Chapter 1

Introduction

Topology Optimization of Elastic Media is a technique used to optimize a structure that is bearing some load. Ideally, we would like to minimize the maximum stress placed on a structure by selecting a region E where material is placed. In other words,

$$\text{Minimize} \|\sigma(u)\|_\infty$$

$$\text{Subject to} \|E\| \leq V_{max}$$

$$\text{And } \nabla \cdot \sigma + F = 0$$

However, the infinity norm creates a problem in that the maximum strain over the domain as a function of location of material is necessarily not everywhere differentiable, making prospects of optimization rather bleak. So instead, we find an approximate solution by optimizing for the strain energy, or compliance. This is a measure of the potential energy stored in an object due to its deformation, but also works as a measure of total displacement over the structure.

$$\text{Minimize } \int_\Omega \frac{1}{2}\sigma : \epsilon \, d\Omega$$

$$\text{Subject to } \|E\| \leq V_{max}$$

$$\text{And } \nabla \cdot \sigma + F = 0$$

The value of the objective function is calculated using a finite element method, where the solution is the displacements. This is placed inside of a nonlinear solver loop that solves for a vector denoting placement of material. If we stick within real-world confines, and allow the material to either be present, or not be present, then this optimization problem becomes combinatorial, and very expensive to solve. Instead, we use an approach called Solid Isotropic Material with Penalization, or SIMP.

1.1 Solid Isotropic Material with Penalization

The Solid Isotropic Material with Penalization, or SIMP, method is based off of an idea of allowing the material to exist in a location with a density between 0 and 1. A density of 0 suggests the material is not there, and it is not a part of the structure, while a density of 1 suggests the material is present. Values between 0 and 1 do not reflect real-world phenomena, but allows us to turn the combinatorial problem into a continuous one. I then look at density values ρ, with the constraint that $0 < \rho_{min} \leq \rho \leq 1$. The minimum value ρ_{min}, typically chosen to be around 10^{-3}, keeps me from the possibility of having an infinite compliance, but is small enough to provide accurate results.

The straightforward application of the effect of this "density" on the elasticity of the media would be to simply multiply the stiffness tensor C of the media by the given density, that is, the $C = \rho C_0$. However, this approach often gives optimal solutions where density values are far from both 0 and 1. As I want to find a real-world solution, meaning the material either is present or it is not, a penalty is applied to these in-between values. A simple and effect way to do this is to multiply the stiffness tensor by the density raised to some integer power penalty parameter p, so that $C = \rho^p C_0$. This makes density values farther away from 0 or 1 less effective. It has been shown that using $p = 3$ is sufficiently high to create 'black-and-white' solutions.

Using this density idea also allows me to reframe the volume constraint on the optimization problem. Use of SIMP then turns the optimization problem into the following:

$$\text{Minimize } \int_\Omega \frac{1}{2}\sigma(\rho) : \epsilon(\rho) \, d\Omega$$

$$\text{Subject to } \int_\Omega \rho(x) \, d\Omega \leq V_{max},$$

$$0 < \rho_{min} \leq \rho(x) \leq 1,$$

$$\text{And } \nabla \cdot \sigma(\rho) + F = 0$$

The final constraint, the elasticity equation, gives a method for finding σ and ϵ given the density ρ.

1.2 Elasticity Equation

The linear elasticity equation is formulated as shown in step 8:

$$\nabla \cdot \sigma + F = \rho \frac{\partial^2 u}{\partial t^2}$$

1.3 Making the solution mesh-independent

Typically, the solutions to topology optimization problems are not mesh-independent, and as such the problem is ill-posed. This is because as the mesh is refined further, fractal structures are often formed. As the mesh gains resolution, the optimal solution typically gains smaller and smaller structures. There are a few competing work-arounds to this issue, but the most popular for first order optimization is the sensitivity filter, while second order optimization methods tend to prefer use of a density filter.

As the filters effect the gradient and hessian of the compliance, the choice of filter has an effect on the solution of the problem. The density filter as part of a second order method, works by introducing an unfiltered density, which I refer to as σ, and then requiring that the density be a convolution of the unfiltered density. This prevents checkerboarding, but also the radius of the filter allows the user to define an effective minimal beam width.

1.4 Complete Problem Formulation

The minimization problem is now

$$\min_{\rho,\sigma,u} \int_\Omega u \cdot f$$

$$\text{s.t. } \rho = H(\sigma)$$

$$\int_\Omega \rho^p \left(\frac{\mu}{2}\left(\epsilon(v) : \epsilon(u)\right)\right) + \lambda\left(\nabla \cdot u \nabla \cdot v\right)\Big) = \int_\Gamma v \cdot f$$

$$0 \leq \sigma \leq 1$$

Using slack variables for the inequalities, the KKT conditions give the requirements listed below. In this formulation, $d_{\{\cdot\}}$ is a test function that is naturally paired with the $\{\cdot\}$ function.

1. Stationarity

$$\int_\Omega -d_\rho y_2 + p\rho^{p-1} d_\rho(\lambda \nabla \cdot y_1 \nabla \cdot u + \mu\varepsilon(u)\varepsilon(y_1)) = 0$$

$$\int_\Gamma d_u \cdot t + \int_\Omega p\rho^p(\lambda \nabla \cdot y \nabla \cdot d_u + \mu\varepsilon(d_u)\varepsilon(y_1)) = 0$$

$$\int_\Omega -d_\sigma z_1 + d_\sigma z_2 + H(d_\sigma)y_2 = 0$$

2. Primal Feasibility

$$\int_\Omega \rho^p \lambda \nabla \cdot d_y \nabla \cdot u + \rho^p \mu\varepsilon(u)\varepsilon(d_y) - \int_\Omega F \cdot d_y - \int_\Gamma t \cdot d_y = 0$$

$$\int_\Omega (\sigma - s_1) d_{z_1} = 0$$

$$\int_\Omega (1 - \sigma - s_2) d_{z_2} = 0$$

$$\int_\Omega (\rho - H(\sigma)) d_{y_2} = 0$$

3. Complementary Slackness

$$\int_\Omega (s_1 z_1 - \alpha) d_{s_1} = 0, \quad \alpha \to 0$$

$$\int_\Omega (s_2 z_2 - \alpha) d_{s_2} = 0, \quad \alpha \to 0$$

Dual Feasibility

$$s, z \geq 0$$

Using Newton's method to find the solution that gives 0s gives

1. Stationarity - these equations ensure we are at a critical point of the objective function when constrained

   Equation 0

$$\int_\Omega -d_\rho c_{y_2} + p(p-1)\rho^{p-2} d_\rho c_\rho(\lambda \nabla \cdot y_1 \nabla \cdot u + \mu\varepsilon(u)\varepsilon(y_1)) +$$

$$p\rho^{p-1}(d_\rho \lambda \nabla \cdot c_{y_1} \nabla \cdot u + d_\rho \mu\varepsilon(u)\varepsilon(c_{y_1}) + d_\rho \lambda \nabla \cdot y_1 \nabla \cdot c_u + d_\rho \mu\varepsilon(c_u)\varepsilon(y_1))$$

$$= -\int_\Omega -d_\rho z_1 + d_\rho z_2 - d_\rho y_2 + p\rho^{p-1} d_\rho(\lambda \nabla \cdot y_1 \nabla \cdot u + \mu\varepsilon(u)\varepsilon(y_1))$$

   Equation 1

$$\int_\Omega p\rho^{p-1} c_\rho(\lambda \nabla \cdot y_1 \nabla \cdot d_u + \mu\varepsilon(d_u)\varepsilon(y)) + \rho^p(\lambda \nabla \cdot c_{y_1} \nabla \cdot d_u + \mu\varepsilon(d_u)\varepsilon(c_{y_1}))$$

$$= -\int_\Gamma d_u \cdot t - \int_\Omega \rho^p(\lambda \nabla \cdot y \nabla \cdot d_u + \mu\varepsilon(d_u)\varepsilon(y_1))$$

   Equation 2

$$\int_\Omega -d_\sigma c_{z_1} + d_\sigma c_{z_2} + H(d_\sigma) c_{y_2} = -\int_\Omega -d_\sigma z_1 + d_\sigma z_2 + H(d_\sigma) y_2$$

2. Primal Feasibility - these equations ensure the equality constraints are met

   Equation 3

$$\int_\Omega p\rho^{p-1} c_p(\lambda \nabla \cdot d_{y_1} \nabla \cdot u + \mu\varepsilon(u)\varepsilon(d_{y_1})) + \rho^p(\lambda \nabla \cdot d_{y_1} \nabla \cdot c_u + \mu\varepsilon(c_u)\varepsilon(d_{y_1}))$$

$$= -\int_\Omega \rho^p \lambda \nabla \cdot d_{y_1} \nabla \cdot u + \rho^p \mu\varepsilon(u)\varepsilon(d_{y_1}) + \int_\Gamma t \cdot d_{y_1}$$

   Equation 4

$$-\int_\Omega (c_\sigma - c_{s_1}) d_{z_1} = \int_\Omega (\sigma - s_1) d_{z_1}$$

   Equation 5

$$-\int_\Omega (-c_\sigma - c_{s_2}) d_{z_2} = \int_\Omega (1 - \sigma - s_2) d_{z_2}$$

   Equation 6

$$-\int_\Omega (c_\rho - H(c_\sigma)) d_{y_2} = \int_\Omega (\rho - H(\sigma)) d_{y_2}$$

3. Complementary Slackness - these equations essentially ensure the barrier is met - in the actual solution, we need $s^T z = 0$

   Equation 7

$$\int_\Omega (c_{s_1} z_1 / s_1 + c_{z_1}) d_{s_1} = -\int_\Omega (z_1 - \alpha/s_1) d_{s_1}, \quad \alpha \to 0$$

   Equation 8

$$\int_\Omega (c_{s_2} z_2 / s_2 + c_{z_2}) d_{s_2} = -\int_\Omega (z_2 - \alpha/s_2) d_{s_2}, \quad \alpha \to 0$$

   Dual Feasibility - Multiplier on slacks and slack variables must be kept greater than 0. (This is the only part not implemented in the "assemble_block_system()" function)

$$s, z \geq 0$$

1.5 Discretization

I use a quadrilateral mesh with $Q1$ element to discretize the displacement and displacement lagrange multiplier. Piecewise constant $Q0$ elements are used to discretize the density, unfiltered density, density slack variables, and multipliers for the slack variables and filter constraint.

1.6 Nonlinear Algorithm

As the problem posed is non-convex, I implement a watchdog-search algorithm as follows:

Set barrier value, β
Set descent requirement, ν
Set initial guess x_0
while *Barrier above minimal value* **do**
  while *Convergence not reached* **do**
    for $i = 0$ *to* $\hat{t}$ *- typically 5 or 8* **do**
      Compute step p_{k+i} with Newton's Method
      Compute $x_{k+i+1} = x_{k+i} + p_{k+i}$
      if *New state's merit is lower than watchdog merit* **then**
        Make current step watchdog step
      end
      if $\phi(x_{k+i+1}) < \phi(x_k) + \nu D(\phi(x_k), p_k)$ *-* ϕ *is merit function* **then**
        Accept x_{k+i+1}
        $k = k + i + 1$
        Found Step = True
        Break for loop
      end
    end
    if *Found Step = False* **then**
      Compute step $p_{k+\hat{t}+1}$ with Newton's Method
      Find $\alpha_{k+\hat{t}+1}$ so that $\phi(x_{k+\hat{t}+2}) \leq \phi(k+\hat{t}+1) + \nu\alpha_{k+\hat{t}+1} D(\phi(x_{k+\hat{t}+1}); p_{k+\hat{t}+1})$
      $x_{k+\hat{t}+2} = x_{k+\hat{t}+1} + \alpha_{k+\hat{t}+1} p_{k+\hat{t}+1}$
      if $\phi(x_{k+\hat{t}+1}) \leq \phi(x_k)$ ***or*** $\phi(x_{k+\hat{t}+2}) \leq \phi x_k + \nu D(\phi(x_k); p_k)$ **then**
        Accept $x_{k+\hat{t}+2}$
        $k = k + \hat{t} + 2$
      else
        if $\phi(x_{k+\hat{t}+2}) > \phi(x_k)$ **then**
          Find α_k such that $\phi(x_{k+\hat{t}+3}) \leq \phi(x_k) + \nu\alpha_k D(\phi(x_k); p_k)$
          $x_{k+\hat{t}+3} = x_k + \alpha_k p_k$ Accept $x_{k+\hat{t}+3}$
          $k = k + \hat{t} + 3$
        else
          Compute $p_{k+\hat{t}+2}$
          Find $\alpha_{k+\hat{t}+2}$ such that
          $\phi(x_{k+3}) \leq \phi(x_{k+\hat{t}+2}) + \nu\alpha_{k+\hat{t}+2} D(\phi(x_{k+\hat{t}+2}); p_{k+\hat{t}+2})$
          $x_{k+\hat{t}+3} = x_{k+\hat{t}+2} + \alpha_{k+\hat{t}+2} p_{k+\hat{t}+2}$
          Accept $x_{k+\hat{t}+3}$
          $k = k + \hat{t} + 3$
        end
      end
    end
  end
  Reduce Barrier Size
end

The barrier is reduced by a method similar to that used in IPOPT - at larger values, it is multiplied by a constant (around .6), and at lower values the barrier value is replaced by the barrier value raised to some exponent (around 1.2). While taking large steps at reducing the barrier size when convergence is reached is largely used, it is typically faster to use algorithms that adaptively update barrier each iteration.

1.7 Merit Function

I use an exact l1 merit function in this problem to test the steps. This is partially due to already having most of the necessary parts calculated in finding the right hand side, but also the use of an exact merit function ensures that it is minimized in the same location as the overall problem. In the future, it should be considered to use a filter method instead as they prove tot be more efficient.

1.8 Convergence check

I use the l_∞ norm of the KKT conditions to check for convergence at each barrier size. The requirement is that $\|KKT\|_{l_\infty} < c \cdot \beta$ where c is a constant over any barrier size and β is the barrier size. This forces better convergence in later iterations, and is the same requirement as is used in IPOPT.

Chapter 2

Commented Code

Chapter 3

Results

3.1 Test Problem

The algorithms I explored this semester are tested against a traditional topology optimization problem called the MBB Beam. This problem considers the optimal 2-d structure that can be built on a rectangle 6 units wide, and 1 unit tall. The bottom corners are fixed in place in the y direction using a 0 Dirichlet boundary conditions, and a downward force is applied in the center of the top of the beam by enforcing a Neumann boundary condition. The rest of the boundary is allowed to move, and has no external force applied, which takes the form of a 0 Neumann boundary condition. While the total volume of the domain is 6, 3 units of material are allowed for the structure. Because of the symmetry of the problem, it can be posed on a rectangle of width 3 and height 1 by cutting the original domain in half, and using 0 dirichlet boundary conditions in the x direction along the cut edge, as shown in figure 3.1. That said, symmetry was nice to look for in debugging, so I solved the problem on the whole domain.

The-MBB-beam-problem-geometry.png

Figure 3.1: The MBB problem domain and boundary conditions

The following solutions to the MBB Beam have been found using this code.

visit0000.png

Figure 3.2: Filtered density

visit0001.png

Figure 3.3: Unfiltered Density

These results took over 100 iterations to find, which is quite concerning. Looking at the evolution, it does look as though the convergence has moments of happening quickly and moments of happening slowly. I believe this to be due to both a lack of precision on when and how to decrease the boundary values, as well as a less-than optimized merit function not allowing me to find optimal step sizes.

The barrier decrease is most sensitive in the middle of the convergence, which is problematic, as it seems like I need it to decrease quickly, then slowly, then quickly again.