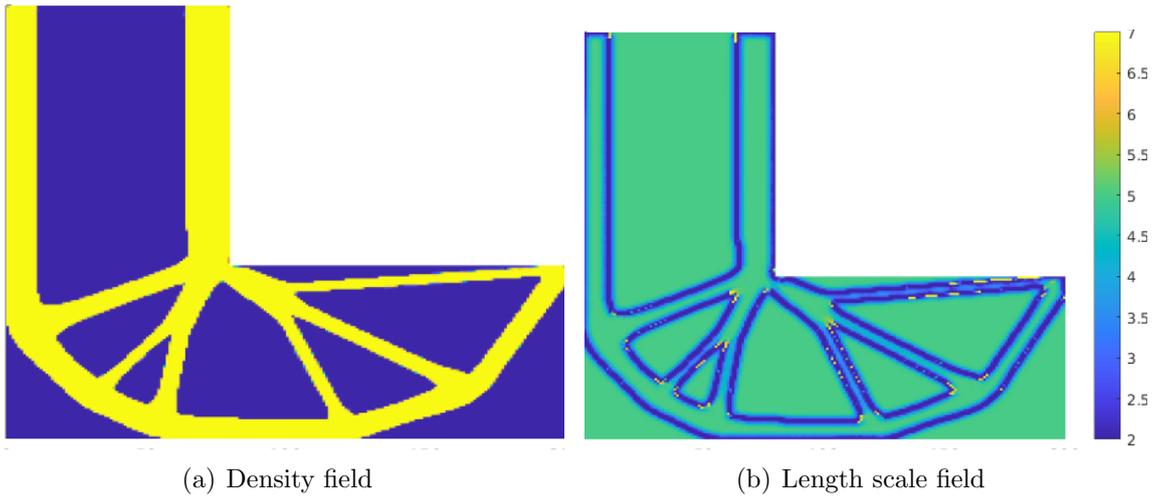


Figure 63: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8047$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 600$, $\sigma_{max} = 600$, $r_{mean} = 3.8719$

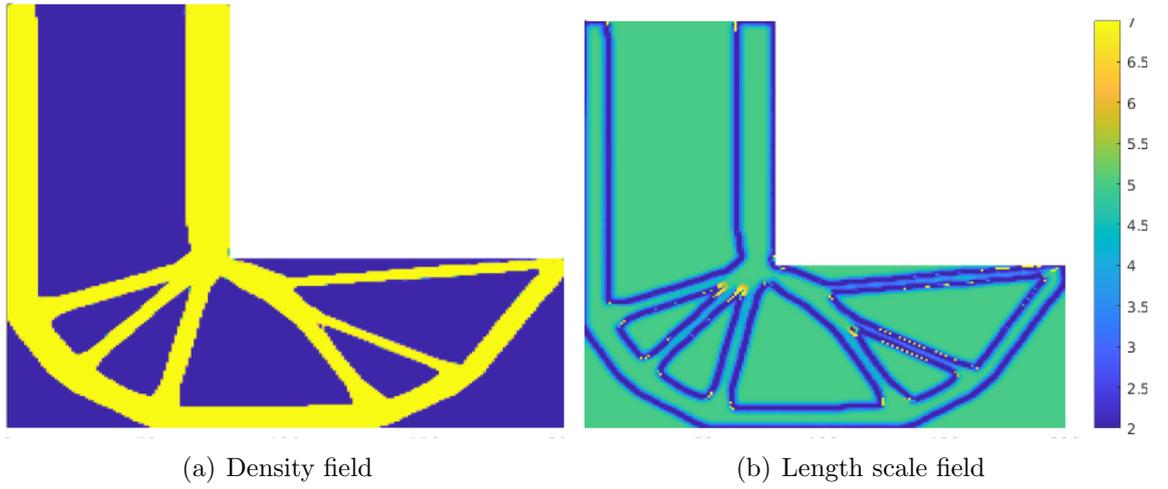


Figure 64: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 8103$, $r_c = 5$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 499$, $\sigma_{max} = 500$, $r_{mean} = 3.8552$

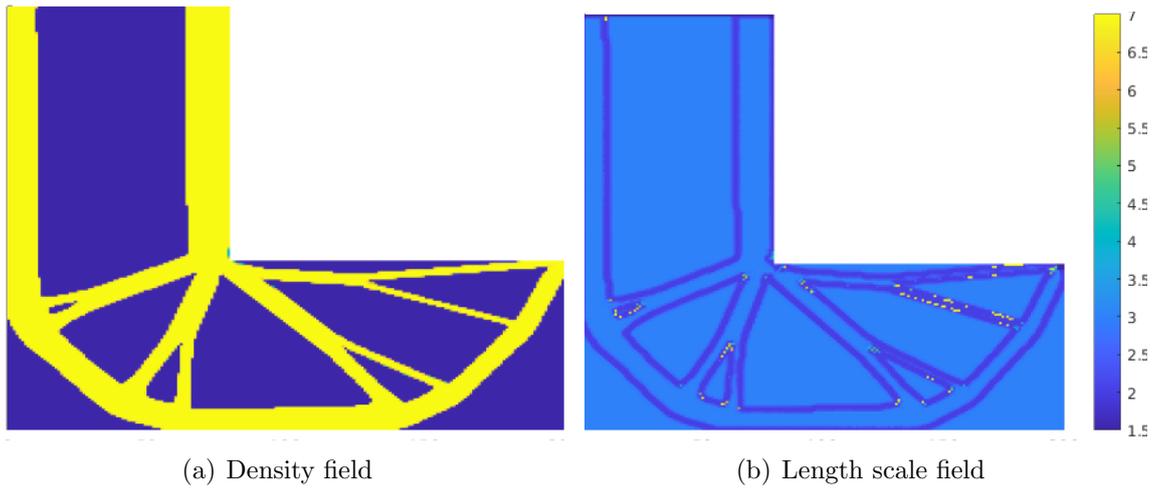


Figure 65: Optimization of \mathbb{P}_2 with $\gamma = 0.0001$, $C = 7979$, $r_c = 3$, $r_{max} = 7$, $r_{min} = 2$, $\sigma^{PN} = 499$, $\sigma_{max} = 500$, $r_{mean} = 2.6619$