MDO_ML_21

Multidisciplinary Optimization and Machine Learning for Engineering Design

19 July 2021 – 5 August 2021

https://mdoml2021.ftmd.itb.ac.id/

Jointly organized by

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Design for Additive Manufacturing: Topology Optimization

Prof. Joseph Morlier

Multidisciplinary Optimization and Machine Learning for Engineering Design

19 July 2021 – 5 August 2021

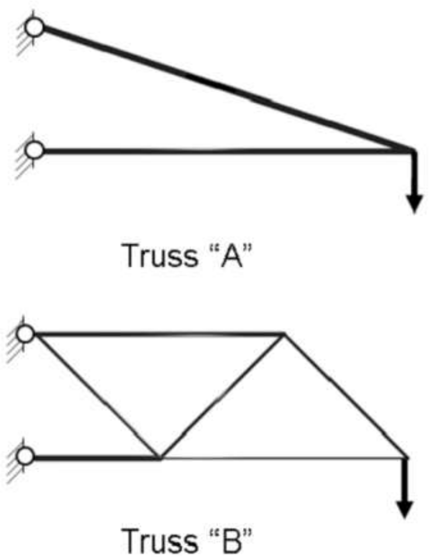
https://mdoml2021.ftmd.itb.ac.id/

Jointly organized by

Part 2

Topology Optimization

Topology is the Key to Sustainability?



Strength $\dfrac{V_A}{V_B} =$ 27% More

Deflection $\dfrac{V_A}{V_B} =$ 60% More

History

- Homogenization of Microstructures was introduced by mathematics in the 1970s.
- First paper by Martin Bendsoe (Technical University of Denmark) and Noboru Kikuchi (University of Michigan) in 1988

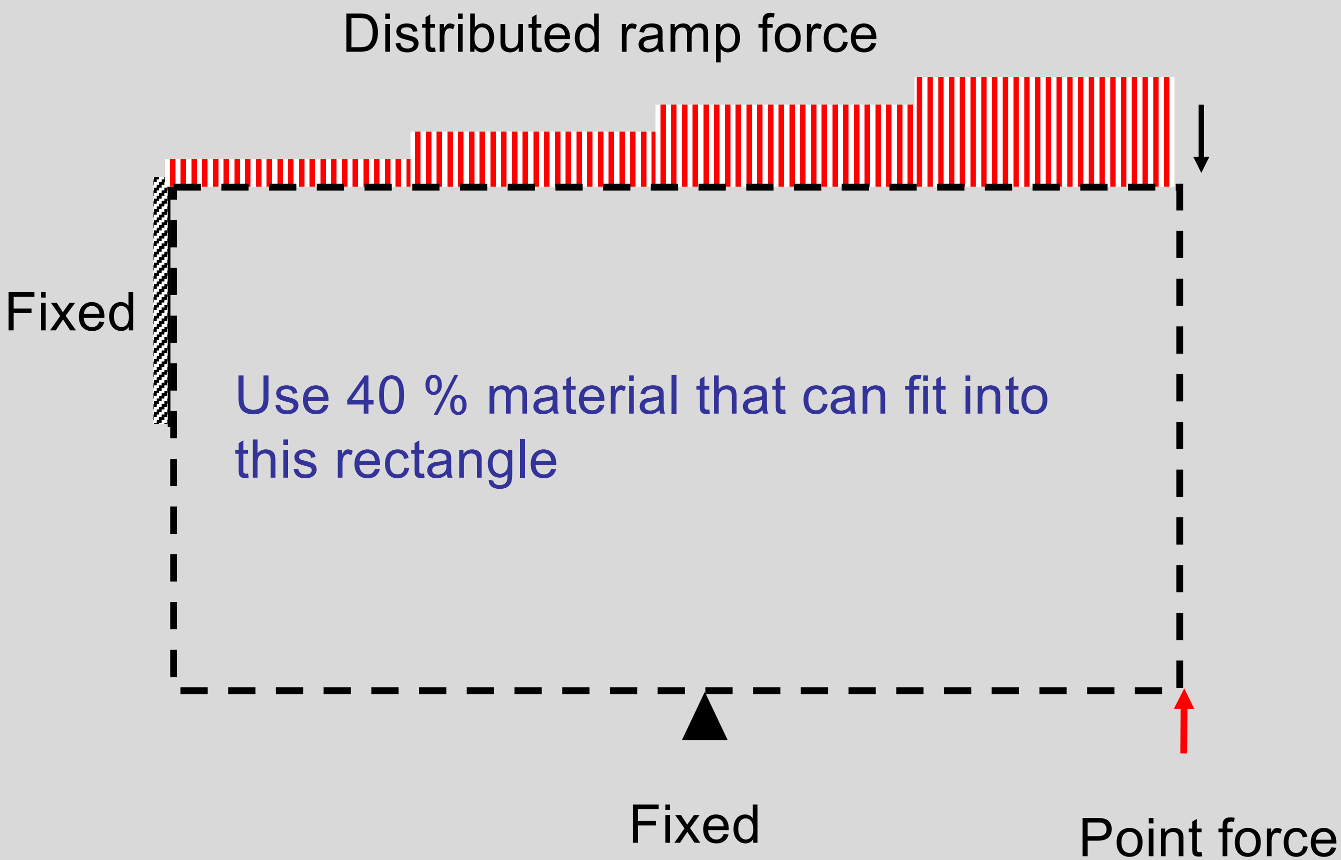
A topology optimisation problem can be written in the general form of an optimization problem as

$$\min_{\rho} \; F = F(\mathbf{u}(\rho), \rho) = \int_{\Omega} f(\mathbf{u}(\rho), \rho) \mathrm{d}V$$

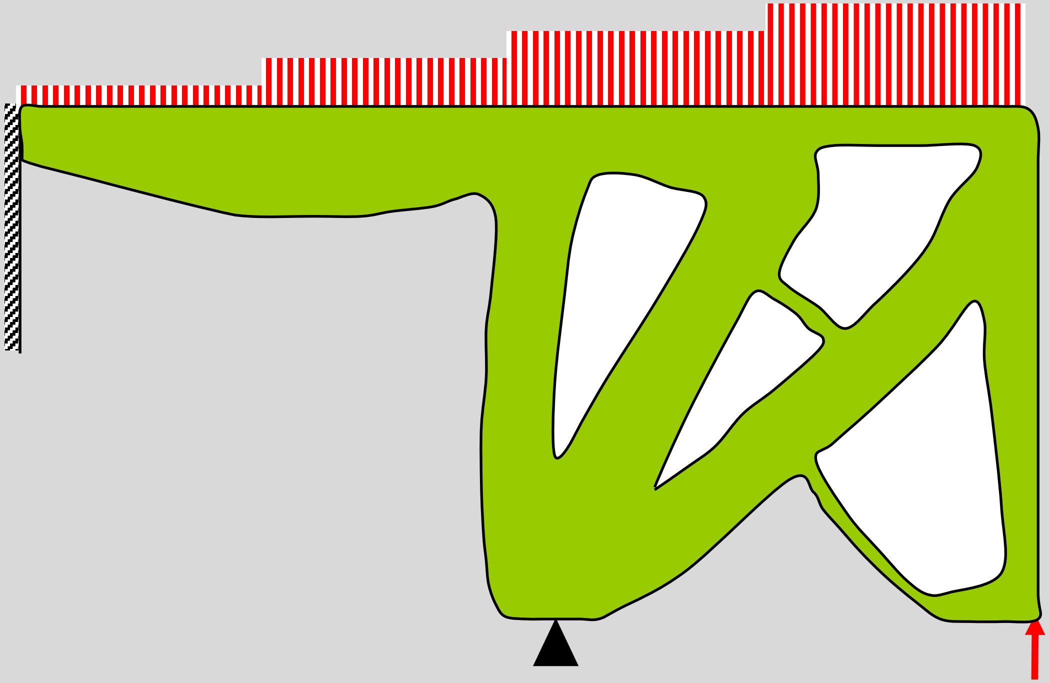
subject to

- $\rho \in \{0, 1\}$
- $G_0(\rho) = \int_{\Omega} \rho(\mathbf{u}) \mathrm{d}V - V_0 \leq 0$
- $G_j(\mathbf{u}(\rho), \rho) \leq 0$ with $j = 1, \ldots, m$

Stiff structure for your specifications



Stiff structure for your specifications



Well-Known example

- Wing stiffening ribs of Airbus A380:

AIRBUS

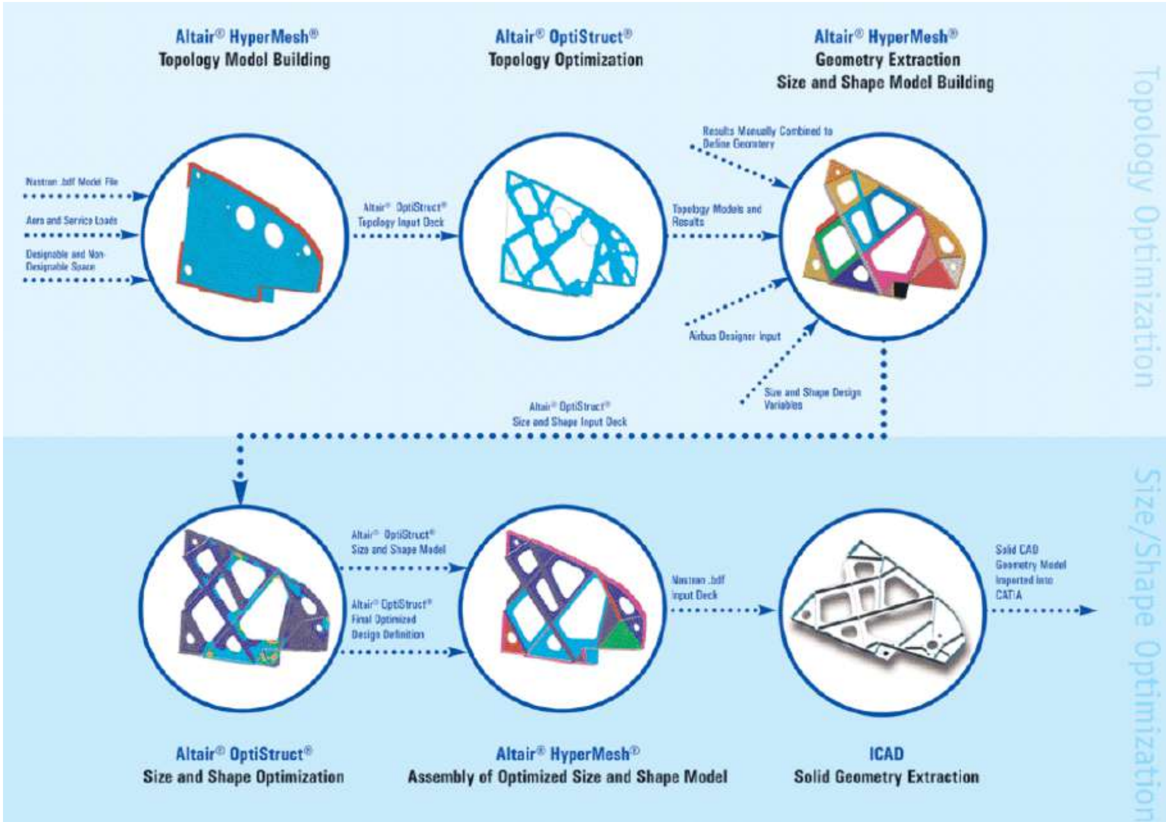
Leading edge ribs

- Objective: reduce weight
- Constraints: stress, buckling

Altair Engineering

Topology and shape optimization

This process will be shortcuted in Part4

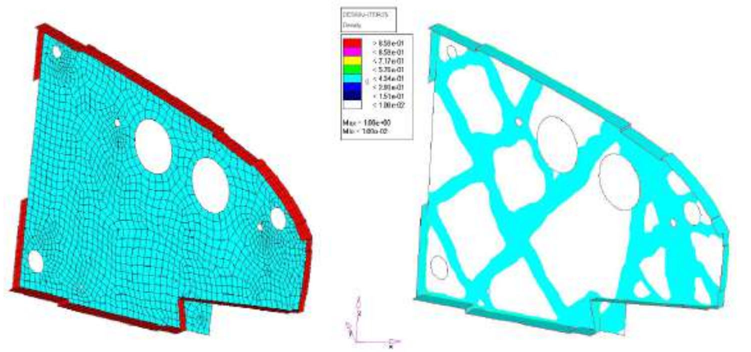


Schematic Representation of Enhanced Rib Design Process

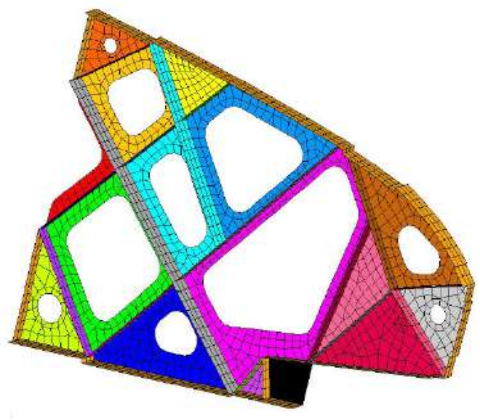
Altair Engineering

Airbus A380 example (cont.)

- Topology optimization:

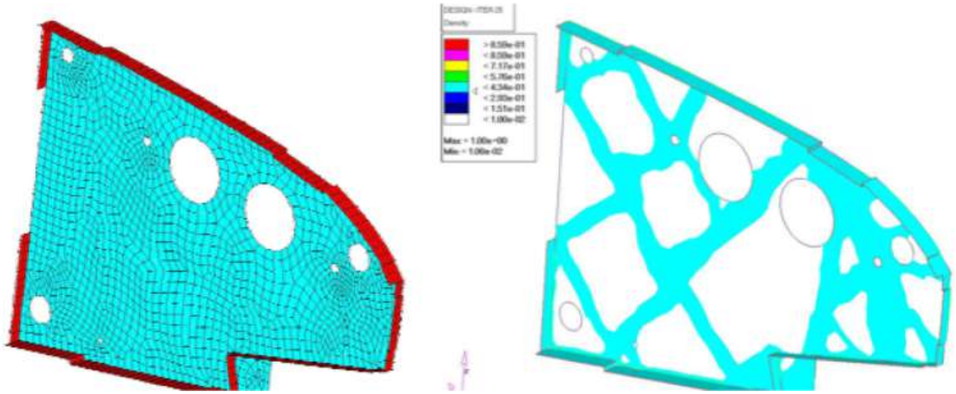


- Sizing / shape optimization:



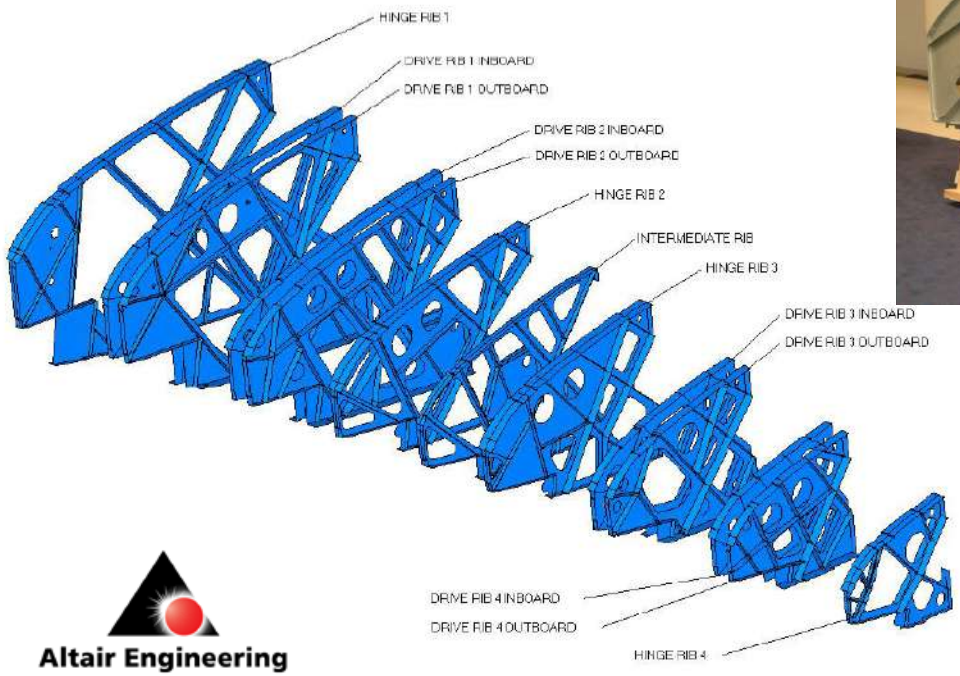
Altair Engineering

Finally…


Density
> 8.59e-01
< 8.59e-01
< 7.17e-01
< 5.76e-01
< 4.34e-01
< 2.93e-01
< 1.51e-01
< 1.00e-02
Max = 1.00e+00
Min = 1.00e-02

Airbus A380 example (cont.)

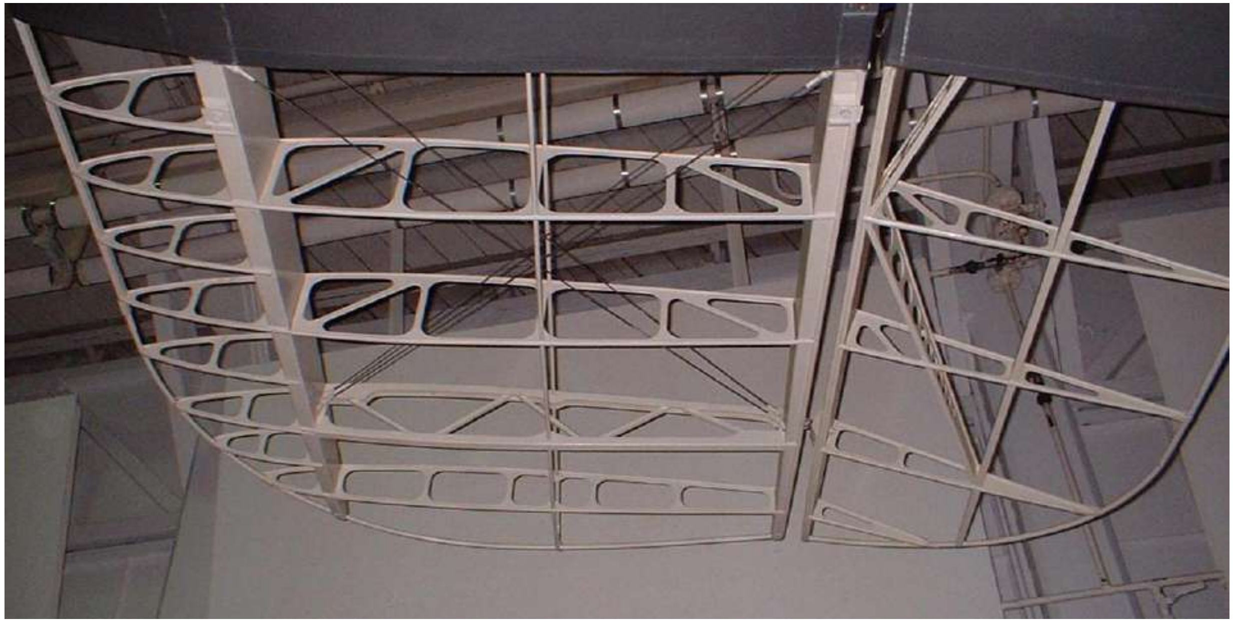
- Result: 500 kg weight savings!



Altair Engineering

Is this really a discovery?

Supermarine Southampton, 1925



System approach automates the process!

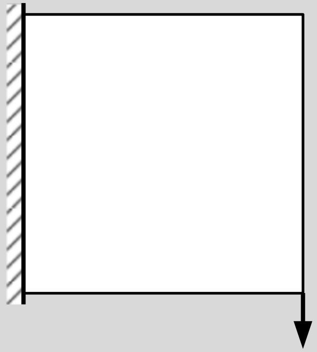
Industrial probems

- TopOpt: Preliminary phases of a project
- The idea is to find the best path of stiffening in a given volume of matter.
- The mass is only found where it is needed, which is a good starting point for optimization of shape or dimensioning.

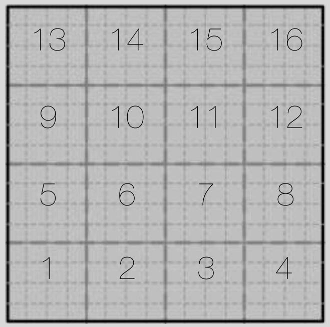
- The structure with the best static behavior.
- The paths of internal forces identified are those which help to rigidify the structure as well as possible

Inside the material ?→ DMO

Maximum stiffness in the plane of a plate by selecting the best orientations of fibers



Loads and boundary conditions



Design model with 4* 4 patches

Table 4 Material properties

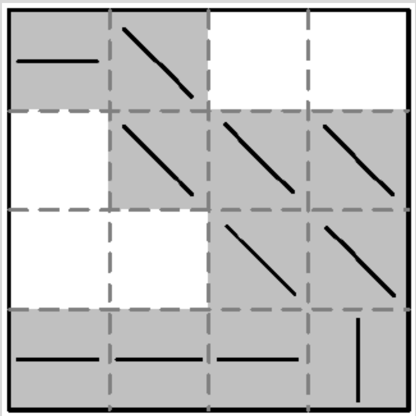
E_x	E_y	G_{xy}	v_{xy}
146.86GPa	10.62GPa	5.45GPa	0.33

Table 3 Orientations

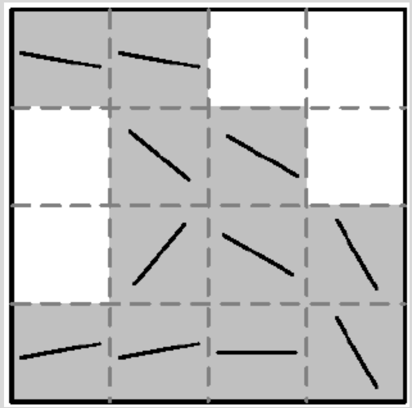
Number of material phases (m)	Number of design variables for each region (m_v)	Discrete orientation angle (°)
4	2	90/45/0/-45
9	4	80/60/40/20/0/-20/-40/-60/-80
12	4	90/75/60/45/30/15/0/-15/-30/-45/-60/-75

Discrete Material Optimization: exemple

Topological optimization: vacuum + composite laminate
Volume constraints: V <11/16



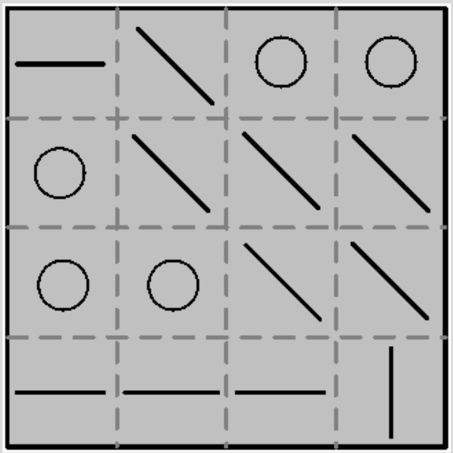
4 orientations
90/45/0/-45



18 orientations
90/80/70/60/50/40/30/20/10/0/
-10/-20/-30/-40/-50/-60/-70/-80

Discrete Material Optimization: exemple

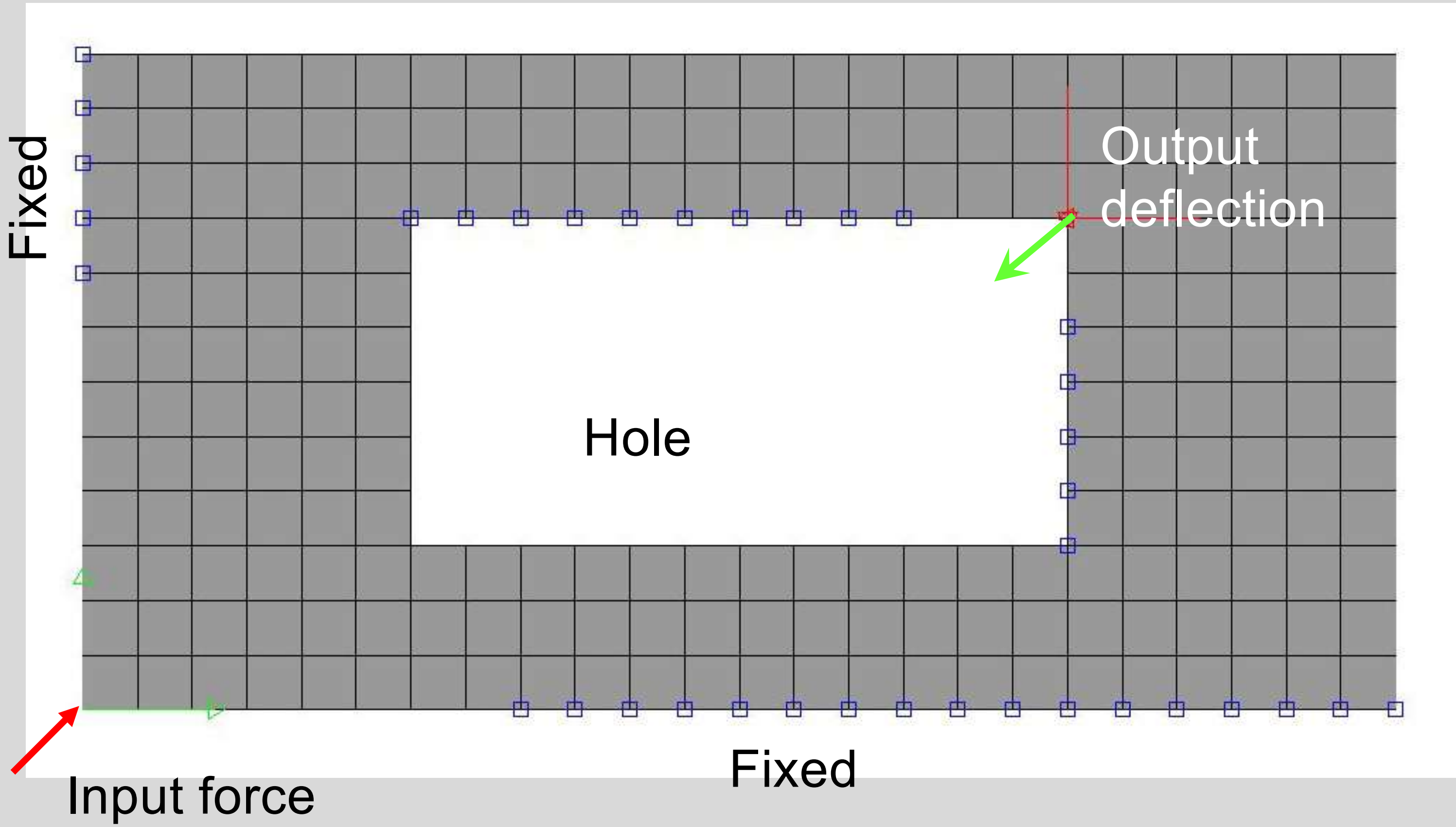
Use of both glass fibers and foam
Limitation of the number of domains occupied by the fiber of glass



Optimization result of the square plate under vertical force with volume constraint
Glass-epoxy with 4 orientations (90/45/0/-45) and polymer-foam

Compliant mechanism to your specifications

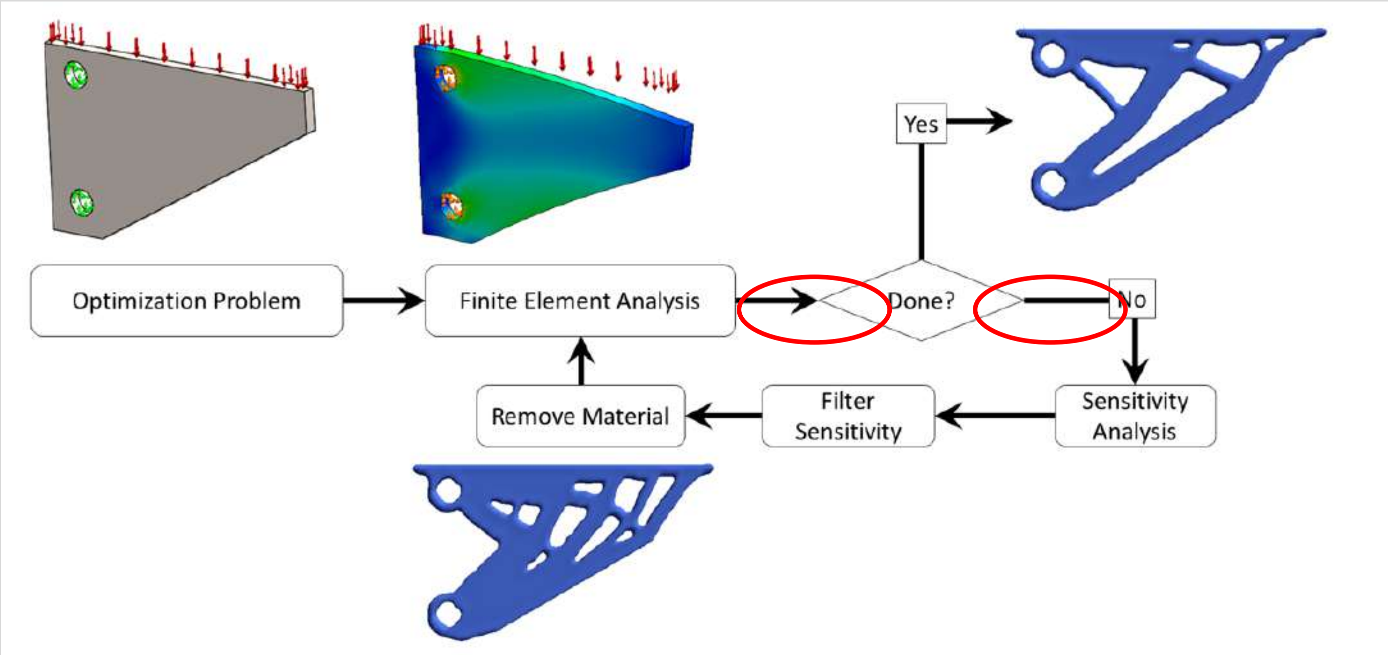
Use 30 % material



Compliant mechanism to your specifications

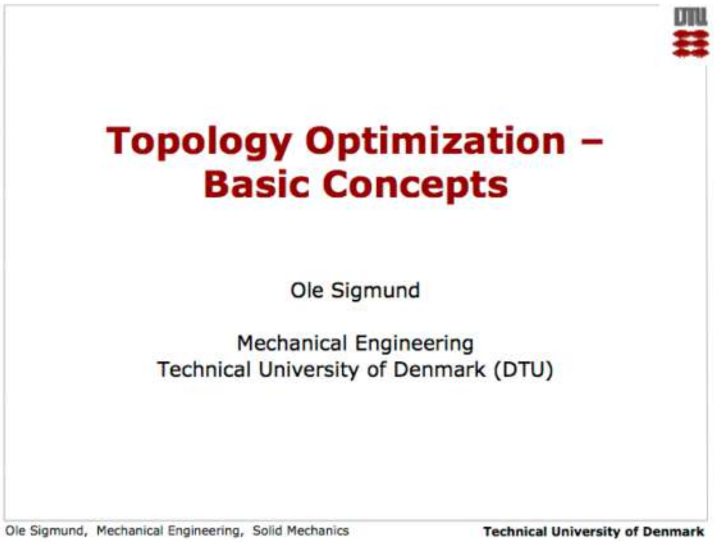
www.mecheng.iisc.ernet.in/~suresh/shortcourse

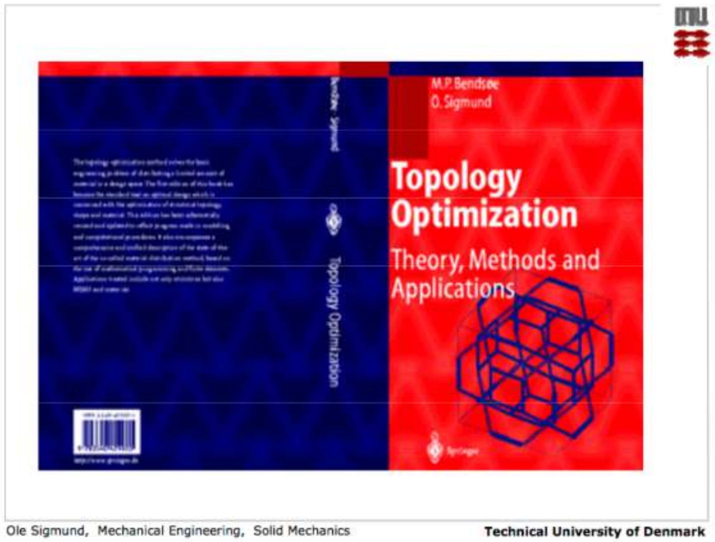
TopOpt relies on FEA
Online computation: http://www.cloudtopopt.com



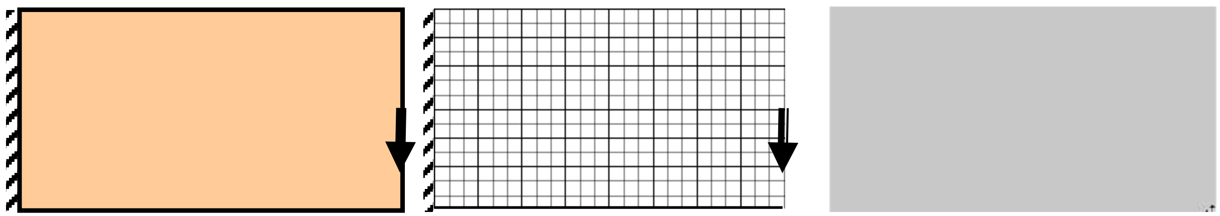
One pioneer, **SIMP** (Solid Isotropic Material with **Penalization**)

~~Homogenization~~ ~~Level-Set~~ ~~Evolutionary~~





SIMP: Solid Isotropic Material with Penalization



Min Compliance

$v = 0.5v_0$

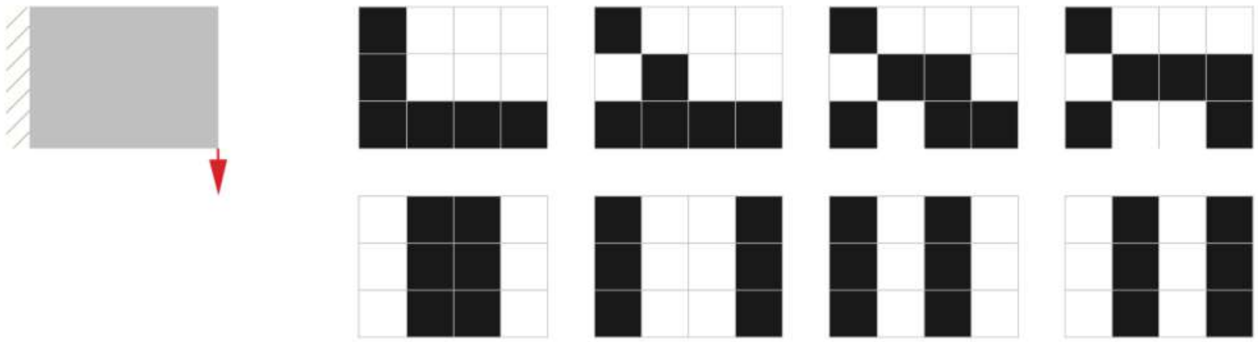
$$0 < \rho_e \leq 1 : \text{'PseudoDensity'}$$

$$\underset{\rho_e}{Min} \ \text{Compliance}$$

$$\sum \rho_e v_e = 0.5 v_0$$

Pixels

- Finding a solution by checking all the possible combinations IS impossible since the number of topologies nT increases exponentially with the number of finite elements n
- nT = 2^n,



The legal (top) and some illegal (bottom) topologies with 4 by 3 elements

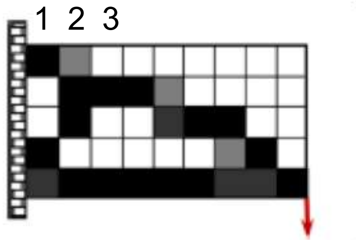
Division into elements (pixels or voxels) and binary decision for each
or example 10,000 elements --> 210,000 possible configurations!

Pixels?

When the size of the FE model is increasing, the SIMP optimization problem is … increasing

Chris Columbus et al, Pixels, movie 2015

Intuitive Problem? Quadratic Form



$x_1 = 1$
$x_2 = 0.5$
$x_3 = 0$
$\dots$

- Objective function; Strain energy

$$\min c(\mathbf{x}) = \mathbf{U}^T\mathbf{F} = \mathbf{U}^T\mathbf{K}\mathbf{U} \qquad \text{with} \quad x_e = \frac{\rho_e}{\rho_0} \qquad (4)$$

with $\mathbf{K} = \mathbf{K}_0 \sum_{e=1}^{N} x_e^p$ one can write:

$$\min c(\mathbf{x}) = \sum_{e=1}^{N} (x_e)^p \mathbf{u}_e^T \mathbf{k}_0 \mathbf{u}_e \qquad (5)$$

Scalar

- Contraints: mass target

$$\frac{V(\mathbf{x})}{V_0} = f = const \iff \sum_{e=1}^{N} V_e x_e - V_0 f = 0 = h(\mathbf{x})$$

Scalar

$$0 < \rho_{\min} \le \rho_e \le 1$$

$$\min c(\mathbf{x}) = \sum_{e=1}^{N} (x_e)^p \mathbf{u}_e^T \mathbf{k}_0 \mathbf{u}_e$$

Quadratic Form

$x \in \mathbb{R}^{m\times 1}$, $\mathbb{A} \in \mathbb{R}^{m\times m}$

$$\boxed{\text{Quadratic form : } x^T \mathbb{A} x}$$

$x^T \mathbb{A} x$ is a scalar value.

$(1\times m)\times(m\times m)\times(m\times 1) \rightarrow 1\times 1$

K is linked through E and xe

Rozvany, G.I.N. , Zhou, M., and Gollub, M. (1989). Continuum Type Optimality Criteria Methods for Large Finite Element Systems with a Displacement Connstraint, Part 1. *Structural Optimization* 1:47-72.

$$\mathbf{K} = \mathbf{K}_0 \sum_{e=1}^{N} x_e^p \qquad x_e = \frac{\rho_e}{\rho_0}$$

What is p, (simP)?????????

- **But HOW ??**

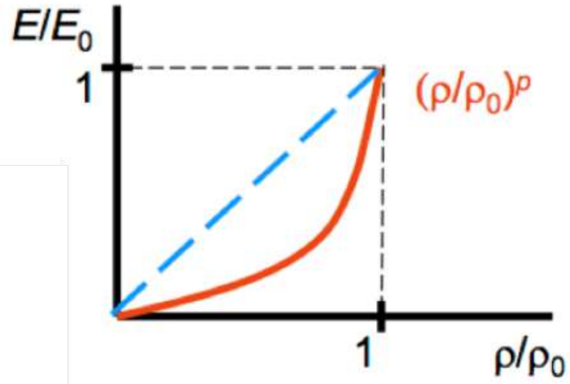
Avoid intermediate densities !

Solid Isotropic Material with Penalization (SIMP)

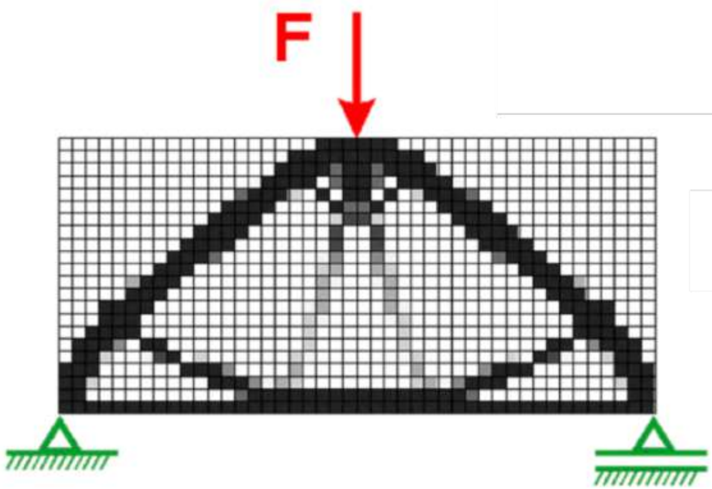
$$E(x) = E_{min} + (E_0 - E_{min})x^p$$

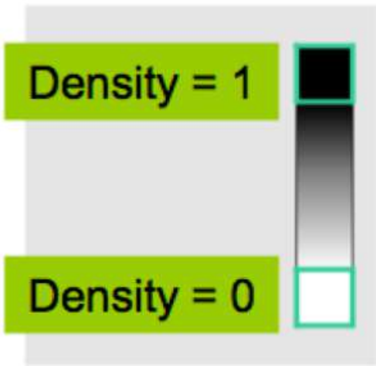
p is the penalty parameter to push densities to black (1) and white (0).

E_{min} is a small value that avoid stiffness matrix singularity

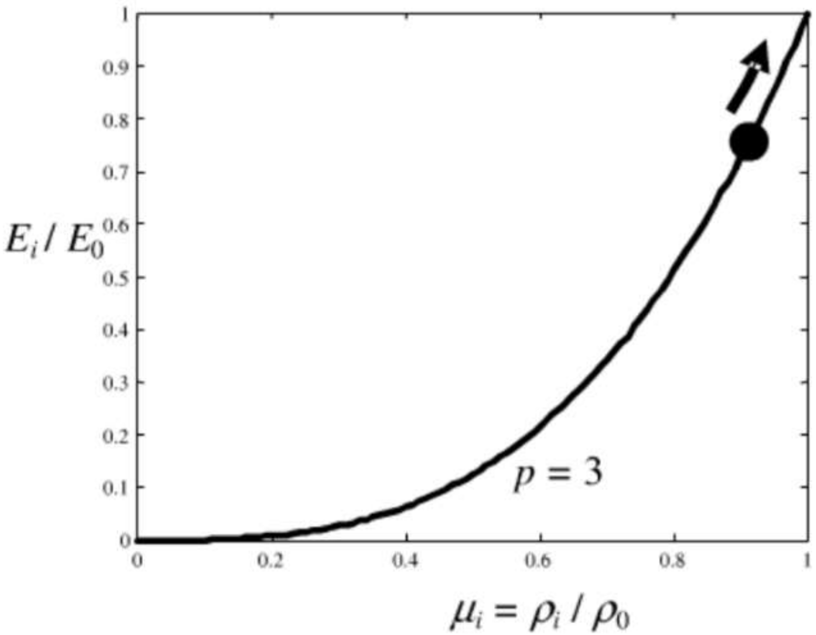


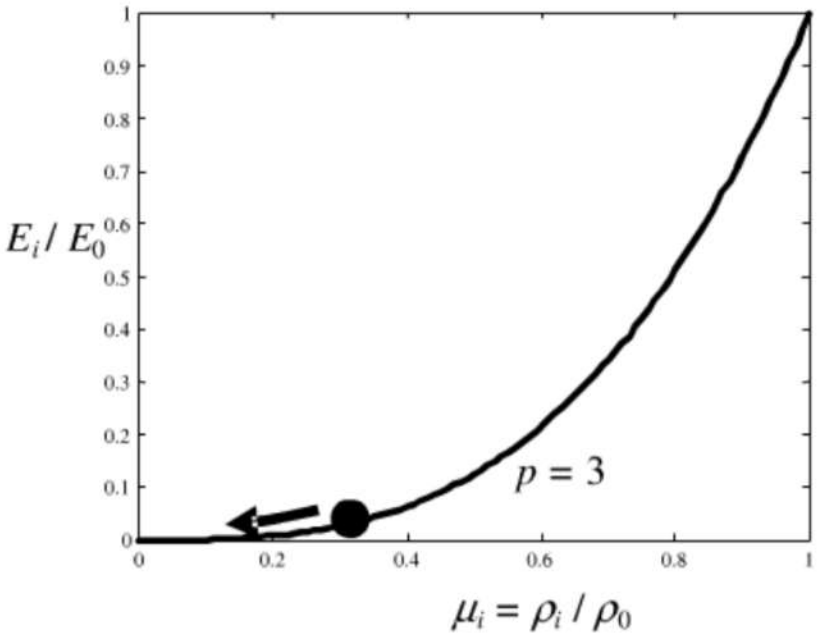
Penalization for altering stiffness localy





Let' s take p=3





Penalty parameter in the SIMP method: Proof

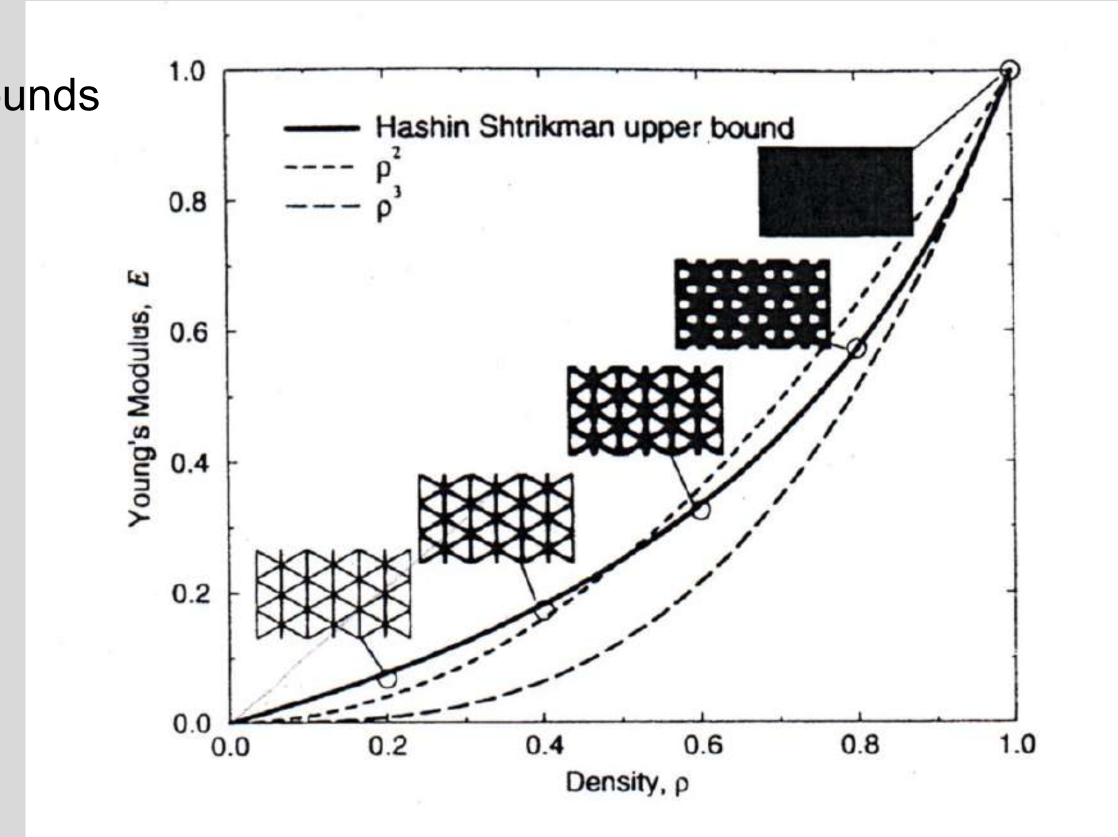
Hashin-Shtrikman bounds

$$0 \leq E \leq \frac{\rho E^0}{3 - 2\rho}$$

Therefore,

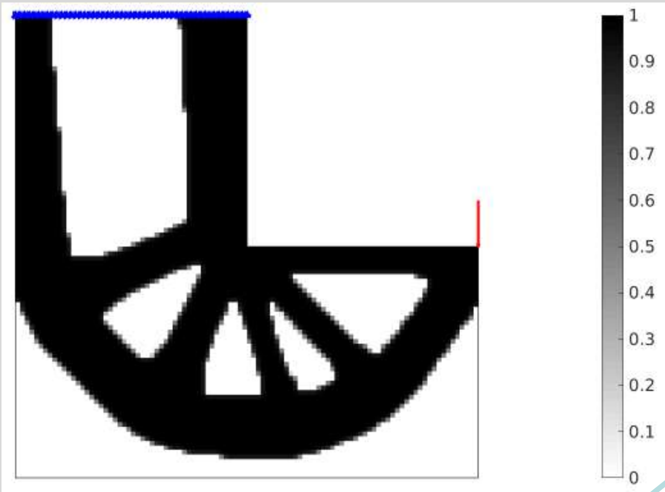
$$\rho^p E^0 \leq \frac{\rho E^0}{3 - 2\rho}$$

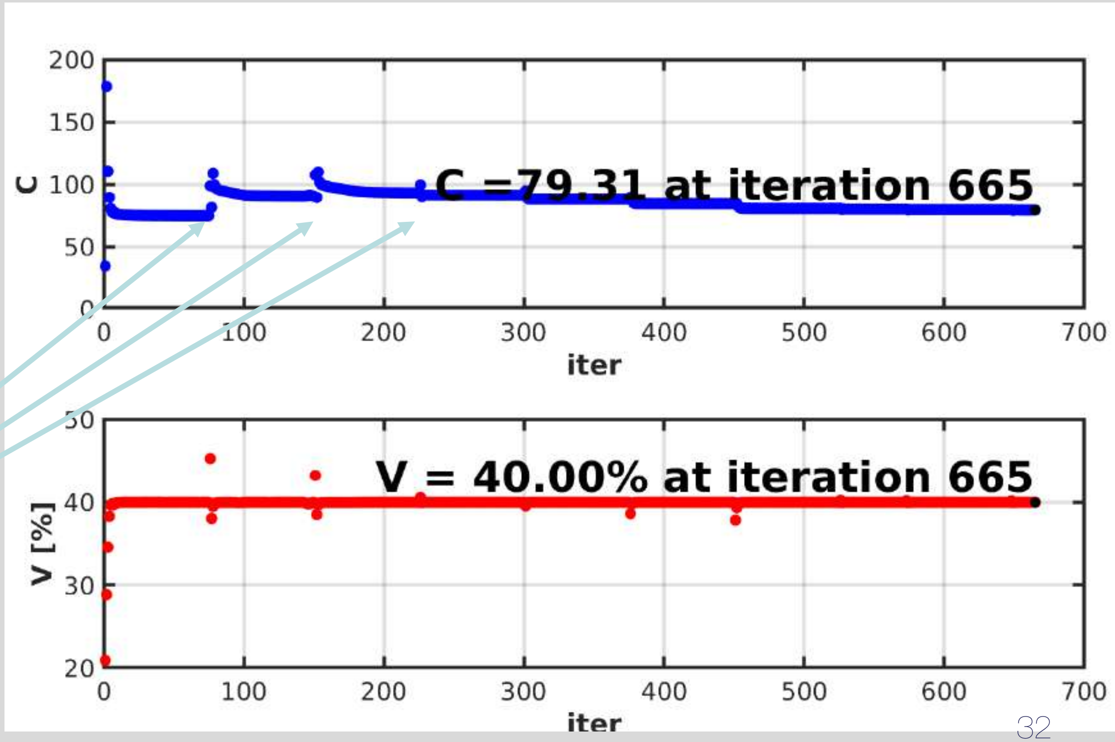
$$\Rightarrow p \geq 3$$



Bendsøe, M.P. and Sigmund, O., "Material Interpolation Schemes in Topology Optimization," *Archives in Applied Mechanics*, Vol. 69, (9-10), 1999, pp. 635-654.

Penalty parameter in the SIMP method: Continuation Methods





Increase p
Reduce the filter radius
Increase beta

REALLY Nice idea !

1. Transform discrete variables continuously (TO USE gradient-based algorithms)
2. Find an objective function with "cheap" derivatives (we will see this later)

Others formulations

$$\min_{\boldsymbol{\mu}} \max_{l=1,\ldots,nc} C_l = \mathbf{F}_l^T \mathbf{q}_l$$
$$\sum_i \mu_i V_i \leq \overline{V}$$
$$0 < \underline{\mu}_i \leq \mu_i \leq 1$$

- If several load cases nc

→ we can minimize the maximal compliance

→ with ql obtained by solving Klql=Fl

$$\min_{\boldsymbol{\mu}} \sum_i \mu_i V_i$$
$$q_j \leq \overline{q}_j \qquad j = 1,\ldots,m$$
$$0 < \underline{\mu}_i \leq \mu_i \leq 1$$

- Prescribed displacement

→ we can minimize the volume (mass)

→ wrt amplitude at node j inferior to a certain displacement

Others formulations

$$\max_{\boldsymbol{\mu}} \min_{k=1,nf} \omega_k$$

$$\sum_i \mu_i V_i \le \overline{V}$$

$$0 < \underline{\mu}_i \le \mu_i \le 1$$

- Eigensolver to obtain the stiffest structure at a certain volfrac

→ wrt a vibration ccriteria

linear buckling, thermal, thermoelasticity etc

This is not Sauron's eye

DOI. 10.1137/070699822

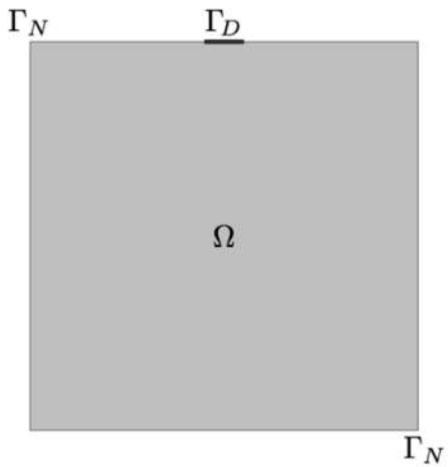
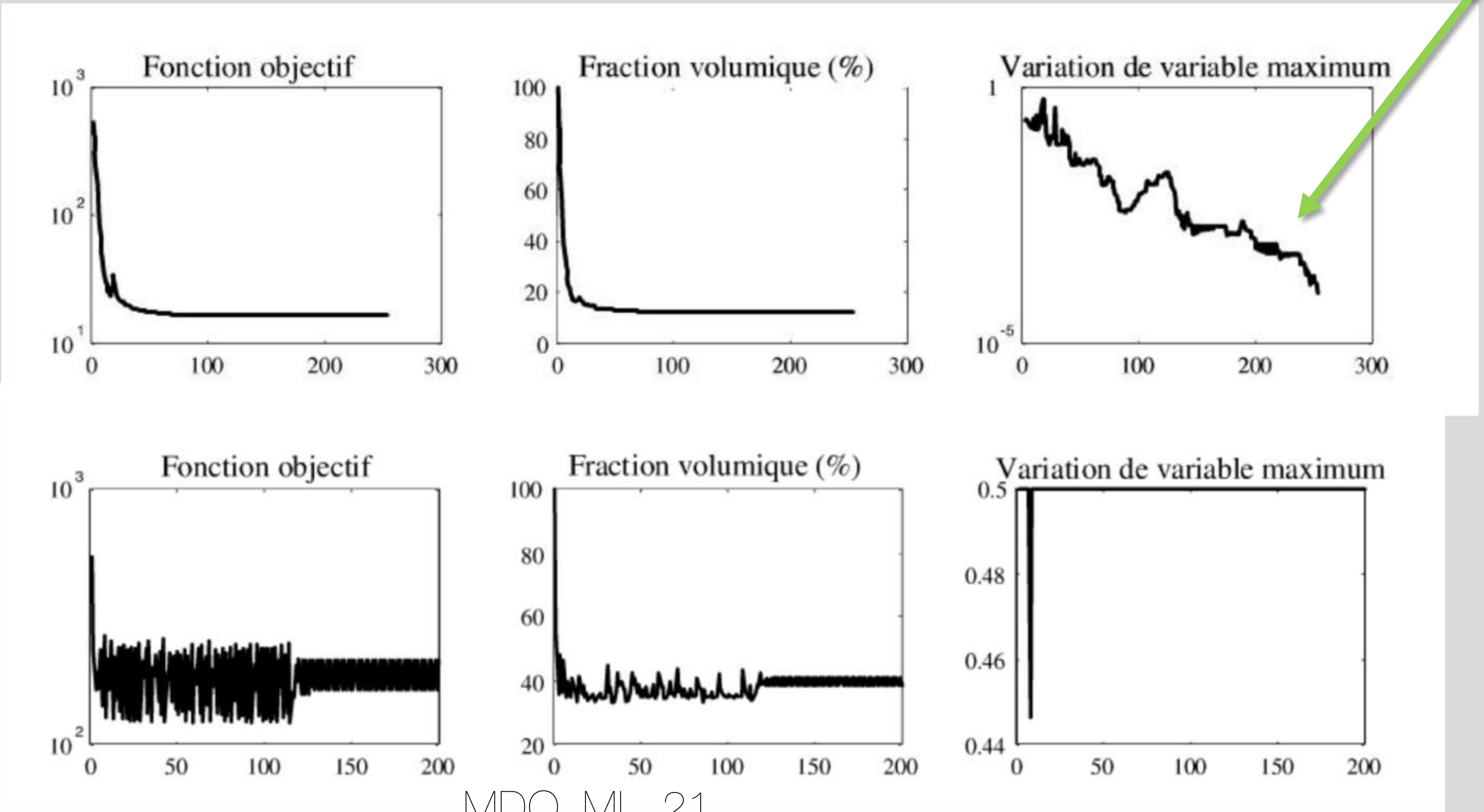


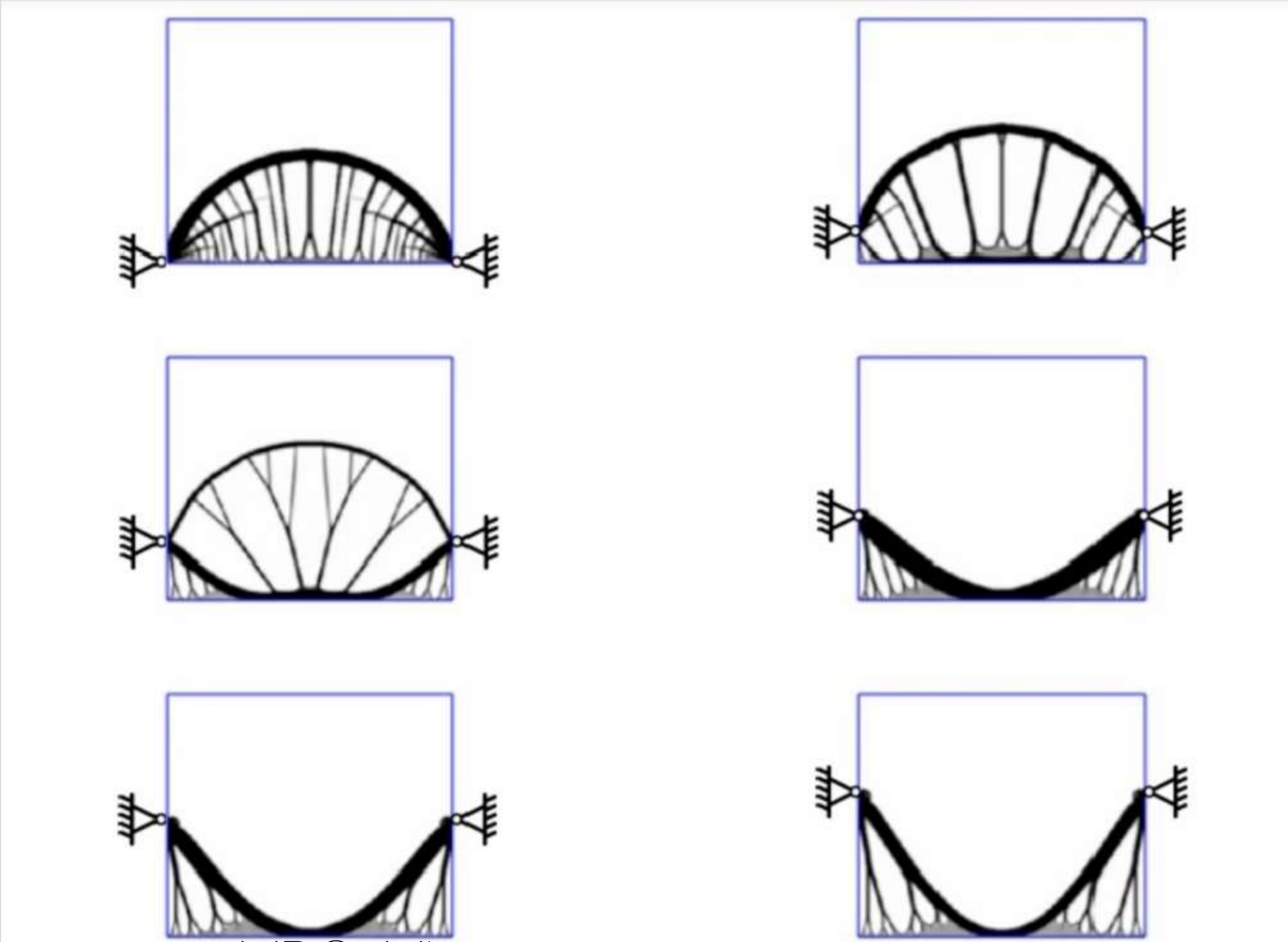
Fig. 2.1 *The problems consist of finding the distribution within Ω of two materials with different heat conduction properties in order to obtain a temperature field that is as even as possible.*

Which is the best optimizer? why ?



MDO_ML_21

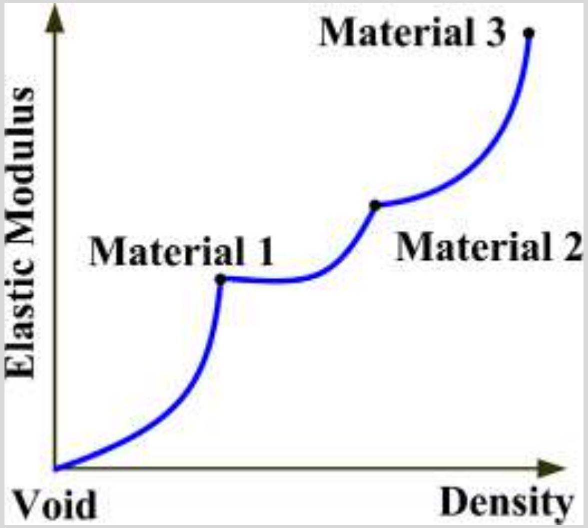
Small changes in BCs ...



MULTIMATERIALS

- Solid Isotropic Material with Penalization (SIMP)
- $E_e(\rho_e) = A_E * \rho_e^p + B_E$,

$\rho_e \in [\rho_i, \rho_{i+1}]$, $A_E = \frac{E_i - E_{i+1}}{\rho_i^p - \rho_{i+1}^p}$, $B_E = E_i - A_E * \rho_i^p$



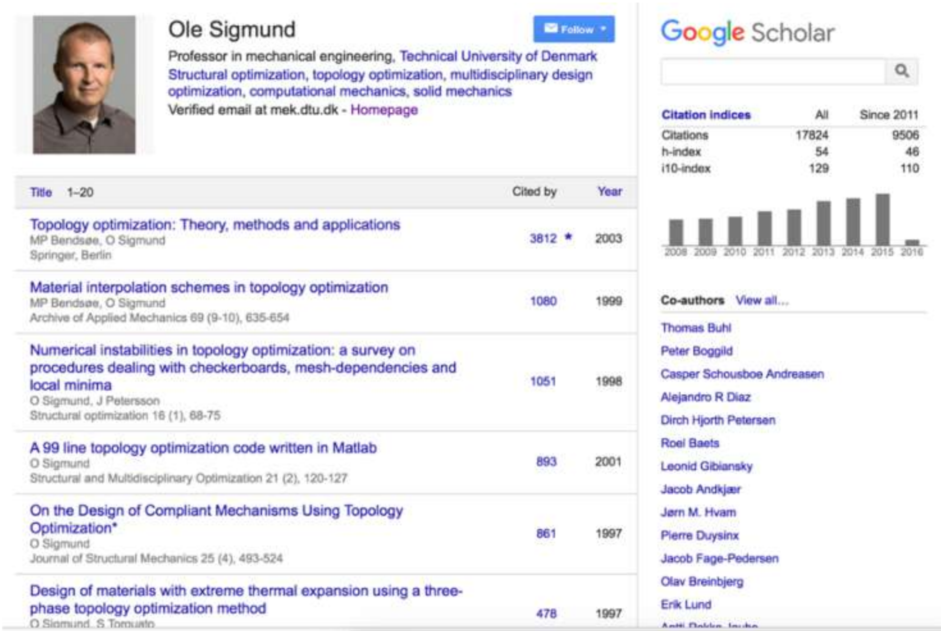
Zuo, W., & Saitou, K. (2016). Multi-material topology optimization using ordered SIMP interpolation. *Structural and Multidisciplinary Optimization, 55*(2), 477-491. doi:10.1007/s00158-016-1513-3

BUT …IN PRACTICE?

Educational article:

O. Sigmund , A 99 line topology optimization code written in Matlab Struct Multidisc Optim 21, 120–127 Springer-Verlag 2001

Heuristic formulation (intuitiive method of optimisation, but with no convergency proofs) to update xe by bi-section algorithm



Ole Sigmund

Follow

Professor in mechanical engineering, Technical University of Denmark
Structural optimization, topology optimization, multidisciplinary design optimization, computational mechanics, solid mechanics
Verified email at mek.dtu.dk - Homepage

Title 1–20	Cited by	Year
Topology optimization: Theory, methods and applications MP Bendsøe, O Sigmund Springer, Berlin	3812 *	2003
Material interpolation schemes in topology optimization MP Bendsøe, O Sigmund Archive of Applied Mechanics 69 (9-10), 635-654	1080	1999
Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima O Sigmund, J Petersson Structural optimization 16 (1), 68-75	1051	1998
→ A 99 line topology optimization code written in Matlab O Sigmund Structural and Multidisciplinary Optimization 21 (2), 120-127	893	2001
On the Design of Compliant Mechanisms Using Topology Optimization* O Sigmund Journal of Structural Mechanics 25 (4), 493-524	861	1997
Design of materials with extreme thermal expansion using a three-phase topology optimization method O Sigmund, S Torquato	478	1997

Google Scholar

Citation indices	All	Since 2011
Citations	17824	9506
h-index	54	46
i10-index	129	110

Co-authors View all...

Thomas Buhl
Peter Boggild
Casper Schousboe Andreasen
Alejandro R Diaz
Dirch Hjorth Petersen
Roel Baets
Leonid Gibiansky
Jacob Andkjær
Jørn M. Hvam
Pierre Duysinx
Jacob Fage-Pedersen
Olav Breinbjerg
Erik Lund

History (1988, Bendsoe)

A topology optimization problem based on the power-law approach, where the objective is to minimize compliance can be written as

$$\left.\begin{array}{rl} \min_{\mathbf{x}}: & c(\mathbf{x}) = \mathbf{U}^T\mathbf{K}\mathbf{U} = \sum_{e=1}^{N}(x_e)^p\ \mathbf{u}_e^T\ \mathbf{k}_0\ \mathbf{u}_e \\ \text{subject to}: & \dfrac{V(\mathbf{x})}{V_0} = f \\ : & \mathbf{K}\mathbf{U} = \mathbf{F} \\ : & \mathbf{0} < \mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{1} \end{array}\right\}, \tag{1}$$

where $\mathbf{U}$ and $\mathbf{F}$ are the global displacement and force vectors, respectively, $\mathbf{K}$ is the global stiffness matrix, $\mathbf{u}_e$ and $\mathbf{k}_e$ are the element displacement vector and stiffness matrix, respectively, $\mathbf{x}$ is the vector of design variables, $\mathbf{x}_{\min}$ is a vector of minimum relative densities (non-zero to avoid singularity), N $(= \texttt{nelx} \times \texttt{nely})$ is the number of elements used to discretize the design domain, p is the penalization power (typically $p = 3$), $V(\mathbf{x})$ and V_0 is the material volume and design domain volume, respectively and f (`volfrac`) is the prescribed volume fraction.

Compliance minimization **self adjoint**

- Compliance is the opposite of stiffness

$$C = \mathbf{f}^T \mathbf{u} = \mathbf{u}^T K \mathbf{u}$$

- Inexpensive derivatives (use chain rule)

$$\frac{dC}{dx} = 2\mathbf{u}^T K \frac{d\mathbf{u}}{dx} + \mathbf{u}^T \frac{dK}{dx}\mathbf{u}$$

But since $K\mathbf{u} = \mathbf{f}$ if $\mathbf{f}$ does not depend on x

$$K\frac{d\mathbf{u}}{dx} = -\frac{dK}{dx}\mathbf{u}$$

$$\frac{dC}{dx} = -\mathbf{u}^T \frac{dK}{dx}\mathbf{u}$$

Knowing displacements you also know gradients

Density design variables

Need a DEMO ? see next slides

- Recall $$\frac{dC}{dx} = -\mathbf{u}^T \frac{dK}{dx}\mathbf{u}$$
- For density variables

$$\frac{dC}{d\rho^e} \propto -\mathbf{u}^T \rho^{p-1} K^e \mathbf{u}$$

- Want to increase density of elements with high strain energy and vice versa
- To minimize compliance for given weight can use an optimality criterion method.

$$\boxed{C = f^T u}$$

F is not depending on x

$$\frac{\partial C}{\partial x} = \frac{\partial f^T}{\partial x} u + f^T \frac{\partial u}{\partial x} = \phi + f^T \frac{\partial u}{\partial x} \quad (1)$$

starting by expressing the derivative of $K u = f$.

it gives $\frac{\partial K}{\partial x} u + K \frac{\partial u}{\partial x} = \frac{\partial f}{\partial x} = 0$

$$\Leftrightarrow \boxed{K \frac{\partial u}{\partial x} = - \frac{\partial K}{\partial x} u}$$

$$(1) \quad \frac{\partial c}{\partial x} = f^T \frac{\partial u}{\partial x} = u^T \left(K \frac{\partial u}{\partial x}\right) = u^T \left(- \frac{\partial K}{\partial x}\right) u = - u^T \frac{\partial K}{\partial x} u$$

$$= \frac{\partial c}{\partial x}$$

or by expressing $C = f^T u$

as $C = u^T K u$ (quadratic form) (2)

$$dC = \frac{\partial C}{\partial u} du + \frac{\partial C}{\partial x} dx \implies \boxed{\frac{dC}{dx}} = \frac{\partial C}{\partial u}\frac{du}{dx} + \frac{\partial C}{\partial x}$$

$$\left(\text{Recap } \frac{\partial u^T K u}{\partial u} = 2u^T K\right)$$

from (2) $\quad \frac{\partial C}{\partial u} = 2u^T K \frac{du}{dx}$

itwones $\quad \frac{dC}{dx} = 2u^T K \frac{du}{dx} + u^T \frac{\partial K}{\partial x} u$

$$= -2u^T \frac{\partial K}{\partial x} u + u^T \frac{\partial K}{\partial x} u = \boxed{-u^T \frac{\partial K}{\partial x} u} = \frac{\partial C}{\partial x}$$

Matlab Code (the tutorial will be in python)

```
x(1:nely,1:nelx) = volfrac;                                    % INITIALIZE

loop = 0; change = 1.;
while change > 0.01                                            % START ITERATION While Xk+1>>Xk
              loop = loop + 1;
              xold = x;
              [U]=FE(nelx,nely,x,penal);       % FE-ANALYSIS
              [KE] = lk;
              c = 0.;

for ely = 1:nely
  for elx = 1:nelx
    n1 = (nely+1)*(elx-1)+ely;
    n2 = (nely+1)* elx +ely;
    Ue = U([2*n1-1;2*n1; 2*n2-1;2*n2; 2*n2+1;2*n2+2; 2*n1+1;2*n1+2],1);
    c = c + x(ely,elx)^penal*Ue'*KE*Ue;                        % OBJECTIVE FUNCTION
    dc(ely,elx) = -penal*x(ely,elx)^(penal-1)*Ue'*KE*Ue; % SENSITIVITY ANALYSIS
end
```

Sensitivity

$$\frac{\partial c}{\partial x_e} = -p(x_e)^{p-1}\, \mathbf{u}_e^T\, \mathbf{k}_0\, \mathbf{u}_e$$

Update rule

- OPTIMALITY CRITERIA METHOD

$$\begin{cases} \max(x_{\min}, x_e - m) & \text{if} \quad x_e B_e^\eta \le \max(x_{\min}, x_e - m)\,, \\ x_e B_e^\eta & \text{if} \quad \max(x_{\min}, x_e - m) < x_e B_e^\eta < \min(1, x_e + m) \\ \min(1, x_e + m) & \text{if} \quad \min(1, x_e + m) \le x_e B_e^\eta\,, \end{cases}$$

$$B_e = \frac{-\dfrac{\partial c}{\partial x_e}}{\lambda \dfrac{\partial V}{\partial x_e}}$$

Element Stiffness Matrix

```
function [KE]=lk
%Element Stiffness Matrix

E = 1.;
nu =1/3.;
k=[ 1/2-nu/6 1/8+nu/8 -1/4+nu/12 -1/8-nu/8
-1/4-nu/12 -1/8+3*nu/8 ... nu/6 1/8-3*nu/8];
KE = E/(1-nu^2)* ...
[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3) k(3) k(8) k(1) k(6)
k(7) k(4) k(5) k(2) k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4) k(6) k(5) k(4) k(3)
k(2) k(1) k(8) k(7) k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)

k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];
```

FEM Analysis (2D mesh is invariant wrt to homotheties)

```
function [U]=FE(nelx,nely,x,penal)
[KE] = lk;
K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
F = sparse(2*(nely+1)*(nelx+1),1); U = zeros(2*(nely+1)*(nelx+1),1);
for elx = 1:nelx
 for ely = 1:nely
  n1 = (nely+1)*(elx-1)+ely;
  n2 = (nely+1)* elx +ely;
  edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1; 2*n1+2];
  K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
end
 end
F(2*(nelx+1)*(nely+1),1)=-1;
fixeddofs=union([1,2],[2*nely+1:2*(nely+1)]);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs,fixeddofs);
% SOLVING
U(freedofs,:) = K(freedofs,freedofs) \ F(freedofs,:);
U(fixeddofs,:)= 0;
```

TO HAVE REAL DISPLACEMENT (see tutorial on github)

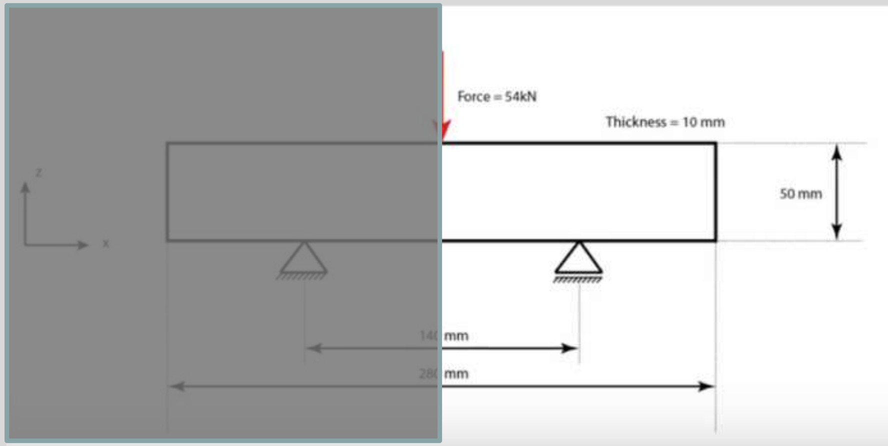
1) Choose consistent units N, mm, MPa for example *(Remember Nastran Course)*
2) Put the real Young's modulus E=210e3 MPa for example;
3) Multiply the unit load by true amplitude F for example 54*1e3 N ;
4) Multiply the elementary stiffness matrix by the thickness (mm)
5) 2D mesh is invariant wrt to homotheties ; Need to check that nelx and nely are related to the true value for example 140 and 50 mm
6) Apply the BCs

The compliance unit is mJ.

An engineering example (tutorial available on Github)

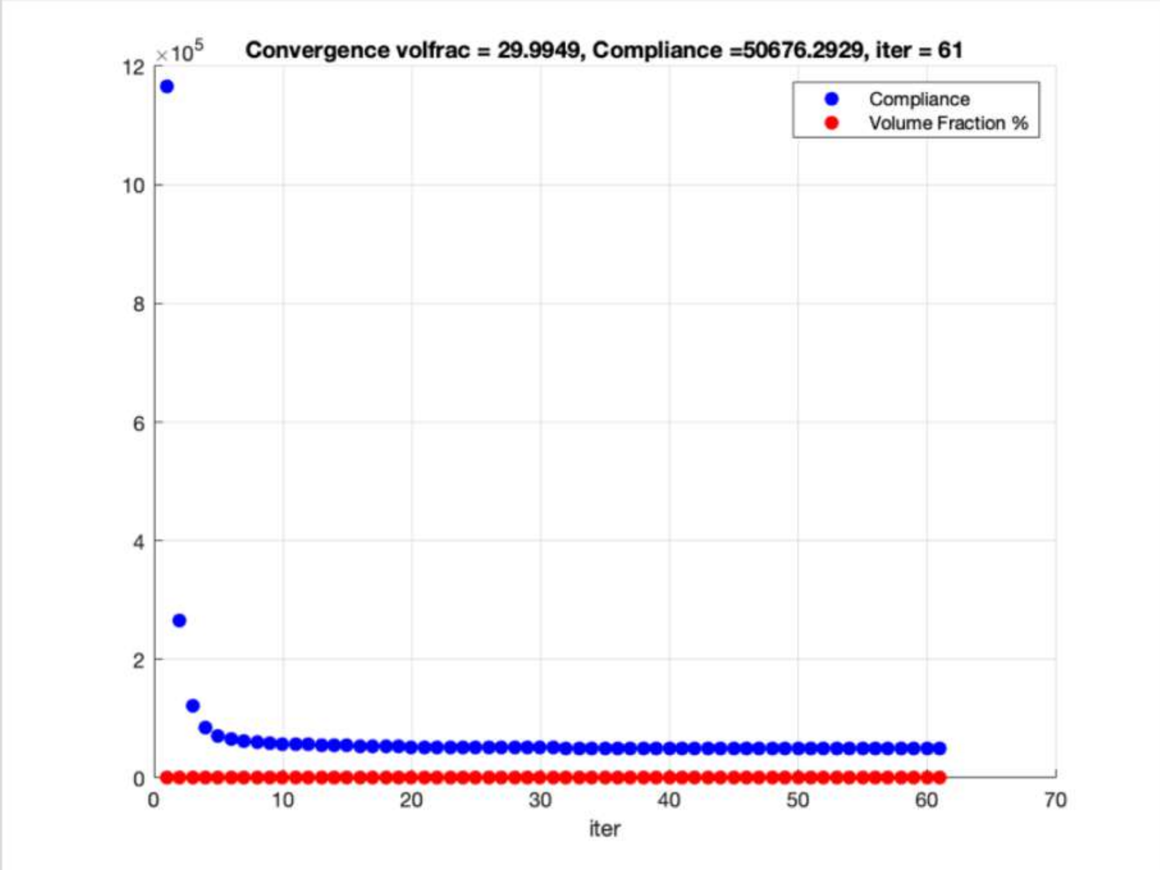
- Search the optimal 2D topology using symmetry

--> modify top88.m



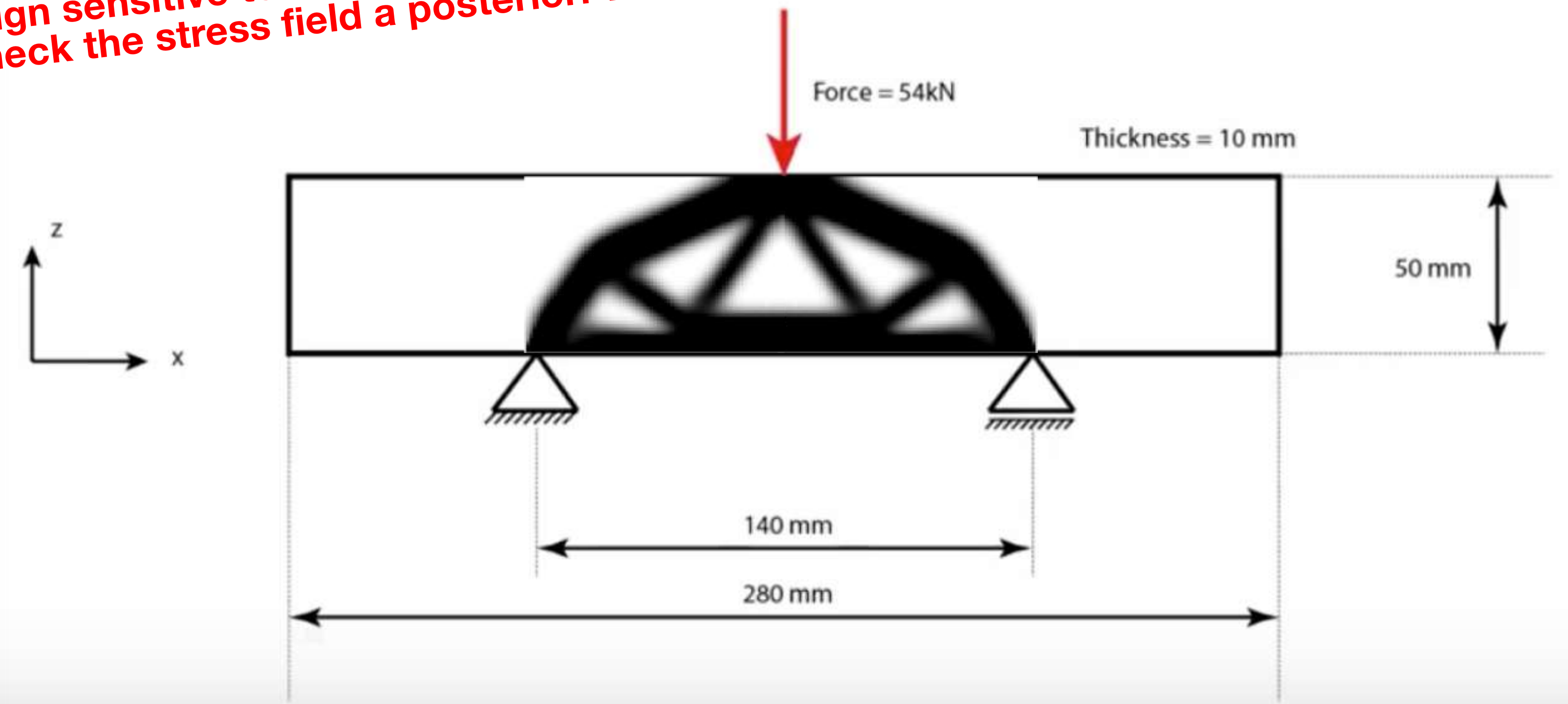
top88_ptBENDING(140, 50, 0.3, 3, 2, 2)

2 Outputs, X* + feasibilty/optimality graph



Outputs

Need to add constraint max displacement ?
Need to produce a Pareto front design vs volfrac?
How do you experimentaly introduce the external force?
Is the design sensitive to BCs size ?
Do you check the stress field a posteriori ?



```
%%%% AN 88 LINE TOPOLOGY OPTIMIZATION CODE Nov, 2010 %%%%
function x=top88(nelx,nely,volfrac,penal,rmin,ft)
%% MATERIAL PROPERTIES
E0 = 1;
Emin = 1e-9;
nu = 0.3;
%% PREPARE FINITE ELEMENT ANALYSIS
A11 = [12  3 -6 -3;  3 12  3  0; -6  3 12 -3; -3  0 -3 12];
A12 = [-6 -3  0  3; -3 -6 -3 -6;  0 -3 -6  3;  3 -6  3 -6];
B11 = [-4  3 -2  9;  3 -4 -9  4; -2 -9 -4 -3;  9  4 -3 -4];
B12 = [ 2 -3  4 -9; -3  2  9 -2;  4  9  2  3; -9 -2  3  2];
KE = 1/(1-nu^2)/24*([A11 A12;A12' A11]+nu*[B11 B12;B12' B11]);
nodenrs = reshape(1:(1+nelx)*(1+nely),1+nely,1+nelx);
edofVec = reshape(2*nodenrs(1:end-1,1:end-1)+1,nelx*nely,1);
edofMat = repmat(edofVec,1,8)+repmat([0 1 2*nely+[2 3 0 1] -2 -1],nelx*nely,1);
iK = reshape(kron(edofMat,ones(8,1))',64*nelx*nely,1);
jK = reshape(kron(edofMat,ones(1,8))',64*nelx*nely,1);
% DEFINE LOADS AND SUPPORTS (HALF MBB-BEAM)
F = sparse(2,1,-1,2*(nely+1)*(nelx+1),1);
U = zeros(2*(nely+1)*(nelx+1),1);
fixeddofs = union([1:2:2*(nely+1)],[2*(nelx+1)*(nely+1)]);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs,fixeddofs);
%% PREPARE FILTER
[58 unmodified lines hidden]
  %% PRINT RESULTS
  fprintf(' It.:%5i Obj.:%11.4f Vol.:%7.3f ch.:%7.3f\n',loop,c, ...
    mean(xPhys(:)),change);
  %% PLOT DENSITIES
  colormap(gray); imagesc(1-xPhys); caxis([0 1]); axis equal; axis off; drawnow;
end
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[26 unmodified lines hidden]
```

```
. %%%% AN 88 LINE TOPOLOGY OPTIMIZATION CODE Nov, 2010 %%%%
x %example top88_joseph(140, 50, 0.3, 3, 2, 2)
> function x=top88_ptBENDING(nelx,nely,volfrac,penal,rmin,ft)
> close all
. %% MATERIAL PROPERTIES
x E0 = 210e3;
. Emin = 1e-9;
. nu = 0.3;
> thickness=10;
> Force_amplitude=54e3;
. %% PREPARE FINITE ELEMENT ANALYSIS
. A11 = [12  3 -6 -3;  3 12  3  0; -6  3 12 -3; -3  0 -3 12];
. A12 = [-6 -3  0  3; -3 -6 -3 -6;  0 -3 -6  3;  3 -6  3 -6];
. B11 = [-4  3 -2  9;  3 -4 -9  4; -2 -9 -4 -3;  9  4 -3 -4];
. B12 = [ 2 -3  4 -9; -3  2  9 -2;  4  9  2  3; -9 -2  3  2];
x KE = thickness*1/(1-nu^2)/24*([A11 A12;A12' A11]+nu*[B11 B12;B12' B11]);
. nodenrs = reshape(1:(1+nelx)*(1+nely),1+nely,1+nelx);
. edofVec = reshape(2*nodenrs(1:end-1,1:end-1)+1,nelx*nely,1);
. edofMat = repmat(edofVec,1,8)+repmat([0 1 2*nely+[2 3 0 1] -2 -1],nelx*nely,1);
. iK = reshape(kron(edofMat,ones(8,1))',64*nelx*nely,1);
. jK = reshape(kron(edofMat,ones(1,8))',64*nelx*nely,1);
. % DEFINE LOADS AND SUPPORTS (HALF MBB-BEAM)
x F = Force_amplitude*sparse(2,1,-1,2*(nely+1)*(nelx+1),1);
. U = zeros(2*(nely+1)*(nelx+1),1);
x fixeddofs = union([1:2:2*(nely+1)],[(nelx+2)*(nely+1)]);
. alldofs = [1:2*(nely+1)*(nelx+1)];
. freedofs = setdiff(alldofs,fixeddofs);
. %% PREPARE FILTER
[58 unmodified lines hidden]
.   %% PRINT RESULTS
.   fprintf(' It.:%5i Obj.:%11.4f Vol.:%7.3f ch.:%7.3f\n',loop,c, ...
.     mean(xPhys(:)),change);
> figure(2)
> hold on
> plot(loop,c,'bo','MarkerFaceColor','b')
> plot(loop,mean(xPhys(:))*100,'ro','MarkerFaceColor','r')
> % plot(outeriter,(1+GKS1)*VM1,'ko','MarkerFaceColor','k')
> title(['Convergence volfrac = ',num2str(mean(xPhys(:))*100),', Compliance =',num
> grid on
> legend('Compliance','Volume Fraction %')
> xlabel('iter')
.   %% PLOT DENSITIES
>    figure(1)
.   colormap(gray); imagesc(1-xPhys); caxis([0 1]); axis equal; axis off; drawnow;
>   print(['DZ_it',num2str(loop,'%3d')],'-dpng')
. end
. %
. %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[26 unmodified lines hidden]
```

OPTIMALITY CRITERIA

```
function [xnew]=OC(nelx,nely,x,volfrac,dc)
l1 = 0; l2 = 100000; move = 0.2;
while (l2-l1 > 1e-4)
 lmid = 0.5*(l2+l1);
 xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
 if sum(sum(xnew)) - volfrac*nelx*nely > 0;
   l1 = lmid;
 else
   l2 = lmid;
 end
end
```

lmid: 50.0000	l1: 0.0000	l2: 50.0000
lmid: 25.0000	l1: 0.0000	l2: 25.0000
lmid: 12.5000	l1: 0.0000	l2: 12.5000
lmid: 6.2500	l1: 6.2500	l2: 12.5000
lmid: 9.3750	l1: 6.2500	l2: 9.3750
lmid: 7.8125	l1: 6.2500	l2: 7.8125
lmid: 7.0313	l1: 7.0313	l2: 7.8125
lmid: 7.4219	l1: 7.0313	l2: 7.4219
lmid: 7.2266	l1: 7.2266	l2: 7.4219
lmid: 7.3242	l1: 7.3242	l2: 7.4219
lmid: 7.3730	l1: 7.3242	l2: 7.3730
lmid: 7.3486	l1: 7.3242	l2: 7.3486
lmid: 7.3364	l1: 7.3364	l2: 7.3486
lmid: 7.3425	l1: 7.3425	l2: 7.3486
lmid: 7.3456	l1: 7.3425	l2: 7.3456
lmid: 7.3441	l1: 7.3441	l2: 7.3456
lmid: 7.3448	l1: 7.3448	l2: 7.3456
lmid: 7.3452	l1: 7.3448	l2: 7.3452
lmid: 7.3450	l1: 7.3450	l2: 7.3452
lmid: 7.3451	l1: 7.3450	l2: 7.3451

Can also use:

- fmincon
- MMA…or any other optimizer

The MMA approach, which was initially proposed by Svanberg (see Mini Project) is based on the first-order Taylor series expansion of the objective and constraint functions.

With this method, an explicit convex subproblem is generated to approximate the implicit nonlinear problem.

Matlab code command

top(nelx, nely, volfrac, penal, **rmin**)

- nelx and nely: number of elements in the horizontal and vertical directions,
- volfrac: volume fraction,
- penal: penalization power,
- **rmin: filter size (divided by element size).**

Example 1

Numerical instability

- top(40, 20, 0.5, 3, 1.0)

effect→ Checkerboard Pattern

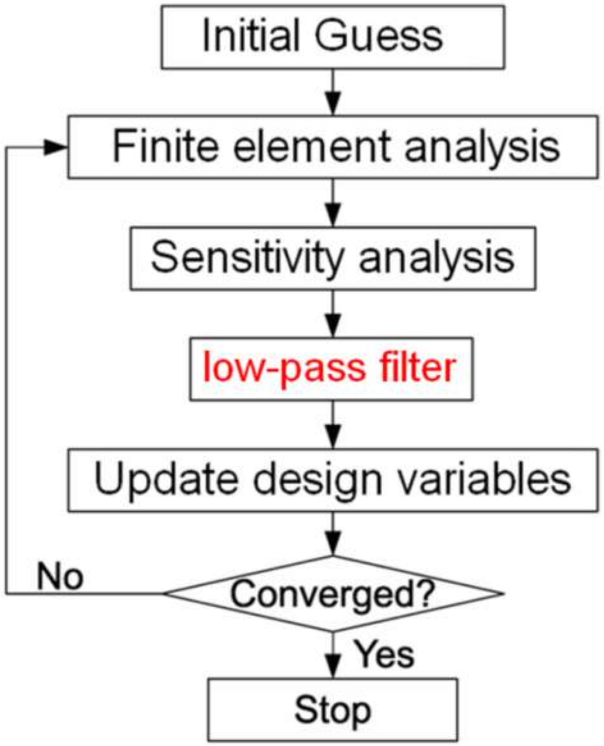
Example 1 -- Checkerboard Pattern Problem

→ Solution: LOW PASS Filter

$$\widehat{\frac{\partial c}{\partial x_e}} = \frac{1}{x_e \sum_{f=1}^{N} \hat{H}_f} \sum_{f=1}^{N} \hat{H}_f \, x_f \, \frac{\partial c}{\partial x_f}.$$

$$\hat{H}_f = r_{\min} - \operatorname{dist}(e,f),$$

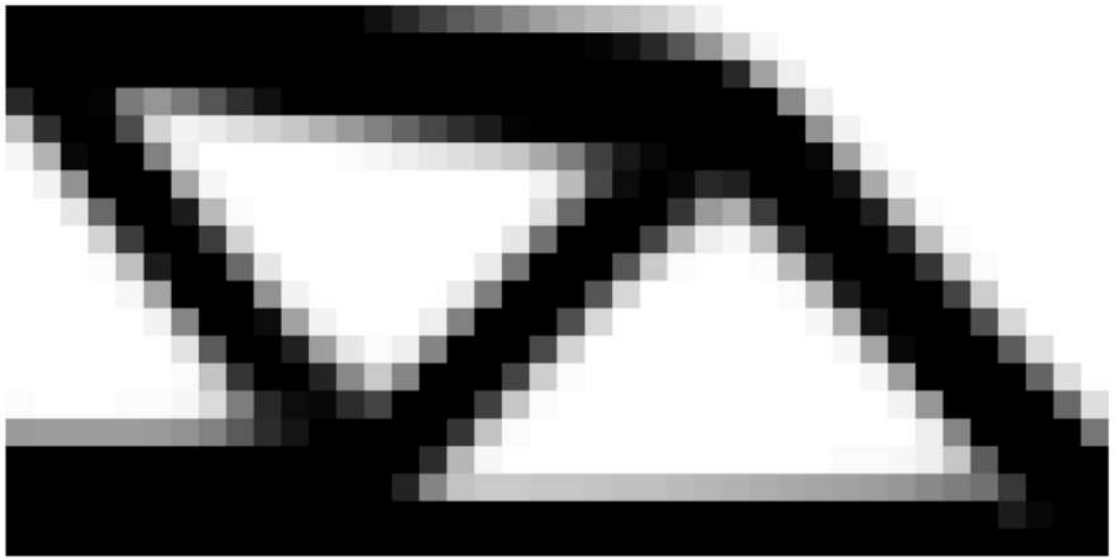
$$\{f \in N \mid \operatorname{dist}(e,f) \le r_{\min}\}, \quad e = 1,\dots,N$$




```
[dc] = check(nelx, nely, rmin, x, dc);
% FILTERING OF SENSITIVITIES
```

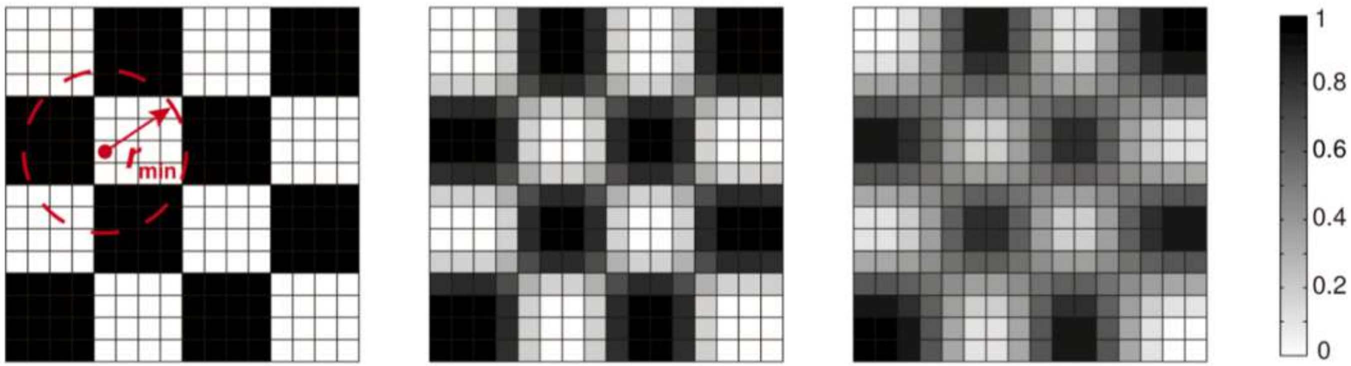
Example 2

- top(40, 20, 0.5, 3, 1.5)



- top(40, 20, 0.5, 3, 3) effect?

Filter



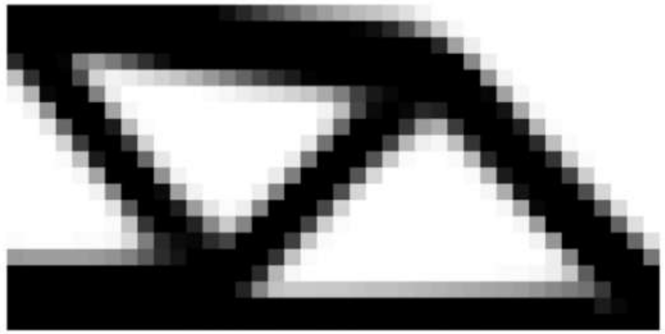
A checkerboard field and filtered fields ($r_{\min} = 1.5l_e$ and $3l_e$)

Example 3

→ Size of the filter makes it possible to obtain a more physical representation (Unblur?)

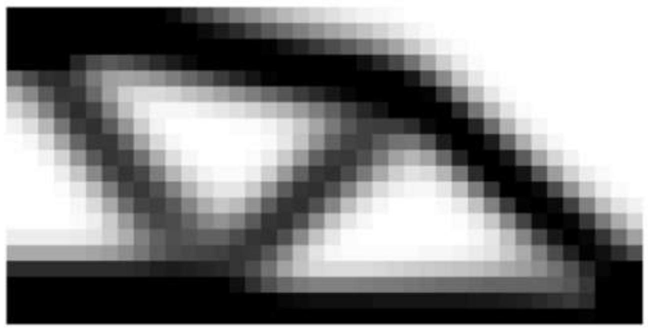
- rmin=1.5

Obj=82.7562;



- rmin=3

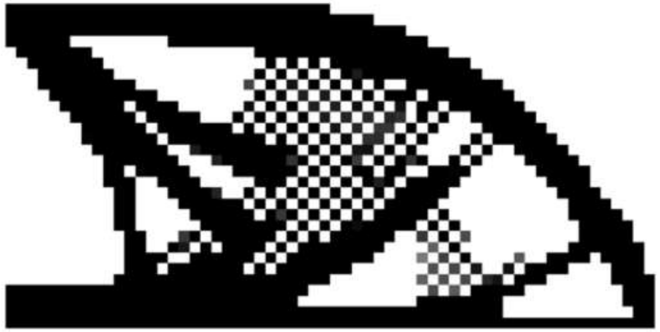
Obj=99.1929;



Example 3: Change mesh !

- top(60, 30, 0.5, 3, 1.0)

Obj: 83.0834



- top(40, 20, 0.5, 3, 1.0)

Obj: 80.4086;

Example 4: mixing

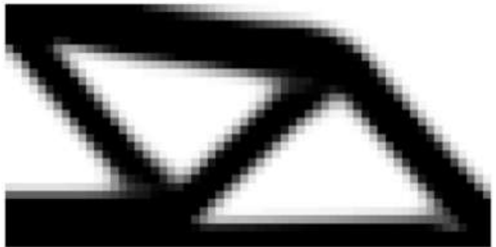
- top(60, 30, 0.5, 3, 1.5)

Obj: 81.3491

→ Filter size

- top(60, 30, 0.5, 3, 2.25)

Obj: 83.5963



→ +mesh dependency, combined eddect It's complicated

- top(40, 20, 0.5, 3, 1.5)

Obj=82.7562;

Common questions ?

- Grayness level ?

```
%% Greyness Level
   gl = 4/nele*sum(xPhys(:).*(1-xPhys(:)));
```

- How to obtain B&W for the final design ?

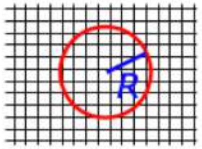
```
function xPhys=automatic_threshold(xPhys)
% %code to inputÂ xPhys and obtain the B&W design at respected volfrac

xPhys01=xPhys;
volfrac=mean(mean(xPhys))
lowlim=0;
highlim=1;
difVF=1;
while difVF>0.0001 && highlim-lowlim > 0.000001
midlim=(lowlim+highlim)/2;
xPhys01(xPhys>=midlim)=1;
xPhys01(xPhys<midlim)=0;
volfrac01=mean(mean(xPhys01));
difVF=abs(volfrac01-volfrac);
if volfrac01<volfrac
highlim=midlim;
else
lowlim=midlim;
end
end
xPhys=xPhys01;
```

Regularization by low-pass filtering

Neighborhood:

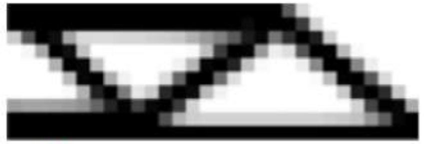
$$N_e = \{ i \mid ||\mathbf{x}_i - \mathbf{x}_e|| \leq R \}$$



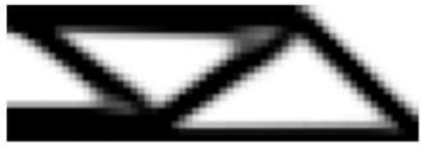
Checkerboards

Sensitivity filtering (Sigmund 1997, Sigmund&Maute 2012)

$$\frac{\widetilde{\partial \Phi}}{\partial \rho_e} = \frac{\sum_{i \in N_e} H(\mathbf{x}_i) \rho_i \frac{\partial \Phi}{\partial \rho_i}}{\rho_e \sum_{i \in N_e} H(\mathbf{x}_i)}$$



Mesh refinement



Density filtering (Bruns&Tortorelli/Bourdin 2001)

$$E_e(\boldsymbol{\rho}) = \tilde{\rho}_e^p E_0, \quad \tilde{\rho}_e = \frac{\sum_{i \in N_e} H(\mathbf{x}_i) \rho_i}{\sum_{i \in N_e} H(\mathbf{x}_i)}$$

PDE-based filtering (Lazarov&Sigmund 2011)

$$-r^2 \Delta \tilde{\rho} + \tilde{\rho} = \rho$$

Public Codes

99 Line basic Matlab (Including FE, grad's, OC)

OS, A 99 line topology optimization code written in MATLAB, *SMO,* **2001***, 21*, 120-127

88 line advanced Matlab (+advanced filters)

Andreassen, E.; Clausen, A.; Schevenels, M.; Lazarov, B. & *OS*, Efficient topology optimization in MATLAB using 88 lines of code, *SMO,* **2011**, *43*, 1-16

On multigrid-CG for efficient topology optimization

Amir, O.; Aage, N. & Lazarov, B.S., *SMO*, **2014**, *49, 815-829*

Topology optimization using PETSc:

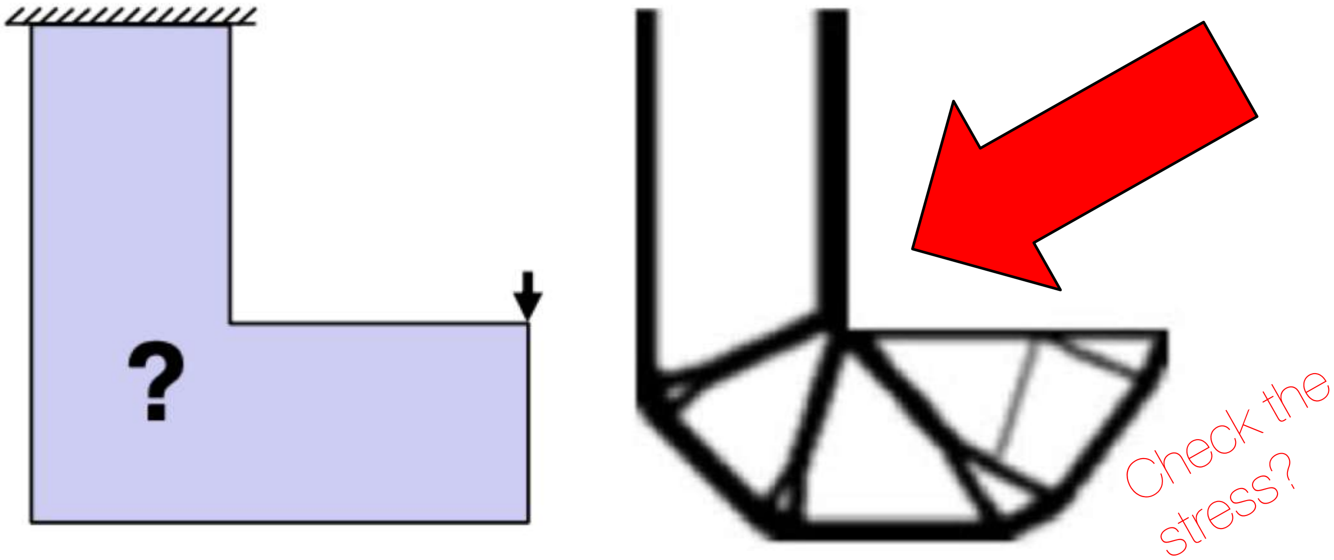
An easy-to-use, fully parallel, open-source topology optimization framework

Aage, N; Andreassen, E. & Lazarov, B.S., **2015,** *SMO, 51*, 565-572

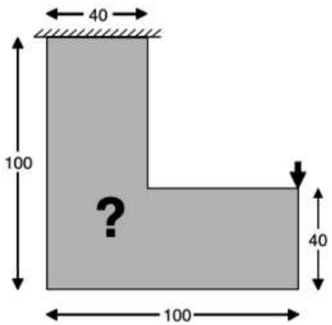
Freely downloadable from *www.topopt.dtu.dk*

At this time the structure is rigid… but feasible?

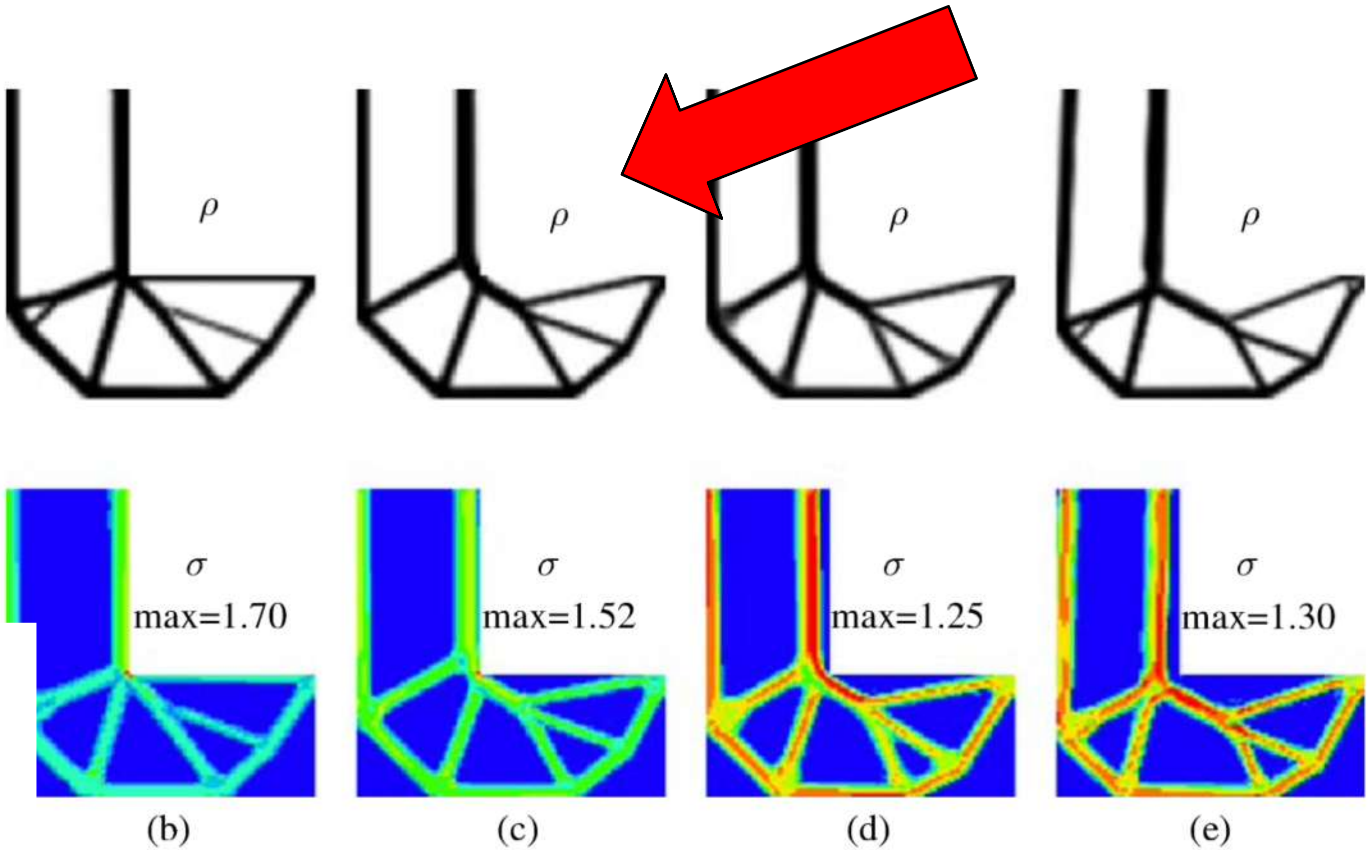
Le, C., Norato, J., Bruns, T., Ha, C., & Tortorelli, D. (2010). Stress-based topology optimization for continua. *Structural and Multidisciplinary Optimization*, *41*(4), 605-620.



Results



$$
\begin{aligned}
&Find \quad \mathbf{X} = [x_1, x_2, \cdots, x_N]^T \\
&minimize \quad f(\mathbf{X}) = W(\mathbf{X}) \\
&subject \quad \mathbf{K}(\mathbf{X})\mathbf{U}(\mathbf{X}) = \mathbf{F} \\
&\qquad \boldsymbol{\sigma}(\mathbf{X}) - [\boldsymbol{\sigma}] \leq \mathbf{0} \\
&\qquad \frac{V(\mathbf{X})}{V_0} \leq f \\
&\qquad 0 < x_{\min} \leq x_e \leq x_{\max} \leq 1
\end{aligned}
$$



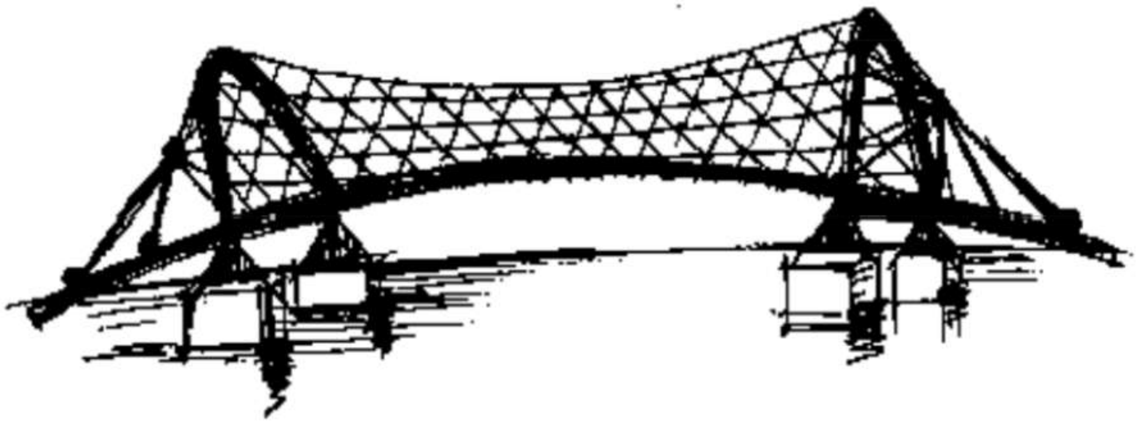
r P on the L-bracket. **a** Problem definition, **b** $P = 4$ (63 iter.), **c** $P = 6$ (72 iter.), **d** $P = 8$ (78 iter.), **e** $P =$

Is it better?

Conclusion

"The art of structure is where to put the holes"

Robert Le Ricolais
French-American engineer and philosopher
(1894-1977)

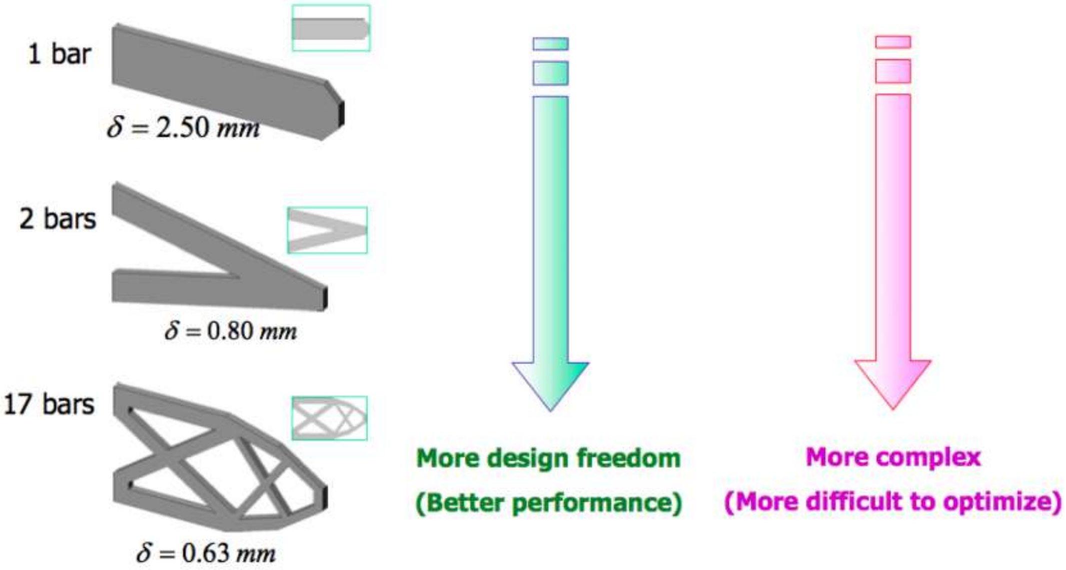


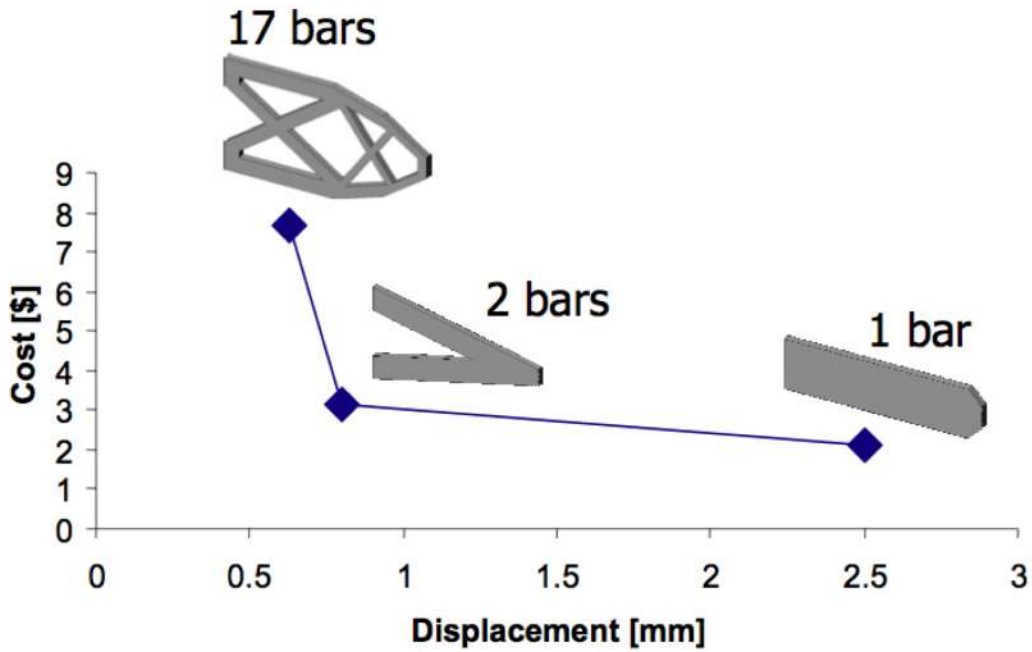
TOPOPT vs Generative Design

- https://www.autodesk.com/products/fusion-360/blog/topology-optimization-is-not-generative-design/

- **Is the outcome returned CAD ready, or does it have to be rebuilt?**
  - *A mesh model that must be rebuilt as valid geometry in a CAD system. (This is topology optimization)*
    OR
  - *CAD ready for any CAD System Geometry (This is generative design in Fusion 360)*
- **→ just wait for PART 4 and GGP framework**

WAIT A MINUTE.., this is an engineer's conclusion





The engineer or the art of compromise?

Multidisciplinary Optimization and Machine Learning for Engineering Design

19 July 2021 – 5 August 2021

https://mdoml2021.ftmd.itb.ac.id/

Jointly organized by

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Break before Part 3

Prof. J. Morlier