Proper Generalized Decomposition solutions of composite laminates parameterized with fibre orientations for fast computations

PhD thesis defense

Karim M. El-Ghamrawy

Advisors: Prof. Ferdinando Auricchio & Prof. Pedro Díez
Co-Advisor & Programme coordinator: Dr. Sergio Zlotnik

Dipartimento di Ingegneria Civile ed Architettura (DiCAr), Università di Pavia

Department d'Enginyeria Civil i Ambiental, Universitat Politècnica de Catalunya

Pavia & Barcelona, February 24th, 2021

Background and Motivation

Existence through time:

- Composite materials have been used since Mesopotemia and the Pharaonic civilisations
- Reinforced construction components for enhanced mechanical properties $\Rightarrow$ brick and straw, reinforced concrete, etc...

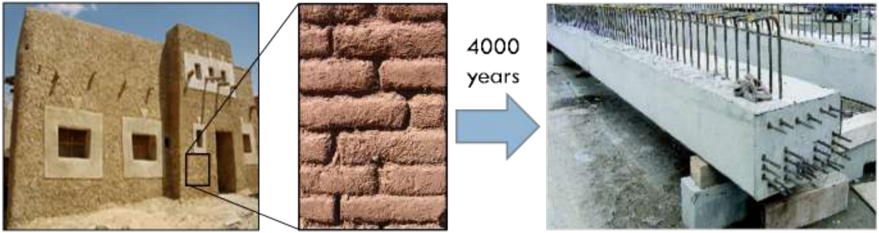
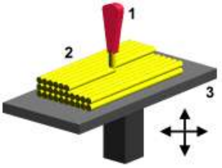


Figure: Typical ancient house in south of Egypt and a modern reinforced beam

Background and Motivation

Industrial needs:

- Enhanced mechanical properties ⇒ stiffness, load-carrying capacity, increased strength to weight ratio, etc...
- Numerical analyses for complex shapes and designs
- Ability to manufacture optimized complex designs
- Additive manufacturing deposits thermoplastic molten filament layer by layer.
- **3D printing steps:**
  CAD based 3D model ⇒ STL file ⇒ sliced layers ⇒ 3D printing ⇒ part finishing.



What are the different types of structural optimization?

Structural optimization

- **Topology optimization:** how to remove material
- **Shape optimization:** how to change the shape of the boundaries
- **Size optimization:** how to change the thicknesses of components
- **Material optimization:** how to orient material

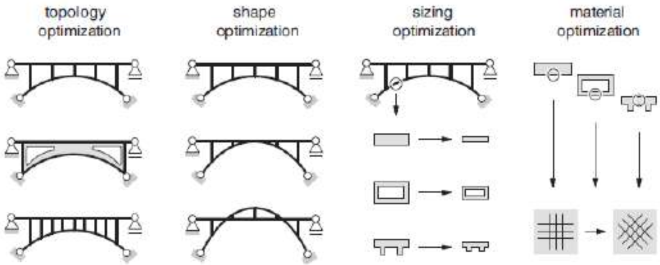
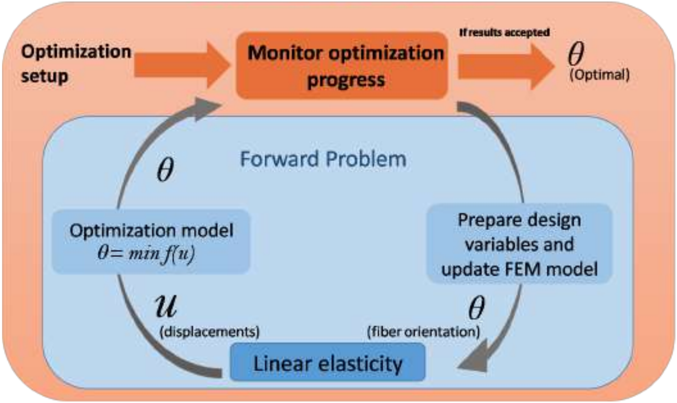


Figure: Four levels of structural optimization. Figure adapted from Ramm et al. (1998)

Optimization Problem



Optimization problem
Requires solving a large number of 3D forward models corresponding to different values of the parameters (orientation of material)

Proper Generalized Decomposition (PGD) framework

Steps of PGD

Obtain pre-computed solutions in the form of a *computational vademecum* by

1. Considering the parameters as extra-coordinates in the problem
2. Making use of a separated representation of the solution to overcome the *curse of dimensionality*

Offline phase

- Important computational resources only once
- Results in a generalized solution

Online phase

- Very fast browsing of solutions
- Availability of the solution for any value in the parametric space

Objectives

Present a new computational tool for the 3D numerical analysis of fibrous composite laminates with the goal of optimizing the orientation of fibers for 3D printing applications

Main goals

1. Apply a set of newly in-house developed tools known as *encapsulated* PGD [Díez et al. (2018, 2019)]
2. Implement a post-process algortihm to solve the optimization problem
3. Apply the methodology to a couple of numerical examples to assess the capabilities of the model
4. Enhance the model by implementing and applying a data analysis algorithm

Governing Equations: Linear Elasticity

Strong form (Voigt's notation)

- Given a 3D domain $\Omega \in \mathbb{R}^3$, find the displacement $\boldsymbol{u}(\boldsymbol{x})$ satisfying the following:

$$
\begin{aligned}
\nabla_{\mathrm{S}}^{\mathsf{T}} \boldsymbol{\sigma} + \boldsymbol{b} &= 0 && \text{in } \Omega && \text{(equilibrium)} \\
\boldsymbol{u} &= \boldsymbol{u}_D && \text{on } \Gamma_D && \text{(Dirichlet BC)} \\
\boldsymbol{n}^{\mathsf{T}} \boldsymbol{\sigma} &= \boldsymbol{t}_N && \text{on } \Gamma_N && \text{(Neumann BC)} \\
\boldsymbol{\sigma} &= \boldsymbol{\mathcal{C}} \boldsymbol{\varepsilon} && && \text{(Constitutive law)} \\
\boldsymbol{\varepsilon} &= \nabla_{\mathrm{S}} \boldsymbol{u} && &&
\end{aligned}
$$

Weak form

- The weak form is as follows, find $\boldsymbol{u} \in U$ such that

$$
\int_{\Omega} (\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w})^{\mathsf{T}} \boldsymbol{\mathcal{C}} \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{u} \, d\Omega = \int_{\Gamma_N} \boldsymbol{w}^{\mathsf{T}} \boldsymbol{t} d\Gamma + \int_{\Omega} \boldsymbol{w}^{\mathsf{T}} \boldsymbol{b} d\Omega \qquad \forall \boldsymbol{w} \in U_o
$$

Governing Equations: Linear Elasticity

Finite dimensional

- The integration over the whole domain could be a sum of integrals over the elements Ω_e.

$$\sum_{e=1}^{\mathtt{n_{el}}} \left\{ \int_{\Omega^e} (\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}_{\boldsymbol{h}}^{\boldsymbol{e}})^{\mathsf{T}} \boldsymbol{\mathcal{C}} \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{u}_{\boldsymbol{h}}^{\boldsymbol{e}} \, d\Omega - \int_{\Gamma_N^e} \boldsymbol{w}_h^{e\mathsf{T}} \boldsymbol{t} \, d\Gamma - \int_{\Omega^e} \boldsymbol{w}_h^{e\mathsf{T}} \boldsymbol{b} \, d\Omega \right\} = 0$$

- After derivations, the element stiffness matrix and force vector read:

$$\boldsymbol{K}^e = \int_{\Omega^e} \boldsymbol{B}^{e\mathsf{T}} \boldsymbol{\mathcal{C}} \boldsymbol{B}^e \, d\Omega \qquad \boldsymbol{f}^e = \int_{\Omega^e} \boldsymbol{N}^{e\mathsf{T}} \boldsymbol{b} \, d\Omega + \int_{\Gamma_N \bigcap \bar{\Omega}^e} \boldsymbol{N}^{e\mathsf{T}} \boldsymbol{t} \, d\Gamma$$

- Applying assembly operators, the global stiffness matrix and force vector read:

$$\boldsymbol{K} := \sum_{e=1}^{\mathtt{n_{el}}} \boldsymbol{L}^{e\mathsf{T}} \boldsymbol{K}^e \boldsymbol{L}^e \text{ and } \boldsymbol{f} := \sum_{e=1}^{\mathtt{n_{el}}} \boldsymbol{L}^{e\mathsf{T}} \boldsymbol{f}^e \quad \text{yielding: } \boxed{\boldsymbol{K}\boldsymbol{d} = \boldsymbol{f}}$$

Material Parameterization

- The material properties are described by $\mathcal{C}_0$ and its orientation is described by the angle θ.
- The oriented material is described as follow:

$$\boxed{\mathcal{C}(\theta) = \boldsymbol{T}^{-1}(\theta)\mathcal{C}_0\boldsymbol{T}^{-\mathsf{T}}(\theta)}$$

- Each parameter θ_i is assigned to a sub-domain Ω_i, $\Omega_i \subset \Omega$, and many $\Omega^e \subset \Omega_i$.
- The element stiffness:

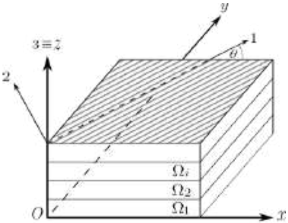
$$\boxed{\boldsymbol{K}^e(\theta_i) = \int_{\Omega^e} \boldsymbol{B}^{e\mathsf{T}}\mathcal{C}(\theta_i)\boldsymbol{B}^e d\Omega}$$

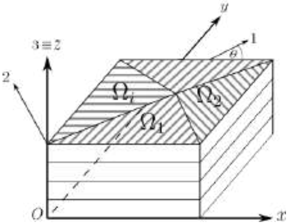
.

- The parametric linear system of equations:

$$\boxed{\boldsymbol{K}(\boldsymbol{\theta})\boldsymbol{d}(\boldsymbol{\theta}) = \boldsymbol{f}}$$

- The $\mathtt{n_p}$ parameters are gathered in vector $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_{\mathtt{n_p}}]^{\mathsf{T}}$.





Tsai-Wu Failure Criterion

- The Tsai-Wu failure index:

$$\boxed{\mathcal{I}_\mathrm{f}\left(\boldsymbol{\sigma}\right) = \boldsymbol{\sigma}^\mathsf{T}\boldsymbol{\mathcal{F}}\boldsymbol{\sigma} + \boldsymbol{\sigma}^\mathsf{T}\boldsymbol{F}}$$

- $\mathcal{I}_\mathrm{f}(\boldsymbol{\sigma}) \leq 1 \Rightarrow$ Material is safe.
- Alternative expression of the failure criterion:

$$\boxed{\mathcal{I}_\mathrm{f}\left(\bar{\boldsymbol{\sigma}}\right) = \mathcal{I}_\mathrm{f}\left(\lambda\boldsymbol{\sigma}\right) = \lambda^2\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{\mathcal{F}}\boldsymbol{\sigma} + \lambda\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{F}}$$

- The critical value of λ corresponds to the onset of failure $\mathcal{I}_\mathrm{f}\left(\bar{\boldsymbol{\sigma}}\right) = 1$.
- Assuming that $\boldsymbol{\mathcal{F}}$ is symmetric positive definite, $\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{\mathcal{F}}\boldsymbol{\sigma} \geq 0$, there is a unique positive root of the equation $\mathcal{I}_\mathrm{f}\left(\bar{\boldsymbol{\sigma}}\right) = 1$.

$$\boxed{\lambda_s = \frac{1}{2\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{\mathcal{F}}\boldsymbol{\sigma}}\left(\sqrt{(\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{F})^2 + 4\,\boldsymbol{\sigma}^\mathsf{T}\boldsymbol{\mathcal{F}}\boldsymbol{\sigma}} - \boldsymbol{\sigma}^\mathsf{T}\boldsymbol{F}\right)}$$

- The smallest positive root, denoted as λ_s, is the safety factor

Tsai-Wu Failure Criterion

- **Our goal** is to obtain expressions $\mathcal{I}_{\mathfrak{f}}(\boldsymbol{\theta})$ and $\lambda_s(\boldsymbol{\theta})$ that could be evaluated very fast
- Expressions of $\boldsymbol{\mathcal{F}}$ and $\boldsymbol{F}$ with respect to the global axes are obtained using the transformation matrices:

$$\boxed{\begin{aligned} \boldsymbol{\mathcal{F}}(\theta_i) &= \boldsymbol{T}^{\mathsf{T}}(\theta_i)\boldsymbol{\mathcal{F}}_0\boldsymbol{T}(\theta_i) \\ \boldsymbol{F}(\theta_i) &= \boldsymbol{T}^{\mathsf{T}}(\theta_i)\boldsymbol{F}_0 \end{aligned}}$$

- Marking explicitly the parametric dependence, for $\boldsymbol{x} \in \Omega_i$, the failure index $\mathcal{I}_{\mathfrak{f}}$ and the safety factor λ_s are rewritten as:

$$\boxed{\begin{aligned} \mathcal{I}_{\mathfrak{f}}\big(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})\big) &= \boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{\mathcal{F}}(\theta_i)\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta}) + \boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{F}(\theta_i) \\ \lambda_s\big(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})\big) &= \frac{\sqrt{(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{F}(\theta_i))^2 + 4\,\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{\mathcal{F}}(\theta_i)\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})} - \boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{F}(\theta_i)}{2\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})^{\mathsf{T}}\boldsymbol{\mathcal{F}}(\theta_i)\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})} \end{aligned}}$$

- The failure index $\mathcal{I}_{\mathfrak{f}}$ and the safety factor λ_s are our objective functions for the optimization problem.

Optimization problem

- The first choice is to find $\boldsymbol{\theta}$ that minimizes the maximum value of $\mathcal{I}_{\mathtt{f}}\big(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})\big)$ evaluated at all points $\boldsymbol{x}$ in Ω.

$$\boxed{\boldsymbol{\theta}_{\mathtt{f}}^{\mathtt{Opt}} = \arg\min_{\boldsymbol{\theta}} \;\max_{\boldsymbol{x}} \mathcal{I}_{\mathtt{f}}\big(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})\big)}$$

- The second choice is to find $\boldsymbol{\theta}$ that maximizes the minimum value of $\lambda_s(\boldsymbol{x},\boldsymbol{\theta})$ evaluated at all points $\boldsymbol{x}$ in Ω.

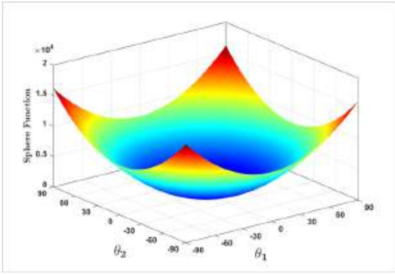
$$\boxed{\boldsymbol{\theta}_{\mathtt{s}}^{\mathtt{Opt}} = \arg\max_{\boldsymbol{\theta}} \;\min_{\boldsymbol{x}} \lambda_s(\boldsymbol{x},\boldsymbol{\theta})}$$

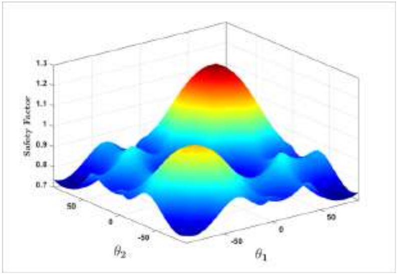
- The objective functions are not necessarily smooth and they are non convex-concave which might lead to being stuck in local minima/maxima

Optimization Algorithms

Types of algorithms

- The optimization algorithms are classified into deterministic and stochastic algorithms.
- Gradient-based methods (Newton method) converge fast but are easily stuck in local minima/maxima.
- Evolutionary methods (Genetic Algorithm) converge slow but yield a global optimal in complex problems.





PGD at a glance

1. The parameters are taken as extra coordinates stating the problem in a multidimensional framework; finding an approximation to $\boldsymbol{d}(\boldsymbol{\theta})$ in $\mathbb{R}^{\mathtt{n_d}} \times \boldsymbol{I_\theta}$.

2. The solution is sought in a separable format reducing the order of the problem

$$\mathtt{n_{full}} = \mathtt{n_d} \prod_{i=1}^{\mathtt{n_p}} n_D, i \quad \rightarrow \quad \mathtt{n_{PGD}} = \mathtt{n_d} + \sum_{i=1}^{\mathtt{n_p}} n_{D,i} \qquad \text{with } \mathtt{n_{PGD}} << \mathtt{n_{full}}$$

.

3. The PGD solver is based on a greedy strategy (computing one rank-one term at a time) and an alternating directions method to solve the nonlinear rank-one problems.

Encapsulated PGD

Separated global stiffness matrix is needed for the PGD solver!

- **Input:** the global separated stiffness matrix $\boldsymbol{K}(\boldsymbol{\theta})$.

$$\boldsymbol{K}(\boldsymbol{\theta}) \approx \boldsymbol{K}^{\mathrm{sep}}(\boldsymbol{\theta}) = \sum_{k=1}^{\mathtt{n_k}} \boldsymbol{K}^k \prod_{j=1}^{\mathtt{n_p}} \varphi_j^k(\theta_j)$$

- **Output:** the unknown vector of displacements $\boldsymbol{d}(\boldsymbol{\theta})$.

$$\boldsymbol{d}(\boldsymbol{\theta}) \approx \boldsymbol{d}_{\mathtt{PGD}}^{n}(\boldsymbol{\theta}) = \sum_{m=1}^{n} \beta^m \boldsymbol{d}^m \prod_{j=1}^{\mathtt{n_p}} G_j^m(\theta_j)$$

- *Encapsulated PGD* provides tools that directly produce computational vademecums for the high-dimensional tensor data.
- The toolbox[1] permits the performance of operations such as: solving linear system of equations, compression, addition, multiplication, division, etc...

[1] Publicly available at `https://git.lacan.upc.edu/zlotnik/algebraicPGDtools`

PGD compression

- The goal is to remove excess terms associated with redundant information from the PGD solution and increase othogonality between terms
- Least-squares projection of the PGD solution into the same approximation space:
  find a PGD-type approximation $\boldsymbol{d}_{\mathtt{com}}^{\mathtt{n_c}}$ minimizing

$$\boxed{\|\boldsymbol{d}_{\mathtt{com}}^{\mathtt{n_c}} - \boldsymbol{d}_{\mathtt{PGD}}^{n}\|_{L^2(I_{\boldsymbol{\theta}})} = \int_{I_1} \cdots \int_{I_{\mathtt{n_p}}} \left(\boldsymbol{d}_{\mathtt{com}}^{\mathtt{n_c}} - \boldsymbol{d}_{\mathtt{PGD}}^{n}\right)^2 d\theta_{\mathtt{n_p}} \ldots d\theta_1}$$

- The number of terms $\mathtt{n_c}$ in the compressed solution $\boldsymbol{d}_{\mathtt{com}}^{\mathtt{n_c}}$ is significantly lower than the original one ($\mathtt{n_c} \ll n$)

Separation of input for PGD solver

- The separated representation of $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta})$:

$$\boldsymbol{\mathcal{C}}(\theta_i) = \sum_{\ell=1}^{\mathtt{n_t}} \boldsymbol{\mathcal{C}}^{\ell} \prod_{j=1}^{\mathtt{n_p}} \phi_j^{\ell,i}(\theta_j) \qquad \phi_j^{\ell,i}(\theta_j) \equiv 1 \text{ for } j \neq i$$

- The element stiffness, $\Omega_e \in \Omega_i$, yields:

$$\boldsymbol{K}^e(\theta_i) = \sum_{\ell=1}^{\mathtt{n_t}} \Big[\int_{\Omega_e} \boldsymbol{B}^{e\mathsf{T}} \boldsymbol{\mathcal{C}}^{\ell} \boldsymbol{B}^e d\Omega \Big] \prod_{j=1}^{\mathtt{n_p}} \phi_j^{\ell,i}(\theta_j)$$

- Assembling the global stiffness matrix yields:

$$\begin{aligned} \boldsymbol{K}(\theta_1, \theta_2, ..., \theta_{\mathtt{n_p}}) &= \sum_{e=1}^{\mathtt{n_{el}}} \boldsymbol{L}^{e\mathsf{T}} \boldsymbol{K}^e(\theta_i) \boldsymbol{L}^e \\ &= \sum_{e=1}^{\mathtt{n_{el}}} \sum_{\ell=1}^{\mathtt{n_t}} \Big[\int_{\Omega_e} \boldsymbol{L}^{e\mathsf{T}} \boldsymbol{B}^{e\mathsf{T}} \boldsymbol{\mathcal{C}}^{\ell} \boldsymbol{B}^e \boldsymbol{L}^e \, d\Omega \Big] \prod_{j=1}^{\mathtt{n_p}} \phi_j^{\ell,i}(\theta_j) \end{aligned}$$

Post-process and Sensitivities

- PGD gives the displacement $\boldsymbol{d}(\boldsymbol{\theta}) = \sum_{m=1}^{n} \beta^m \boldsymbol{d}^m \prod_{j=1}^{\mathtt{n_p}} G_j^m(\theta_j)$
- The parametric strain tensor is a linear output of the overall displacements $\boldsymbol{d}$
- The parametric stress tensor $\Rightarrow \boldsymbol{\sigma}_g(\boldsymbol{\theta}) = \boldsymbol{\mathcal{C}}(\theta_i)\boldsymbol{\varepsilon}_g(\boldsymbol{\theta})$ with $\boldsymbol{\varepsilon}_g^m = \boldsymbol{B}_g^e \boldsymbol{L}^e \boldsymbol{d}^m$

$$\boxed{\boldsymbol{\sigma}_g(\boldsymbol{\theta}) = \sum_{m=1}^{n} \sum_{\ell=1}^{\mathtt{n_t}} \beta^m \boldsymbol{C}^\ell \boldsymbol{\varepsilon}_g^m \prod_{j=1}^{\mathtt{n_p}} \phi_j^{\ell,i}(\theta_j) G_j^m(\theta_j)}$$

- Using the parametric stress tensor and the transformed strength tensors, the failure index could be reconstructed:

$$\boxed{\mathcal{I}_{\mathtt{L}}\big(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\big) = \boldsymbol{\sigma}_g^{\mathsf{T}} \boldsymbol{F}(\theta_i) \text{ and } \mathcal{I}_{\mathtt{Q}}\big(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\big) = \boldsymbol{\sigma}_g^{\mathsf{T}} \boldsymbol{\mathcal{F}}(\theta_i) \boldsymbol{\sigma}_g}$$

Failure Vademecums

- The expressions for the quadratic and linear terms:

$$\mathcal{I}_{\mathtt{Q}}\left(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\right) = \sum_{b=1}^{\mathtt{n_Q}} \tilde{\gamma}^b \tilde{A}_g^b \prod_{j=1}^{\mathtt{n_p}} \tilde{H}_j^{b,i}(\theta_j) \text{ and } \mathcal{I}_{\mathtt{L}}\left(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\right) = \sum_{v=1}^{\mathtt{n_L}} \hat{\gamma}^v \hat{A}_g^v \prod_{j=1}^{\mathtt{n_p}} \hat{H}_j^{v,i}(\theta_j)$$

- The final expression for the failure index $\mathcal{I}_{\mathtt{f}}$ is readily recovered by summing up $\mathcal{I}_{\mathtt{Q}}$ and $\mathcal{I}_{\mathtt{L}}$

$$\mathcal{I}_{\mathtt{f}}\left(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\right) = \mathcal{I}_{\mathtt{Q}}\left(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\right) + \mathcal{I}_{\mathtt{L}}\left(\boldsymbol{\sigma}_g(\boldsymbol{\theta})\right) = \sum_{f=1}^{\mathtt{n_Q}+\mathtt{n_L}} \gamma^f A_g^f \prod_{j=1}^{\mathtt{n_p}} H_j^{f,i}(\theta_j)$$

- The quantities γ^f, A_g^f and $H_j^{f,i}(\theta_j)$ depend on the index f

$$\gamma^f,\, A_g^f,\, H_g^f = \begin{cases} \tilde{\gamma}^f, \tilde{A}_g^f, \tilde{H}_g^f & \text{if} f \leq \mathtt{n_Q} \\ \hat{\gamma}^{f-\mathtt{n_Q}}, \hat{A}_g^{f-\mathtt{n_Q}}, \hat{H}_g^{f-\mathtt{n_Q}} & \text{if } f > \mathtt{n_Q} \end{cases}$$

Failure Index $\mathcal{I}_{\mathtt{f}}$ Sensitivities

- The gradient of the failure index $\nabla_\theta \mathcal{I}_{\mathtt{f}}(\boldsymbol{\theta})$:

$$\frac{\partial \mathcal{I}_{\mathtt{f}}(\boldsymbol{\theta})}{\partial \theta_k} = \sum_{f=1}^{\mathtt{n_Q}+\mathtt{n_L}} \gamma^f A_g^f \left[\frac{dH_k^{f,i}}{d\theta_k}(\theta_k) \right] \prod_{j \neq k}^{\mathtt{n_P}} H_j^{f,i}(\theta_j)$$

- For optimization methods requiring the Hessian matrix, for $k \neq \tilde{k}$

$$\frac{\partial^2 \mathcal{I}_{\mathtt{f}}(\boldsymbol{\theta})}{\partial \theta_k \partial \theta_{\tilde{k}}} = \sum_{f=1}^{\mathtt{n_Q}+\mathtt{n_L}} \gamma^f A_g^f \left[\frac{dH_k^{f,i}}{d\theta_k}(\theta_k) \frac{dH_{\tilde{k}}^{f,i}}{d\theta_{\tilde{k}}}(\theta_{\tilde{k}}) \right] \prod_{j \neq k, \tilde{k}}^{\mathtt{n_P}} H_j^{f,i}(\theta_j)$$

- And for the diagonal terms

$$\frac{\partial^2 \mathcal{I}_{\mathtt{f}}(\boldsymbol{\theta})}{\partial {\theta_k}^2} = \sum_{f=1}^{\mathtt{n_Q}+\mathtt{n_L}} \gamma^f A_g^f \left[\frac{d^2 H_k^{f,i}}{d{\theta_k}^2}(\theta_k) \right] \prod_{j \neq k}^{\mathtt{n_P}} H_j^{f,i}(\theta_j)$$

Safety Factor λ_s Sensitivities

- The gradient of the safety factor $\nabla_\theta \lambda_s(\boldsymbol{\theta})$:

$$\frac{\partial \mathcal{I}_{\mathtt{Q}}(\boldsymbol{\theta})}{\partial \theta_k} = \sum_{b=1}^{\mathtt{n_Q}} \tilde{\gamma}^b \tilde{A}_g^f \left[\frac{d\tilde{H}_k^{b,i}}{d\theta_k}(\theta_k) \right] \prod_{j \neq k}^{\mathtt{n_P}} \tilde{H}_j^{b,i}(\theta_j)$$

$$\frac{\partial \mathcal{I}_{\mathtt{L}}(\boldsymbol{\theta})}{\partial \theta_k} = \sum_{v=1}^{\mathtt{n_L}} \hat{\gamma}^v \hat{A}_g^f \left[\frac{d\hat{H}_k^{v,i}}{d\theta_k}(\theta_k) \right] \prod_{j \neq k}^{\mathtt{n_P}} \hat{H}_j^{v,i}(\theta_j)$$

- Recalling the safety factor expression, and applying the quotient rule for derivatives of divisions

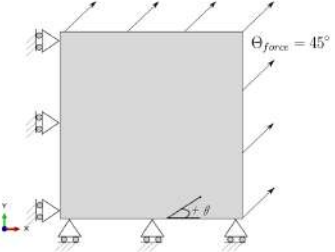
$$\lambda_s\big(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{\theta})\big) = \frac{-\mathcal{I}_{\mathtt{L}} + \sqrt{\mathcal{I}_{\mathtt{L}}^2 + 4\,\mathcal{I}_{\mathtt{Q}}}}{2\mathcal{I}_{\mathtt{Q}}}$$

$$\frac{\partial \lambda_s(\boldsymbol{\theta})}{\partial \theta_k} =$$

$$\frac{\mathcal{I}_{\mathtt{Q}} \left[-\frac{\partial \mathcal{I}_{\mathtt{L}}(\boldsymbol{\theta})}{\partial \theta_k} + 0.5(\mathcal{I}_{\mathtt{L}}^2 + 4\mathcal{I}_{\mathtt{Q}})^{-1/2} \cdot \left(2\mathcal{I}_{\mathtt{L}} \frac{\partial \mathcal{I}_{\mathtt{L}}(\boldsymbol{\theta})}{\partial \theta_k} + 4 \frac{\partial \mathcal{I}_{\mathtt{Q}}(\boldsymbol{\theta})}{\partial \theta_k} \right) \right] - \frac{\partial \mathcal{I}_{\mathtt{Q}}(\boldsymbol{\theta})}{\partial \theta_k} \left[-\mathcal{I}_{\mathtt{L}} + \sqrt{\mathcal{I}_{\mathtt{L}}^2 + 4\mathcal{I}_{\mathtt{Q}}} \right]}{2\mathcal{I}_{\mathtt{Q}}^2}$$

Plate under tensile load: Description

- **Plate dimensions:** $60 \times 60 \times 6\ \mathrm{mm}^3$.
- **Type and number of elements:** Serendipity 800 elements.
- **Parameters range:** $\theta_1 \in I_1 = [-90°, 90°]$ $\theta_2 \in I_2 = [-90°, 90°]$.
- **Parametric mesh:** 181 nodes.
- **Material:** Carbon Fibre ABS.



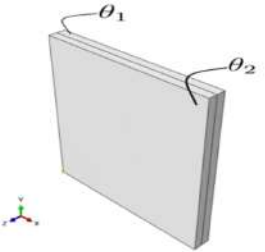
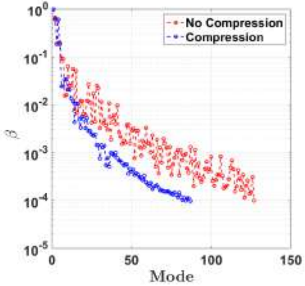


Plate under tensile load: PGD performance

- The stopping criterion for computing terms is controlled by $\xi = \frac{\beta^m}{\beta^1}$.
- The number of modes is reduced by 31.5% in the compressed solution.
- The reltive error between FE and PGD is 0.1%

$$\varepsilon_{glob} = \frac{\|\boldsymbol{d}_{\text{PGD}} - \boldsymbol{d}_{\text{FE}}\|_{\Omega \times I_1 \times \ldots \times I_{\text{n}_\text{p}}}}{\|\boldsymbol{d}_{\text{FE}}\|_{\Omega \times I_1 \times \ldots \times I_{\text{n}_\text{p}}}}$$



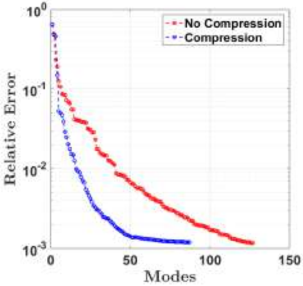
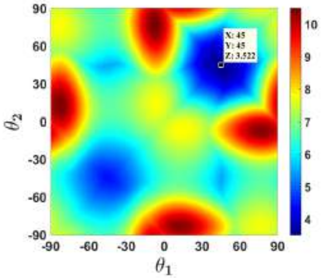
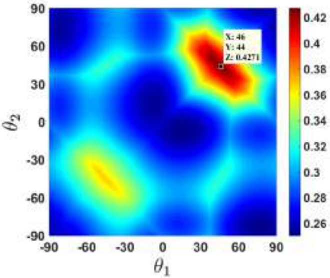


Plate under tensile load: Optimization output

- Maps represent the objective functions in the parametric space.
- The optimal $(\theta_1, \theta_2) = (45°, 45°)$.



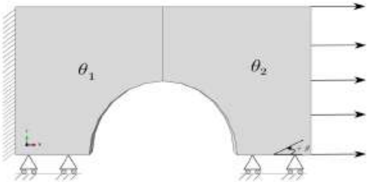


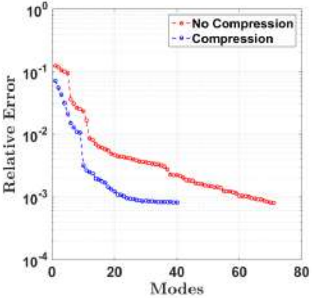
CPU time

- The CPU time for the FE whole solution is $\sim \mathbf{6.5}$ days with 32761 FE solves.
- The CPU time for the offline PGD solution is $\sim \mathbf{2.5}$ hours and the online browsing is in seconds.

Plate with circular hole under tensile load: Two parameters

- No more symmetry in the optimal solution due to patches of elements.
- The optimal solution is ambiguous due to the hole existence.
- The compression yields a reduction in the number of modes 43.5%.





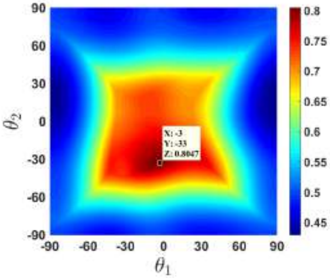
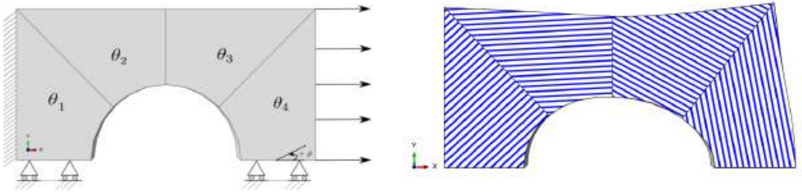


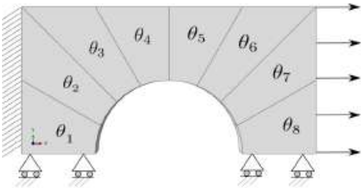
Plate with circular hole under tensile load: Four parameters

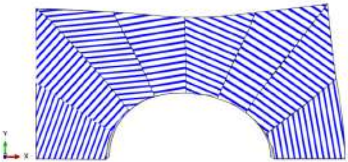


Optimal Angles		
	ga function	`fmincon` function
θ_1	$42°$	$42.062°$
θ_2	$3°$	$2.9944°$
θ_3	$-22°$	$-22.4586°$
θ_4	$-83°$	$-84.1544°$
Index value	$\max(\min \lambda_s) = 0.8254$	$\max(\min \lambda_s) = 0.8255$
CPU time	~ 40 min	~ 1 min

Table: Optimized angles for square plate with circular hole using the safety factor as objective function.

Plate with circular hole under tensile load: Eight parameters



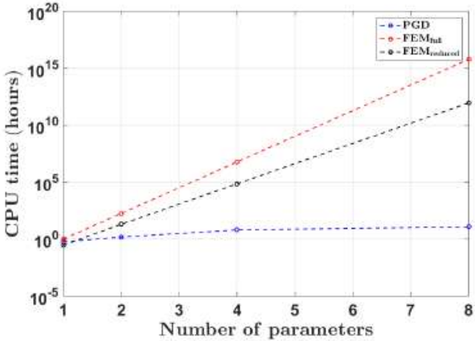


Optimized angles of the safety factor using GA				
# of GA evaluations	1000	10,000	100,000	1,000,000
θ_1	$81°$	$86°$	$70°$	$84°$
θ_2	$42°$	$43°$	$42°$	$41°$
θ_3	$5°$	$6°$	$6°$	$4°$
θ_4	$-6°$	$8°$	$8°$	$-20°$
θ_5	$-20°$	$-21°$	$-23°$	$17°$
θ_6	$-24°$	$-25°$	$-26°$	$-51°$
θ_7	$-26°$	$-25°$	$-30°$	$-61°$
θ_8	$45°$	$-86°$	$-85°$	$-82°$
$\max(\lambda_s)$	0.8249	0.8803	0.879	0.8501
CPU time	$\sim$ 1.2 min	$\sim$ 12 min	$\sim$ 120 min	$\sim$ 1300 min

Table: Different number of evaluations yielding different GA precision

Plate with circular hole under tensile load: Computational cost

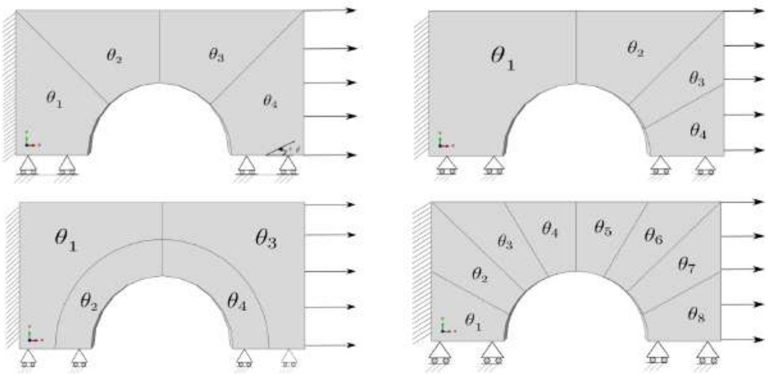
- **Four parameters:** PGD provides a solution in ~ 30 hours while computing the standard FE solution at every parametric point would take $\sim 10^6$ hours.
- **Eight parameters:** PGD provides a solution in ~ 42 hours while computing the standard FE solution at every parametric point would take $\sim 10^{16}$ hours.



The computational cost is drastically reduced when using PGD!

Domain decomposition strategy: Introduction

- Changing the partitioning patterns and increasing the number of partitions affects the optimal fibre orientation results.
- Increasing the number of subdomains does not guarantee fast convergence



We need to find a smarter partitioning strategy!!!

Domain decomposition strategy: Clustering algorithm steps

- Clustering techniques are unsupervised learning techniques such as: K-means, hierarchical clustering.
- Clustering aims to group elements having similar features in a data set into coherent groups.
- The clustering strategy is applied as a preprocess before solving the mechanical problem using PGD.

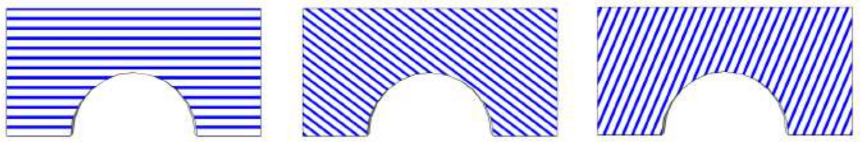
Clustering techniques for efficient partitioning of the domain

1. **Preanalyses:** snapshots of the system at each finite element for different orientations are taken and stored.
2. **Principal Component Analysis:** responsible for the data transformation from correlated fields to uncorrelated new components.
3. **Clustering of factors and their intersection:** the clustering techniques are applied to the factors (components) obtained from PCA.
4. **Error computation and clustering optimization:** clustering optimization in order to find the best clusters representing the data.

Domain decomposition strategy: Clustering algorithm steps

Preanalyses

- Collecting as much data as possible.
- The data represents a quantity of interest taken at different fibres configurations in each FE.
- More data $\Rightarrow$ accurate results.
- We assume a unidirectional laminate in each single snapshot.
- The quantity of interest is the safety factor at each element.
- The data is stored, in the $\mathtt{n_{el}} \times \mathtt{N_c}$ matrix $\tilde{\boldsymbol{\lambda}}_s$, to be manipulated and analyzed.



Domain decomposition strategy: Clustering algorithm steps

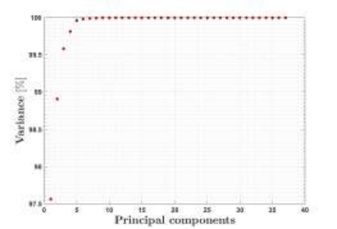
Principal Component Analysis (PCA)

- PCA reduces the dimensionality of the data while maintaining its variance as high as possible.
- First we find the covariance matrix

$$\boxed{\boldsymbol{\Sigma} = \frac{1}{\mathtt{n_{el}}} \tilde{\boldsymbol{\lambda}}_s{}^{\mathsf{T}} \tilde{\boldsymbol{\lambda}}_s}$$

- We then solve for the eigenvalues λ_i and the eigenvectors $\boldsymbol{v}^i$
- The factors or principal components are defined

$$\boxed{\mathbf{f}^i = \tilde{\boldsymbol{\lambda}}_s \boldsymbol{v}^i}$$

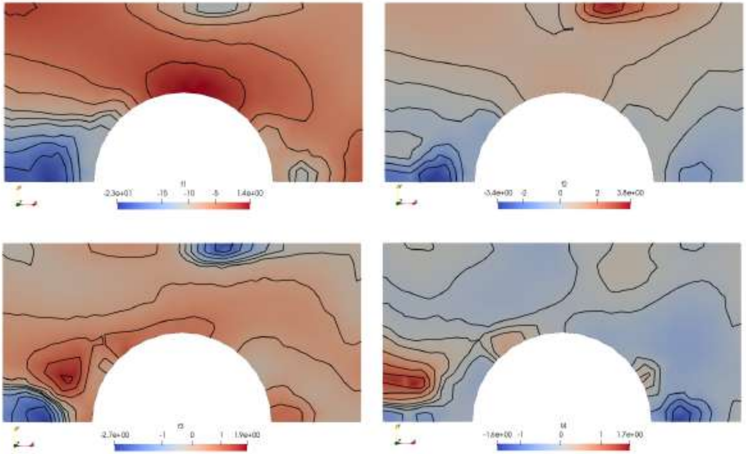


- The eigenvalues expressed in terms of the obtained factors

$$\boxed{\lambda_i = \frac{1}{\mathtt{n_{el}}} \sum_{j=1}^{\mathtt{n_{el}}} (\mathrm{f}_j^i - \hat{\mathrm{f}}^i)^2 \qquad \text{with } \lambda_1 > \lambda_2 > \ldots, > \lambda_{\mathtt{N_c}}}$$

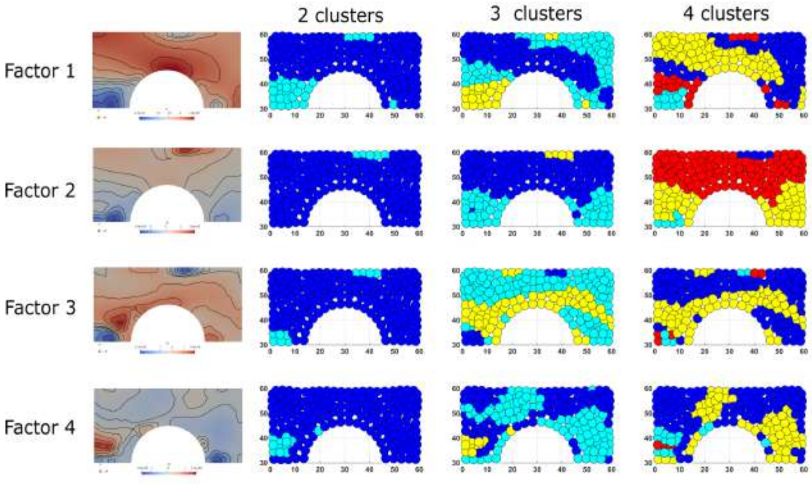
Domain decomposition strategy: Clustering algorithm steps

First four factors from PCA:
Factors based on the safety factor data $\tilde{\lambda}_s$ that will be clustered using the K-means algorithm



Domain decomposition strategy: Clustering of factors

K-means clustering of four factors



Domain decomposition strategy: Error computation

- **Our goal**: find the best partition $\mathcal{P}$
- The optimization objective is to minimize the error measure called Sum of Squares Error (SSE) [Alaimo et al. (2019)].
- SSE is a measure of discrepancy between the data of an element and the average of the data in the cluster where the element belongs.

$$E(\mathcal{P}) = \sum_{s=1}^{\mathtt{N_c}} E^s(\mathcal{P}) = \frac{1}{E_{max}} \sum_{s=1}^{\mathtt{N_c}} \sum_{\ell=1}^{\mathtt{n_s}} \sum_{i=1}^{n_\ell(\mathcal{P})} (\omega_i^s - \overline{\omega_{,\ell}^s})^2$$

$$E_{max} = \sum_{s=1}^{\mathtt{N_c}} \sum_{i=1}^{\mathtt{n_{el}}} (\omega_i^s - \overline{\omega^s})^2$$

- Each finite element is a cluster on its own $\mathtt{n_s} = \mathtt{n_{el}} \Rightarrow E(\mathcal{P}) = 0\%$
- Partition $\mathcal{P}$ consists of only one cluster $\mathtt{n_s} = 1 \Rightarrow E(\mathcal{P}) = 100\%$

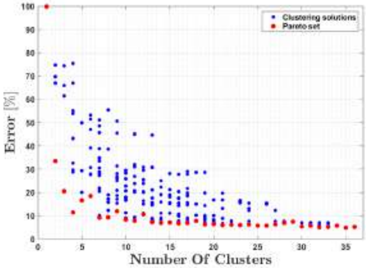
Domain decomposition strategy: Clustering optimization

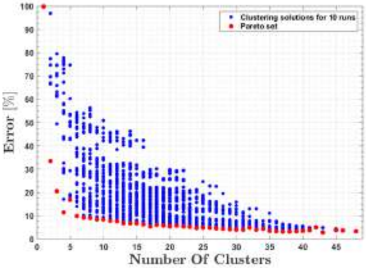
- The clustering multi-objective problem is defined as

$$\mathcal{P}^{\mathtt{Opt}} = \arg\min_{\mathcal{P}} \ \{E(\mathcal{P}), \mathtt{n}_{\mathtt{s}}(\mathcal{P})\} \quad \text{s.t.} \quad \mathcal{P} \in \mathbb{P}$$

- There exists a *Pareto optimality situation*.
- A *Pareto set* has optimization solutions that are superior to the rest of the solutions in the search space $\mathbb{P}$.
- The solutions among the set do not dominate each other.
- A partition $\mathcal{P}_1$ is said to dominate another partition $\mathcal{P}_2$ only when the following inequalities hold

$$\begin{array}{ccc} E(\mathcal{P}_1) \leq E(\mathcal{P}_2) & \text{and} & \mathtt{n}_{\mathtt{s}}(\mathcal{P}_1) \leq \mathtt{n}_{\mathtt{s}}(\mathcal{P}_2) \\ E(\mathcal{P}_1) < E(\mathcal{P}_2) & \text{or} & \mathtt{n}_{\mathtt{s}}(\mathcal{P}_1) < \mathtt{n}_{\mathtt{s}}(\mathcal{P}_2) \end{array}$$



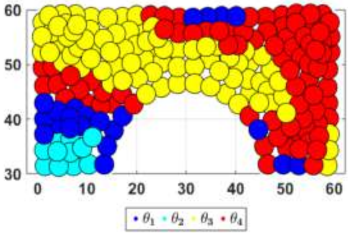


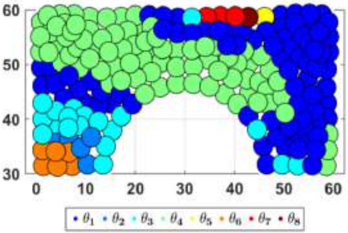
Pareto set error comparison				
	Stress based clusters		Safety factor based clusters	
	4 clusters	8 clusters	4 clusters	8 clusters
K-means single run	38%	29%	11.5%	9.5%
K-means 10 runs	37%	29%	11.2%	9.1%
Ward's method	41%	26%	11.9%	9.3%

Table: Pareto set error comparison between K-means with a single run, K-means with 10 runs, and Ward's method

Domain decomposition strategy: Clustered domains

Optimal domain decomposition obtained from the clustering algorithm





	Domain with 4 parameters	Domain with 8 parameters
Stress based clustering with K-means	0.7863	0.8788
Stress based clustering with Ward's	0.8653	0.9037
Transformed safety factor clustering with K-means	**1.013**	**0.9934**
Transformed safety factor clustering with Ward's	0.7973	0.8244
Based on intuition	0.8254	0.879

Table: Safety factor index λ_s obtained from PGD based on different domain parameterization

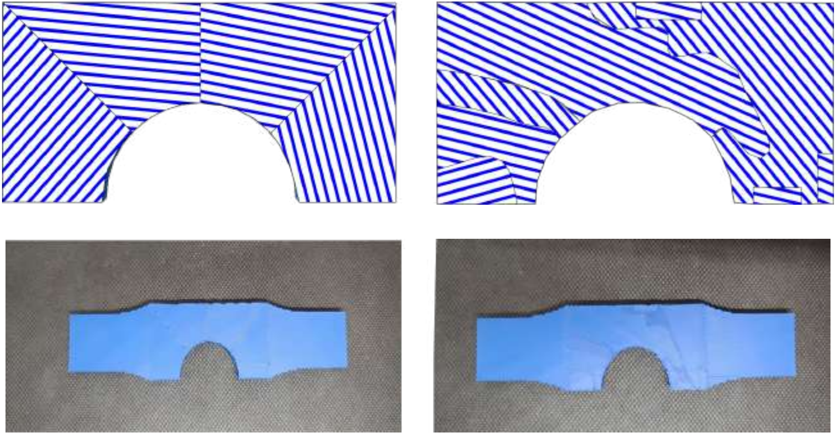
Domain decomposition strategy: Experimental testing

Experimental testing for the validation of the model using 3D printing

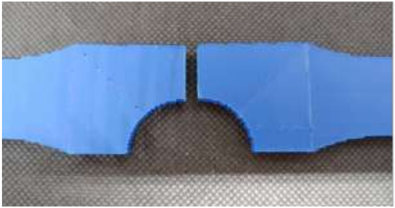
1. **Simulation and analysis:** Run the model to obtain optimal fibre orientation in different domains.
2. **Specimen preparation and 3D printing:** The preparation of the STL files of the components to be printed and slicing the part for the G-Code generation.
3. **Tensile test and monitor results:** Perform traction on the part until failure occurs and then record the corresponding load for comparison.

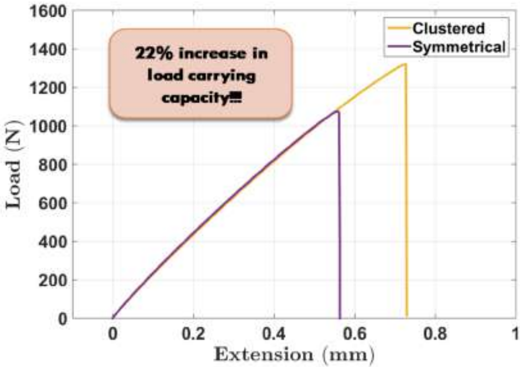
Domain decomposition strategy: Experimental tests

Optimal fibre orientation in optimized partitions in the domain



Domain decomposition strategy: Experimental tests





Conclusions

- PGD reduces the computational cost significantly.
- Applying the encapsulated PGD concept facilitates the manipulation of high-dimensional data.
- Using PGD in optimization problems is extremely efficient since we have the whole space of solutions available.
- Applying clustering techniques as a pre-process leads to better optimization results.
- The whole methodology opens the door for customized mechanical components.
- Experimental tests show the improvement in the load carrying capacity of the optimized 3D printed components
- The clustering techniques approach reduces the CPU time for the PGD.

Future work

- **Enhancement of the model:**
  From the PGD point of view, it is important to enhance the vademecum by including geometrical parameterization, load location parameter, and boundary conditions parameterization.
- **Programming languages:**
  From the programming point of view, the PGD package could be implemented using high-efficiency languages such as C/C++ and/or FORTRAN. Moreover, modern simulation applications on smartphones could be developed to make use of the fast response of the PGD vademecums.
- **Error estimation:**
  Obtaining an error estimator of a quantity of interest to be able to accurately choose the stopping criterion for the greedy algorithm in the PGD.
- **Additive manufacturing:**
  The need to explore the possibility of printing continuous fibres with the aim of enhancing the mechanical properties of 3D printed components by avoiding jumps between partitions.

References

Alaimo, G., Auricchio, F., Marfia, S., and Sacco, E. (2019). Optimization clustering technique for PieceWise Uniform Transformation Field Analysis homogenization of viscoplastic composites. *Computational Mechanics*, 64(6):1495–1516.

Díez, P., Zlotnik, S., García-González, A., and Huerta, A. (2018). Algebraic PGD for tensor separation and compression: An algorithmic approach. *Comptes Rendus - Mecanique*, 346(7):501–514.

Díez, P., Zlotnik, S., García-González, A., and Huerta, A. (2019). Encapsulated PGD Algebraic Toolbox Operating with High-Dimensional Data. *Archives of Computational Methods in Engineering*, 26(5).

Ramm, E., Maute, K., and Schwarz, S. (1998). Conceptual design by structural optimization. *Proceedings of the Euro–C 1998 Conference on Computational Modelling of Concrete Structures, herausgegeben von R. de Borst, N. Bicanic, H. Mang & G. Meschke, S*, pages 879–896.


THANK YOU
TEŞEKKÜR
הדות
감사합니다
GRACIAS
ĎAKUJEM
DĚKUJI
DANKE
спасибо
ХВАЛА
TERIMA KASIH.
NGIYABONGA
הדות
KIITOS
DANKE
ARIGATÔ
OBRIGADO
TEŞEKKÜR
GRACIAS
DHANYAVĀD
TAK
감사합니다
DHANYAVĀD
KÖSZÖNÖM
ευχαριστώ
ARIGATÔ
спасибо
MERCI
GRÀCIES
THANK YOU
SHUKRAN
XIÈXIÈ
DĚKUJI
TAK
ευχαριστώ
DĚKUJI
MERCI
GRAZIE
OBRIGADO
XIÈXIÈ
DĚKUJI
הדות
DANKE
TAK
GRACIAS
GRÀCIES
KIITOS
감사합니다
TERIMA KASIH.
THANK YOU
TERIMA KASIH.
KOP KHUN
DANKE
KOP KHUN
DHANYAVĀD
TACK
KIITOS
HVALA
감사합니다
OBRIGADO
MERCI
GRAZIE
TEŞEKKÜR
KÖSZÖNÖM
DANKE
TERIMA KASIH.
ХВАЛА
THANK YOU
ARIGATÔ
XIÈXIÈ
ĎAKUJEM
KOP KHUN
감사합니다
THANK YOU
MERCI
GRACIAS
GRAZIE
ХВАЛА
NGIYABONGA
SHUKRAN
TACK