Contents

Topology optimization and material optimization

LiDO team

March 24, 2023

1 Introduction

This document presents many thoughts about computing sensitivities for the LiDO 2.0 project.

2 Preliminaries

The derivative $f'(x)$ of a function $f : \mathcal{R} \to \mathcal{R}$, at $x \in \mathcal{R}$, if it exists, is defined as

$$f'(x) = \frac{df}{dx}(x) = \lim_{\epsilon \to 0} \frac{1}{\epsilon}(f(x+\epsilon) - f(x)) \tag{1}$$

This result is well known. Note that the result must be independent of the direction in which $\epsilon \to 0$.

The above definition generalizes to vector valued functions of vectors. The derivative $D\mathbf{f}(\mathbf{x})$ of a function $\mathbf{f} : \mathcal{R}^n \to \mathcal{R}^m$, at $\mathbf{x} \in \mathcal{R}^n$, if it exists, is the linear operator, i.e. $D\mathbf{f}(\mathbf{x}) \in \mathrm{Lin}(\mathcal{R}^m, \mathcal{R}^n)$ defined such that

$$D\mathbf{f}(\mathbf{x})\,\mathbf{s} = \lim_{\epsilon \to 0} \frac{1}{\epsilon}(\mathbf{f}(\mathbf{x}+\epsilon\,\mathbf{s}) - \mathbf{f}(\mathbf{x})) \tag{2}$$

for all $\mathbf{s} \in \mathcal{R}^n$.

When we have functions of functions, i.e. functionals, the above definitions break down. However, motivated by them we define the variation $f(u; \delta u)$ of the function $f \in \mathcal{V}$ at $u \in \mathcal{U}$ in the direction $\delta u \in \mathcal{U}$, if it exists, such that

$$\begin{aligned} \delta f(u; \delta u) &= \lim_{\epsilon \to 0} \frac{1}{\epsilon}(f(u + \epsilon\,\delta u) - f(u)) \\ &= \left.\frac{d}{d\,\epsilon} f(u + \epsilon\,\delta u)\right|_{\epsilon=0} \end{aligned} \tag{3}$$

Note that $\delta f(u; \delta u)$ operates linearly on δu.

In the following we obtain variations of functionals with respect to *fields*, i.e. functions of position. After FEM discretization, the variations are expressed as derivatives.

3 Discrete vs. continuous sensitivity analysis

For optimization we require the sensitivities of the computed quantities of interest (QoIs) to evaluate search directions for the nonlinear programming algorithm. The sensitivities can be computed via finite difference method, e.g. from (1) we have $f'(x) \approx 1/\epsilon\,(f(x+\epsilon) - f(x))$. This approximation breaks down if ϵ is too small or too large due to round-off and truncation error. But for a suitable range of ϵ values, it provides adequate results. Unfortunately, for our PDE constrained optimization problems, we need to evaluate the perturbed value $f(x+\epsilon)$ for each design parameter x and each evaluation requires an expensive FEA solve. For this reason we implement the adjoint sensitivity analysis discussed below.

There is much discussion in the literature about discrete vs. continuous sensitivity analysis. In the both methods, the PDEs whose solutions yield the responses, e.g. displacement, and QoIs that are used for the optimization cost and constraint functions, e.g. the average stress, are discretized to enable their computations. In the discrete sensitivity analysis, the discretized PDEs and QoIs are differentiated wrt. the design parameters; yielding quantities that are readily computed. In the continuous sensitivity analysis, the PDEs and QoIs are first differentiated and subsequently discretized for their computation. For steady-state PDEs and normal finite element computation, e.g. using Lagrange basis functions with Gaussian quadratures, the *two approaches yield identical results.* For finite volume and finite difference methods this is not the case. And possibly it is not the case for "non traditional" FEM discretizations. Being that as it may, the discrete sensitivity analysis produces results that match the finite difference approximations and thus it is best suited for optimization.

My preference is nonetheless to use the continuous approach for two reasons

1. As just mentioned, for normal finite element computation the discrete and continuous approaches are identical. In the cases that they are not, with proper mesh refinement and adaptive time stepping both methods approach the analytical values and hence they converge to the same values.

2. MFEM, SMITH and Serac use a high level approach to enable HPC friendly finite element simulation. As described below, we merely have to mimic this approach to perform all of the sensitivity computations.

4 Sensitivity Analysis

Here we consider the sensitivity analysis of a general Quantity of Interest (QoI) which is defined in terms of a design field d and response field u. The design field is obtained from some sort of parameterization with respect to the parameters p such that $d(x) = \hat{d}(x,p)$ where d is the field and $\hat{d}$ is the function that produces the field from p. The response u for a fixed design d is computed by solving the possibly nonlinear primal problem, i.e PDE: find u such that

$$r(u, w, d) = 0 \tag{4}$$

for all kinematically admissible fields w. Note that r is linear wrt w but possibly nonlinear wrt u and d.

Having the pair (u, d) we define the QoI

$$\begin{aligned} \tilde{\theta}(d) &= \theta(u, d) \\ &= \int_{\Omega} \pi(u, d)\, dv \end{aligned} \tag{5}$$

4.1 Adjoint method

In the adjoint sensitivity method we note that $r(u, w, d) = 0$ to write

$$\tilde{\theta}(d) = \int_{\Omega} \pi(u, d)\, dv - r(u, w, d) \tag{6}$$

The variation of the functional $\tilde{\theta}$ at d in the direction δd, cf. (3), yields, after some rearranging.

$$\delta\tilde{\theta}(d; \delta d) = \overbrace{\int_{\Omega} \frac{\partial \pi}{\partial d} \cdot \delta d\, dv}^{\delta_d \theta(u,d;\delta d)} - \delta_d r(u, w, d; \delta d) + \overbrace{\int_{\Omega} \frac{\partial \pi}{\partial u} \cdot \delta u\, dv}^{\delta_u \theta(u,d;\delta u)} - a(\delta u, w, d) \tag{7}$$

where we use the definition of the tangent, i.e. the bilinear form wrt δu and w, $a(\delta u, w, d) = \delta_u r(u, w, d; \delta u)$. We also define the explicit $\delta_d \theta(u, d; \delta d)$ and implicit $\delta_u \theta(u, d; \delta u)$ variations of the functional θ at (u, d) in the directions δd and δu, respectively; they can be thought of as partial derivatives.

Recall that w is any kinematically admissible function, cf. (239). To eliminate the implicitly defined response variation δu in (7) we solve the adjoint problem: find w such that

$$a(\delta u, w, d) = \delta_u \theta(u, d; \delta u) \tag{8}$$

for all δu. Upon solving for the adjoint response w the sensitivity reduces to

$$\delta\tilde{\theta}(d; \delta d) = \delta_d \theta(u, d; \delta d) - \delta_d r(u, w, d; \delta d) \tag{9}$$

Knowing $\delta\tilde{\theta}(d; \delta d)$ and $\partial \hat{d}/\partial p$ we subsequently evaluate

$$\frac{\partial \tilde{\theta}}{\partial p} = \delta\tilde{\theta}(d; \partial \hat{d}/\partial p) \tag{10}$$

Figure 1 depicts the analysis and sensitivity analysis. Starting from the parameter vector p we 1) define the design field d, 2) evaluate the response u, 3) evaluate the pseudoload bilinear form $\delta r(u, w, d; \delta d)$ wrt. δd and w 4) evaluate the QoI $\tilde{\theta}$, and its variations $\delta_u \theta(u, d; \delta u)$ and $\delta_d \theta(u, d; \delta d)$, 5) evaluate the adjoint response w, 6) evaluate the sensitivity $\delta\tilde{\theta}$ and 7) evaluate the derivative $\partial\tilde{\theta}/\partial p$.

Notes:

1. The adjoint analysis is linear, even if the primal analysis is nonlinear.

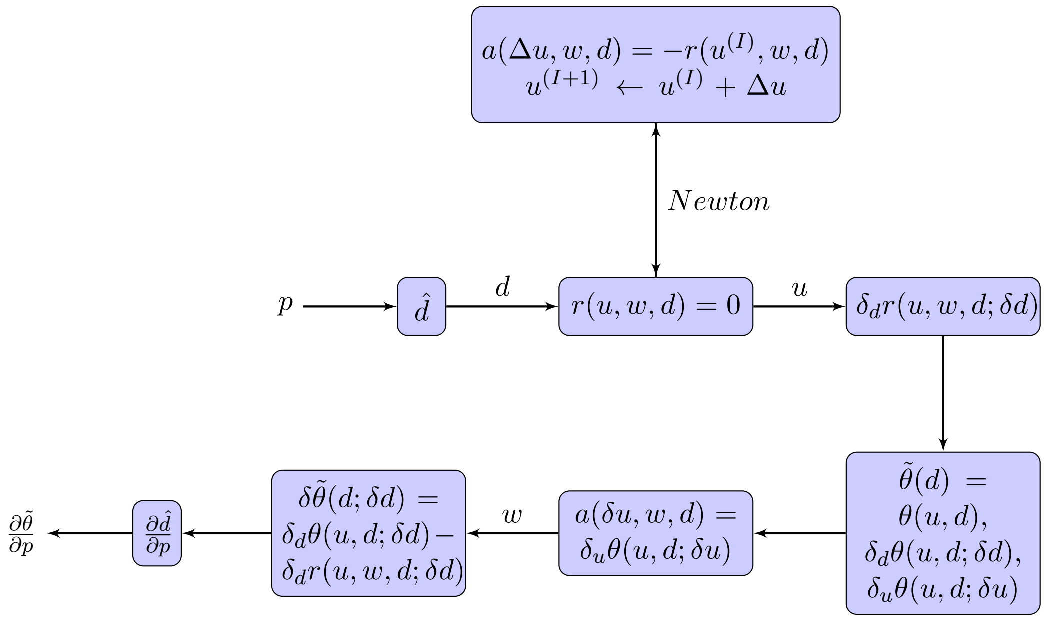


Figure 1: Analysis and sensitivity analysis.

2. The adjoint analysis uses the transpose, i.e. *adjoint*, of the bilinear operator a.

3. Defining some common terminology here seems appropriate. We call

   - the scalar $\theta(u, d)$ the *QoI*,
   - the scalar $\delta_d \theta(u, d; \delta d)$ the *explicit derivative*. Alternatively we can call the *linear form* $\delta_d \theta(u, d; \cdot)$ (with respect to δd) the *explicit derivative vector*, cf. Section 6.
   - the implicit derivative forms the load linear form $\delta_u \theta(u, d; \cdot)$ (with respect to δu) which we call the *adjoint load*.
   - the load linear form $\delta_d r(u, w, d; \cdot)$ (with respect to w) the *pseudo load*. Alternatively we can call the *bilinear* form $\delta r(u, w, d; \delta d)$ (with respect to w and δd) the *pseudo load matrix*, cf. Section 6.

   The QoI integrand, explicit derivatives and adjoint loads are associated with the QoI functional θ whereas the pseudo load is associated with the PDE that is used to evaluate the response u for the QoI. One pseudo load can be used to evaluate the sensitivity to any number of QoIs. So the QoI module should define/compute the numerous $\tilde{\theta}(d) = \theta(u, d)$, $\delta_d \theta(u, d; \cdot)$ and $\delta_u \theta(u, d; \cdot)$. The PDE module should solve the PDE for u and define/compute the pseudo load bilinear form $\delta r(u, w, d; \delta d)$ or load linear forms $\delta_d r(u, w, d; \cdot)$.

4. The pseudo load for hyperelasticity $\delta_d r(u, w, d; \delta d) = \int_\Omega \nabla w \cdot \frac{\partial^2 \hat{\psi}(F,x,d)}{\partial F \partial d}[\delta d]\, dv - \int_\Omega w \cdot \frac{\partial b}{\partial d}\, \delta d\, dv - \int_{A^n} w \cdot \frac{\partial t^p}{\partial d}\, \delta d\, dv$ can be evaluated the *same* as the residual $r(u, w, d) = \int_\Omega \nabla w \cdot \frac{\partial \hat{\psi}(F,x,d)}{\partial F}\, dv - \int_\Omega w \cdot b\, dv - \int_{A^n} w \cdot t^p\, dv$, but

we replace (ψ, b, t^p) with $(\frac{\partial \psi}{\partial d}\,\delta d, \frac{\partial b}{\partial d}\,\delta d, \frac{\partial t^p}{\partial d}\,\delta d)$.

5. For linear elasticity, the tangent bilinear form $\int_\Omega \nabla w \cdot \frac{\partial^2 \hat{\psi}(F,x,d)}{\partial F \partial F}[\Delta u]\,dv = \int_\Omega \nabla w \cdot \mathbb{C}[\Delta u]\,dv$ defines the stiffness matrix. The pseudo load contribution from this integral is $\int_\Omega \nabla w \cdot \frac{\partial^2 \hat{\psi}(F,x,d)}{\partial F \partial d}[\delta d]\,dv = \int_\Omega \nabla w \cdot \delta\mathbb{C}(d;\delta d)[\nabla u]\,dv$ which can be viewed as the design derivative of the stiffness matrix. CAN WE HAVE MFEM COMPUTE THE PSEUDO (LOAD LINEAR) FORM $\int_\Omega \nabla w \cdot \frac{\partial^2 \hat{\psi}(F,x,d)}{\partial F \partial d}[\delta d]\,dv$ wrt. δd? THIS IS IS AKIN TO THE RESIDUAL. FOR SURE WE CAN HAVE MFEM COMPUTE $\int_\Omega \nabla w \cdot \delta\mathbb{C}(d;\delta d)[\nabla u]\,dv$ AS A BILINEAR FORM wrt. w and δd. THE FORMER SAVES MEMORY AND THE LATTER SAVES COMPUTATIONS. **CF. SECTION 6 FOR AN ALTERNATE VIEWPONT OF THIS**.

6. To hasten development time we should have a *material library* that maps the displacement gradient $\nabla \mathbf{u}$ and design d fields to the stress $\mathbf{P} = \partial\psi/\partial\mathbf{F}$, incremental stiffness $\partial\mathbf{P}/\partial\mathbf{F}$ and design derivative $\frac{\partial \mathbf{P}}{\partial d}\,\delta d$ in different regions throughout the domain. This will allow us to readily compute the

   - Internal force residual (load linear form) $\int_\Omega \nabla\mathbf{w} \cdot \mathbf{P}\,dv$
   - Internal force tangent (bi linear form) $\int_\Omega \nabla\mathbf{w} \cdot \partial\mathbf{P}/\partial\nabla\mathbf{u}[\nabla\mathbf{u}]\,dv$
   - Internal force (load linear form) pseudo load contribution $\int_\Omega \nabla\mathbf{w} \cdot \frac{\partial \mathbf{P}}{\partial d}\,\delta d\,dv$

   IN THIS WAY, WE DO NOT HAVE TO KEEP DEFINING MFEM INTEGRATORS FOR EACH NEW MATERIAL MODEL. WE MERELY CHOOSE, OR WRITE, THE STRESS RESPONSE FUNCTION AND DEFINE ITS DERIVATIVES. WE COULD USE AD HERE.

7. To hasten development time we should have a *QoI library* that maps the response $(\mathbf{u}, \nabla\mathbf{u})$ and design d fields to the QoI integrand scalar π and its derivatives $\partial\pi/\partial\mathbf{u}$, $\partial\pi/\partial\nabla\mathbf{u}$ and $\partial\pi/\partial d$ that allows us to readily compute the

   - QoI (scalar) $\tilde{\theta}(d) = \theta(u,d) = \int_\Omega \pi(\mathbf{u}, \nabla\mathbf{u}, d)\,dv$
   - Adjoint (linear form) load $\delta_u\theta(u,d;\delta u) = \int_\Omega (\delta\mathbf{u} \cdot \partial\pi/\partial\mathbf{u} + \nabla\delta\mathbf{u} \cdot \partial\pi/\partial\nabla\mathbf{u})\,dv$
   - Explicit (scalar) derivative $\delta_d\theta(u,d;\delta d) = \int_\Omega \partial\pi/\partial d\,\delta d\,dv$

   To write, e.g. stress dependent QoIs, this module should have access to the PDE data, i.e. material module, load modules, ... IN THIS WAY, WE DO NOT HAVE TO KEEP DEFINING MFEM INTEGRATORS FOR EACH NEW FUNCTIONAL. WE MERELY CHOOSE, OR WRITE, THE π FUNCTION AND DEFINE ITS DERIVATIVES. WE COULD USE AD HERE.

8. With an adequate QoI library, one can readily define a QoI by specifying its π integrand, the region of integration $\Omega' \subset \Omega$ (mask), and the load case/PDE from which to obtain the response.

   - If the load case/PDE identifier is lacking, we know it is an explicit function, e.g. volume. We can omit the adjoint analysis for sensitivity analysis of these functions as $w = 0$.

4.2 Direct method

In the direct differentiation method we note that

$$\delta\tilde{\theta}(d;\delta d) = \int_\Omega \left(\frac{\partial \pi}{\partial d} \cdot \delta d + \frac{\partial \pi}{\partial u} \cdot \delta u \right) dv \tag{11}$$

where we dropped the arguments on $\delta u(d;\delta d)$. We can evaluate δu for a *specific* design variation δd by taking the variation of $r(u,w,d) = 0$ to obtain an equation which we solve for δu, i.e. we find δu such that

$$a(\delta u, w, d) + \delta_d r(u, w, d; \delta d) = 0 \tag{12}$$

for all w. In the above we recall that $a(\delta u, w, d) = \delta r_u(u, w, d; \delta u)$. Having δu we can readily evaluate (11), again, for the *specific* design variation δd.

4.3 Hessian product: $\delta d \cdot H[\delta d]$

We can also evaluate the Hessian wrt. the *specific* design variation δd. To do this we note the equality of (12) to express (11) as

$$\delta\tilde{\theta}(d;\delta d) = \int_\Omega \left(\frac{\partial \pi}{\partial d} \cdot \delta d + \frac{\partial \pi}{\partial u} \cdot \delta u \right) dv - \left(\delta r_u(u, w, d; \delta u) + \delta_d r(u, w, d; \delta d)\right) \tag{13}$$

Taking the variation of the above yields

$$\begin{aligned} \delta^2\tilde{\theta}(d;\delta d, \delta d) \quad = \quad & \int_\Omega \left(\frac{\partial^2 \pi}{\partial d^2}[\delta d] \cdot \delta d + 2\, \frac{\partial^2 \pi}{\partial d \partial u}[\delta d] \cdot \delta u + \frac{\partial^2 \pi}{\partial u^2}[\delta u] \cdot \delta u + \frac{\partial \pi}{\partial u} \cdot \delta^2 u \right) dv - \\ & \big(r_{uu}(u, w, d; \delta u, \delta u) + 2\, r_{ud}(u, w, d; \delta u, \delta d) + r_{dd}(u, w, d; \delta d, \delta d) + r_u(u, w, d; \delta^2 u)\big) \end{aligned} \tag{14}$$

where we dropped the arguments on $\delta^2 u(d;\delta d, \delta d)$. We can annihilate the second variation $\delta^2 u$ by solving the adjoint problem (8), i.e. we find w such that

$$a(\delta^2 u, w, d) = \delta_u \theta(u, d; \delta^2 u)$$

for all $\delta^2 u$. In this way, (14) reduces to

$$\begin{aligned} \delta^2\tilde{\theta}(d;\delta d) \quad = \quad & \int_\Omega \left(\frac{\partial^2 \pi}{\partial d^2}[\delta d] \cdot \delta d + 2\, \frac{\partial^2 \pi}{\partial d \partial u}[\delta d] \cdot \delta u + \frac{\partial^2 \pi}{\partial u^2}[\delta u] \cdot \delta u \right) dv - \\ & (r_{uu}(u, w, d; \delta u, \delta u) + 2\, r_{ud}(u, w, d; \delta u, \delta d) + r_{dd}(u, w, d; \delta d, \delta d)) \end{aligned} \tag{15}$$

4.4 Hessian product: $H[\delta d]$

We can also evaluate the Hessian product $H[\delta d]$ for a *specific* design variation δd. To do this we note the equalities of (239) and (12) to express (11) as

$$\delta\tilde{\theta}(d;\delta d) = \int_\Omega \left(\frac{\partial \pi}{\partial d} \cdot \delta d + \frac{\partial \pi}{\partial u} \cdot \delta u \right) dv + r(u, \tilde{w}, d) - \left(\delta r_u(u, w, d; \delta u) + \delta_d r(u, w, d; \delta d)\right) \tag{16}$$

Taking the variation of the above wrt. yields $\delta\tilde{d}$ gives

$$\begin{aligned} \delta^2\tilde{\theta}(d;\delta d;\delta\tilde{d}) \quad = \quad & \int_\Omega \left(\frac{\partial^2 \pi}{\partial d^2}[\delta d] \cdot \delta\tilde{d} + \frac{\partial^2 \pi}{\partial d \partial u}[\delta d] \cdot \delta\tilde{u} + \frac{\partial^2 \pi}{\partial u \partial d}[\delta u] \cdot \delta\tilde{d} + \frac{\partial^2 \pi}{\partial u^2}[\delta u] \cdot \delta\tilde{u} + \frac{\partial \pi}{\partial u} \cdot \delta^2 u \right) dv - \\ & \left(\delta r_u(u, \tilde{w}, d; \delta\tilde{u}) + \delta r_d(u, \tilde{w}, d; \delta\tilde{d}) \right) - \\ & \left(\delta^2 r_{uu}(u, w, d; \delta u, \delta\tilde{u}) + \delta^2 r_{ud}(u, w, d; \delta u, \delta\tilde{d}) + \right. \\ & \left. \delta^2 r_{du}(u, w, d; \delta d, \delta\tilde{u}) + \delta^2 r_{dd}(u, w, d; \delta d, \delta\tilde{d}) + \delta r_u(u, w, d; \delta^2 u) \right) \end{aligned} \tag{17}$$

where we dropped the arguments on $\delta^2 u(d;\delta d, \delta\tilde{d})$. We again annihilate the second variation $\delta^2 u$ by solving the adjoint problem (8), i.e. we find w such that

$$a(\delta^2 u, w, d) = \delta_u \theta(u, d; \delta^2 u)$$

for all $\delta^2 u$. And we annihilate the variation $\delta\tilde{u}$ by solving another adjoint problem, i.e. we find $\tilde{w}$ such that

$$a(\delta\tilde{u}, \tilde{w}, d) = \int_\Omega \left(\frac{\partial^2 \pi}{\partial d \partial u}[\delta d] + \frac{\partial^2 \pi}{\partial u^2}[\delta u] \right) \cdot \delta\tilde{u}\, dv - \left(\delta^2 r_{uu}(u, w, d; \delta u, \delta\tilde{u}) + \delta^2 r_{du}(u, w, d; \delta d, \delta\tilde{u}) \right) \tag{18}$$

for all $\delta\tilde{u}$. In this way, (17) reduces to

$$\begin{aligned} \delta^2\tilde{\theta}(d;\delta d;\delta\tilde{d}) \quad = \quad & \int_\Omega \left(\frac{\partial^2 \pi}{\partial d^2}[\delta d] + \frac{\partial^2 \pi}{\partial u \partial d}[\delta u] \right) \cdot \delta\tilde{d}\, dv - \delta r_d(u, \tilde{w}, d; \delta\tilde{d}) - \\ & \left(\delta^2 r_{ud}(u, w, d; \delta u, \delta\tilde{d}) + \delta^2 r_{dd}(u, w, d; \delta d, \delta\tilde{d}) \right) \end{aligned} \tag{19}$$

where δd is fixed, but $\delta\tilde{d}$ is arbitrary. So, since $\delta^2\tilde{\theta}(d;\delta d;\delta\tilde{d})$ is linear in the variations, we can readily factor out $\delta\tilde{d}$ to obtain the desired $\delta^2\tilde{\theta}(d;\delta d;\cdot) = H\,\delta d$ result.

5 SMITH - LIDO target problems

5.1 Inverse heat conduction problem

As a first attempt to link SMITH and LiDO we will solve the inverse heat conduction problem (IHCP) of finding the prescribed heat flux acting over the top surface A_{top} of an $l \times h = 4 \times 1$ homogeneous isotropic plate Ω with thermal conductivity $k = 2$. These problems arise in the design of heat shields wherein we cannot measure the heat flux on the outer surface, but instead must infer it from measurements taken on the inner surface. It is known that the plate has homogeneous flux $q^p = 0$ applied to its left A_{left} and right A_{right} surfaces, and a temperature of $T^p(x_1) = -\cos\left(\frac{2\,\pi\,x_1}{l}\right)$ applied to its bottom surface A_{bot}. The flux over the bottom surface has also been obtained via experiment; it is $q_{exp}(x_1) = \frac{2\,k\,\pi}{l}\coth\left(\frac{h\,\pi}{l}\right)\cos\left(\frac{2\,\pi\,x_1}{l}\right)$. We must now determine the prescribed heat flux q^p acting on the top surface so as to obtain the experimentally measured heat flux on the bottom surface.

To solve the IHCP we perform a conduction analysis over Ω in which we apply the $q^p = 0$ flux boundary conditions over the left and right surfaces, the $T^p(x_1) = -\cos\left(\frac{2\,\pi\,x}{l}\right)$ boundary condition over the bottom surface and any guessed heat flux $q^p(x_1)$ over the top surface. Upon completing the analysis we evaluate the the error function

$$\theta(q^p) = \int_{A_{bot}} \left(q^{\mathbf{n}} - q_{exp}\right)^2\, da + \epsilon \int_{A_{top}} (q^p)^2\, da \tag{20}$$

wherein the first integral quantifies the discrepancy between the measured and computed flux for the given q^p and the second integral is a regularization term that is used to make the IHCP well-posed.

To improve our q^p choice, we use optimization wherein we minimize the error function with respect to q^p. This process requires sensitivity analysis, which is actually a bit tricky, so we go through it in detail. First we recall that the temperature T satisfies

$$\begin{aligned} -k\,\triangle T &= 0 && \text{for } \mathbf{x} \in \Omega \\ -k\nabla T\cdot\mathbf{n} &= q^{\mathbf{n}} = 0 && \text{for } \mathbf{x} \in A_{left}\cup A_{right} \\ -k\nabla T\cdot\mathbf{n} &= q^{\mathbf{n}} = q^p && \text{for } \mathbf{x} \in A_{top} \\ T &= T^p && \text{for } \mathbf{x} \in A_{bot} \end{aligned} \tag{21}$$

We convert this strong problem to a weak problem by multiplying the first three equations by w and integrating over their respective domains to define the residual equation

$$r(T) = 0 = -\int_\Omega w\,k\,\triangle T\,dv + \int_{A_{left}\cup A_{right}} w\,k\nabla T\cdot\mathbf{n}\,da + \int_{A_{top}} w\,(k\nabla T\cdot\mathbf{n} + q^p)\,da \tag{22}$$

which holds for all functions w. An application of integration by parts and the divergence theorem transforms the above to

$$r(T) = 0 = \int_\Omega \nabla w\cdot k\,\nabla T\,dv + \int_{A_{top}} w\,q^p\,da - \int_{A_{bot}} w\,k\nabla T\cdot\mathbf{n}\,da \tag{23}$$

which holds for all *smooth* w. We next require that $w = 0$ on A_{bot} to obtain the residual equation which we solve via the FEM, i.e. we solve for $T \in H(T^p)$ such that

$$\begin{aligned} r(T) = 0 &= \int_\Omega \nabla w\cdot k\,\nabla T\,dv + \int_{A_{top}} w\,q^p\,da \\ &= a(T,w) - \ell(w) \end{aligned} \tag{24}$$

for all $w \in H(0)$ where

$$
\begin{aligned}
H(T^p) &= \{T \in H^1 | T = T^p \text{ on } A_{bot}\} \\
a(T, w) &= \int_\Omega \nabla w \cdot k \, \nabla T \, dv \\
\ell(w) &= -\int_{A_{top}} w \, q^p \, da
\end{aligned}
\tag{25}
$$

are the set of admissible functions, bilinear form and load linear form. In obtaining (24) we required that $w \in H(0)$ to eliminate reactive flux $q^{\mathbf{n}} = -k \, \nabla T \cdot \mathbf{n}$ over the surface A_{bot} where the temperature is prescribed. We do this because $T = T^p$ is known on this surface, so we have not lost anything. At any rate we solve (24) for T. But now we take a step back and note that this T also satisfies the equality

$$
r(T) = 0 = \int_\Omega \nabla w \cdot k \, \nabla T \, dv + \int_{A_{top}} w \, q^p \, da - \int_{A_{bot}} w \, \overbrace{k \nabla T \cdot \mathbf{n}}^{-q^{\mathbf{n}}} \, da
\tag{26}
$$

for all $w \in H^1$, i.e. we no longer require that these w satisfy $w = 0$ on $\partial\Omega$.

With this background in hand we now define the augmented functional

$$
\theta(q^p) = \int_{A_{bot}} (q^{\mathbf{n}} - q_{exp})^2 \, da + \epsilon \int_{A_{top}} (q^p)^2 \, da + \left(\int_\Omega \nabla w \cdot k \, \nabla T \, dv + \int_{A_{top}} w \, q^p \, da + \int_{A_{bot}} w \, q^{\mathbf{n}} \, da \right)
\tag{27}
$$

We have not changed a thing since the augmentable term equals zero, cf. (26).

Next we take the variation of (27) and do some rearranging to obtain

$$
\begin{aligned}
\delta\theta(q^p; \delta q^p) &= 2 \, \epsilon \int_{A_{top}} q^p \, \delta q^p \, da + \int_{A_{top}} w \, \delta q^p \, da + \\
&\quad \int_{A_{bot}} 2 \, (q^{\mathbf{n}} - q_{exp}) \, \delta q^{\mathbf{n}} \, da - \int_\Omega \nabla w \cdot k \, \nabla \delta T \, dv + \int_{A_{bot}} w \, \delta q^{\mathbf{n}} \, da
\end{aligned}
\tag{28}
$$

We can eliminate the flux variation $\delta q^{\mathbf{n}}$ and temperature variation δT by requiring w to solve the adjoint problem, i.e. we find $w \in H(-2 \, (q^{\mathbf{n}} - q_{exp}))$ such that

$$
\int_\Omega \nabla w \cdot k \, \nabla \delta T \, dv = 0
\tag{29}
$$

for all $\delta T \in H(0)$. The strong form of the adjoint problem is to find w such that

$$
\begin{aligned}
-k \, \triangle w &= 0 && \text{for } \mathbf{x} \in \Omega \\
-k \nabla w \cdot \mathbf{n} &= 0 && \text{for } \mathbf{x} \in A_{left} \cup A_{right} \cup A_{top} \\
w &= -2 \, (q^{\mathbf{n}} - q_{exp}) && \text{for } \mathbf{x} \in A_{bot}
\end{aligned}
\tag{30}
$$

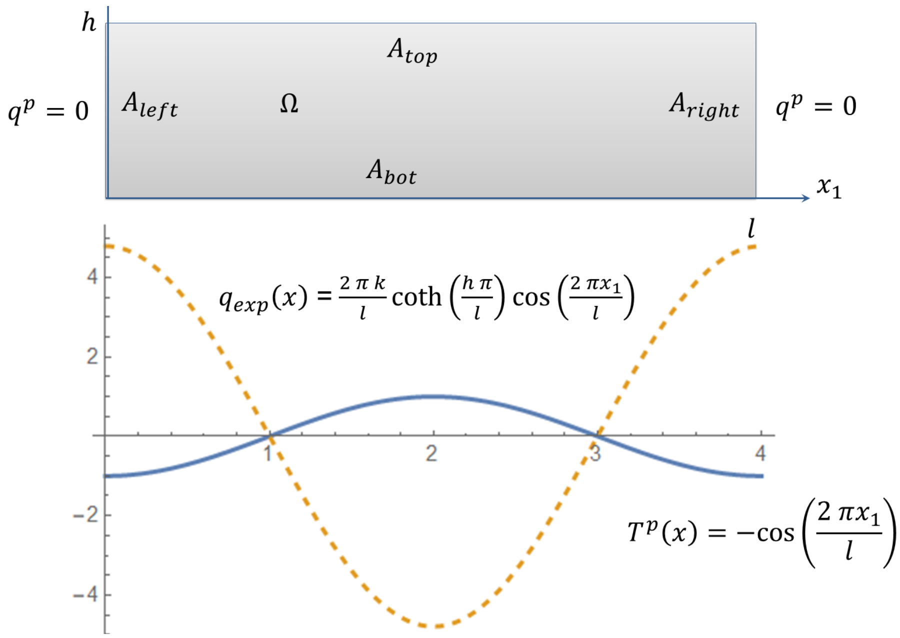


Figure 2: Inverse heat conduction problem illustration.

After solving for w, the sensitivity reduces to

$$\delta\theta(q^p; \delta q^p) = 2\,\epsilon \int_{A_{top}} q^p\,\delta q^p\,da + \int_{A_{top}} w\,\delta q^p\,da \tag{31}$$

For $\epsilon = 0$ we obtain the analytical solution to the above as $q^p(x_1) = -\frac{2\,k\,\pi}{l} \coth\left(\frac{h\,\pi}{l}\right) \cos\left(\frac{2\,\pi\,x_1}{l}\right)$ for which $T(\mathbf{x}) = -\frac{1}{2} \operatorname{csch}^2\left(\frac{h\,\pi}{l}\right) \cos\left(\frac{2\,\pi\,x_1}{l}\right) \left[\cosh\left(\frac{2\,\pi\,(h-x_2)}{l}\right) - \cosh\left(\frac{2\,\pi\,x_2}{l}\right)\right]$.

5.1.1 Smith consideration

For the Smith implementation we should

1. Input the mesh and define the finite element space $\mathcal{H}_{fem}(y)$ for the temperature T and weighting field w, specifically we use the H^1 Sobolev space with the restriction that the functions equal y on A_{bot}. As such $T \in \mathcal{H}_{fem}(T^p)$ and $T \in \mathcal{H}_{fem}(0)$ are now MFEM GridFunctions, i.e. represented via linear algebra vectors $\mathbf{T}$ and $\mathbf{W}$.

2. Partition $T = T_0 + T_e^p$ where $T_0 \in \mathcal{H}_{fem}(0)$ is the to be computed temperature field which equals zero on A_{bot} and $T_e^p \in \mathcal{H}_{fem}(T^p)$ is a known extension function which satisfies the essential BCs on A_{bot}.

3. Represent the unknown heat flux q^p via a spline of some sort in which the control point heights are the design variables $\mathbf{d} \in \mathcal{R}^{n_d}$ such that $q^p(\mathbf{x}, \mathbf{d})$. From this we will define a design field space *for the fixed design* $\mathbf{d}$, i.e. $\mathcal{U}_{des}(\mathbf{d}) = \{f : A_{bot} \to \mathcal{R} |\ f$ is somewhat smooth$\}$ such that $\mathbf{x} \mapsto q^p(\mathbf{x}, \mathbf{d})$. Reemphasizing, for the fixed $\mathbf{d}$, q^p is function of position $\mathbf{x}$ over the surface A_{bot}; e.g. $q^p(\cdot, \mathbf{d})$ can be an L_2 field which is defined as piecewise uniform over the element faces, or an H^1 field which is defined to be continuous and piecewise linear over the element faces, or ... Ultimately $q^p(\cdot, \mathbf{d}) \in \mathcal{U}_{des}(\mathbf{d})$ *is now represented via an MFEM GridFunction which is represented as the linear algebra vector* $\mathbf{Q}^p$. Also we evaluate the partial derivatives of q^p at the location-design pair $(\mathbf{x}, \mathbf{d})$, i.e. the linear operators $\partial q^p(\mathbf{x}, \mathbf{d})/\partial d_i \in \mathrm{Lin}(\mathcal{R} \to \mathcal{R})$. For a fixed $\mathbf{d}$ the derivatives $\partial q^p(\cdot, \mathbf{d})/\partial d_i$ are also elements of the *design field space* $\mathcal{U}_{des}(\mathbf{d})$ *and represented as the linear algebra vectors* $\partial \mathbf{Q}^p/\partial d_i$. Note that if there are many partial derivatives it may be beneficial to loop over these evaluations in Step 9.

4. Solve the primal problem: i.e. find $T_0 \in \mathcal{H}_{fem}(0)$ such that

$$\begin{aligned} \int_\Omega \nabla w \cdot k \, \nabla T_0 \, dv &= \int_{A_{top}} w \, q^p \, da - \int_\Omega \nabla w \cdot k \, \nabla T_e^p \, dv \\ a(T_0, w) &= \ell_{prim}(w) - a(T_e^p, w) \end{aligned} \tag{32}$$

for all $w \in \mathcal{H}_{fem}(0)$. In the above $a : \mathcal{H}_{fem}(\cdot) \times \mathcal{H}_{fem}(\cdot) \to \mathcal{R}$ is an MFEM bilinear form evaluated with the

```
BilinearForm *a = new BilinearForm(Hfem);
a->AddDomainIntegrator(new DiffusionIntegrator(k));
```

combination which is represented as the linear algebra matrix $\mathbf{K}$ and $\ell : \mathcal{H}_{fem}(\cdot) \to \mathcal{R}$ is an MFEM load linear form evaluated with

```
LinearForm *lprim = new LinearForm (Hfem);
lprima->AddBoundaryIntegrator(new BoundaryLFIntegrator(qp), Atop);
```

combination which is represented as the linear algebra vector $\mathbf{P}$ *(Here k is a constant. Can we accommodate the spatially varying flux GridFunction $q^p \in \mathcal{U}_{des}$ which is not in the dual space of the linear form ℓ_{prim}.???* Ultimately we solve

$$\mathbf{K}\,\mathbf{T}_0 = \mathbf{P} - \mathbf{K}\,\mathbf{T}_e^p \tag{33}$$

for $\mathbf{T}_0$.

5. Evaluate QoI relevant quantities

   (a) Integrate the QoI???

$$\theta(q^p) = \int_{A_{bot}} (q^{\mathbf{n}} - q_{exp})^2 \, da + \epsilon \int_{A_{top}} (q^p)^2 \, da \tag{34}$$

If not directly we can introduce the field $z \in \mathcal{U}_{fem}(1)$ defined such that $z(\mathbf{x}) = 1$ is uniform and evaluate the linear form

$$\theta(q^p)(z) = \int_{A_{bot}} z \overbrace{\left((q^{\mathbf{n}} - q_{exp})^2 + \epsilon \, (q^p)^2 \right)}^{xxx} da \tag{35}$$

where $\theta(q^p)(\cdot) \to \mathcal{R}$ is an MFEM load linear form evaluated with the

```
LinearForm * theta  = new  LinearForm (Hfem);
theta-> AddBoundaryIntegrator(new BoundaryLFIntegrator(xxx), Atop)
```

combination which is represented as the linear algebra vector $\mathbf{\Theta}$ To evaluate θ we can then multiply the resulting "load vector" $\mathbf{\Theta}$. with the unit vector $\mathbf{Z} = \mathbf{1} = [1, 1, \cdots]^T$, i.e. $\theta(q^p) = \mathbf{1}^T \, \mathbf{\Theta}$. *(Can we pass in the necessary xxx which has contributions from $q^{\mathbf{n}} \in \nabla\mathcal{H}_{fem}(T^p)$, $q^p \in \mathcal{U}_{des}$ and $q_{exp} \in$???)*?

(b) Evaluate the explicit derivative load linear form. *This linear form is dual to the q^p GridFunction space $\mathcal{U}_{des}$, not the T and w space $\mathcal{H}_{fem}(\cdot)$.*

$$\delta\theta(q^p; \delta q^p) = \int_{A_{top}} \overbrace{2 \, \epsilon \, q^p}^{yyy} \, \delta q^p \, da \tag{36}$$

We see that the linear form $\delta\theta(q^p; \cdot) : \mathcal{U}_{des} \to \mathcal{R}$ is an MFEM load linear form that can be evaluated with the

```
LinearForm * deltatheta  = new  LinearForm (Udes);
deltatheta-> AddBoundaryIntegrator(new BoundaryLFIntegrator(yyy), Atop)
```

combination which is represented as the linear algebra vector $\delta\mathbf{\Theta}$. *(Can we pass in the necessary yyy)*?

(c) Evaluate the adjoint load linear form wrt. the variation δT, i.e. it is dual to the set of temperature fields $\mathcal{H}_{fem}(\cdot)$

$$\ell_{adj}(\delta T) = 0 \tag{37}$$

For this problem we have no adjoint load.

(d) Evaluate the essential adjoint boundary conditions wrt. the variation δq^p

$$w^p = -2 \, (q^{\mathbf{n}} - q_{exp}) \qquad \text{for } \mathbf{x} \in A_{bot} \tag{38}$$

and form the known extension function $w_e^p \in \mathcal{H}_{fem}(w^p)$. *(Here $q_{exp} \in$??? and $q^{\mathbf{n}} \in \nabla H_{fem}$, can we do this?)*

6. Partition $w = w_0 + w_e^p$ such that $w_0 \in \mathcal{H}_{fem}(0)$ is the to be computed adjoint temperature field which equals zero on A_{bot} and w_e^p is the extensin function defined above.

7. Solve the adjoint problem, i.e. we solve for $w_0 \in \mathcal{H}_{fem}(0)$ such that

$$a(\delta T, w_e^p) = \cancelto{0}{\ell_{adj}(w)} - a(\delta T, w_e^p) \tag{39}$$

   for all $\delta T \in \mathcal{H}_{fem}(0)$. *Note that this solve uses the adjoint, i.e. transpose matrix, of the bilbear operator a.* Ultimately we solve

$$\mathbf{K}^T\, \mathbf{W}_0 = -\mathbf{K}^T\, \mathbf{W}_e^p \tag{40}$$

   for $\mathbf{W}_0$.

8. Evaluate the pseudo load contribution load linear form

$$\ell_{pseudo}(\delta q^p) = \int_{A_{top}} w\, \delta q^p \, da \tag{41}$$

   Again, this linear form is dual to the q^p GridFunction space $\mathcal{U}_{des}$, not the T and w space $\mathcal{H}_{fem}(\cdot)$. We also note that $\ell_{pseudo}(\cdot) : \mathcal{U}_{des} \to \mathcal{R}$ is an MFEM load linear form that should be added to $\delta\theta$ of (36) via

```
deltatheta-> AddBoundaryIntegrator( BoundaryLFIntegrator(w), Atop)
```

   so it is ultimately added to the linear algebra vector $\delta\mathbf{\Theta}$. As seen here, we are traversing the graph of Figure 3, i.e. adding $\ell_{pseudo}(\delta q^p)$ to the appropriate $\bar{q}^p$ node. *(Here $w \in \mathcal{H}_{fem}(w^p)$ and $\delta q^p \in \mathcal{U}_{des}$, is this possible? How can we pass in w?)*

9. Finally apply the chain rule to obtain the derivative wrt. the design parameters $\mathbf{d}$

$$\partial\theta/\partial d_i = \delta\theta(q^p; \partial q^p/\partial d_i) = \left(\frac{\partial \mathbf{Q}^p}{\partial d_i}\right)^T \delta\mathbf{\Theta} \tag{42}$$

5.2 Topology optimization

As a second attempt to link SMITH and LiDO we will solve the "heat sink" topology optimization problem of maximizing the "thermal compliance" subject to a volume constraint. [1] To do this we adapt the discussion of Section 6.

As seen in Figure 4, a square plate of length $L = 1$m is subjected to a uniform heat source $r = 1$ W/m^3, prescribed temperature $T = 0$ on the $0.1\,L$ portion of the left edge and zero heat flux over the remainder of the boundary. To test the SMITH-LiDO API we consider two design scenarios, one with an isotropic material and one with an orthotropic material.

[1] Schevenels, M., Lazarov, B.S., Sigmund, O. "Robust topology optimization accounting for spatially varying manufacturing errors," (2011) Computer Methods in Applied Mechanics and Engineering, 200 (49-52), pp. 3613-3627.

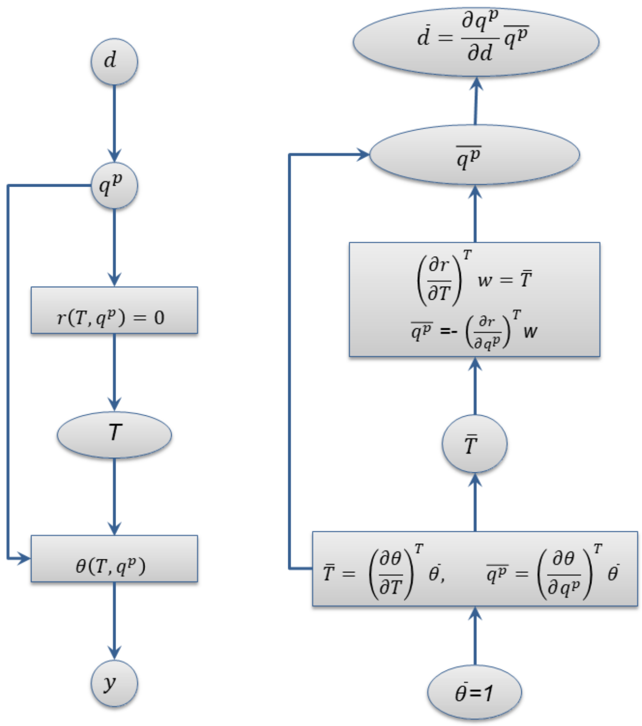


Figure 3: Inverse heat conduction problem AD graph.

5.2.1 Isotropic material

The material is of the SIMP variety, i.e. the thermal conductivity of the isotropic material is

$$\hat{\mathbf{k}}(\nu) = \mathbf{k} = (\nu^p + (1 - \nu^p)\,\epsilon)\,k_o\,\mathbf{I} \tag{43}$$

where $k_o = 1\mathrm{W}/(\mathrm{m}\cdot\mathrm{C})$ is the conductivity of the base material and $\epsilon = 10^{-3}$ is a small constant in this ersatz material model.

The constitutive relation is that of Fourier‘s law, i.e.

$$\mathbf{q} = \hat{\mathbf{q}}(\nabla T, \nu) = -\hat{\mathbf{k}}(\nu)\,\nabla T \tag{44}$$

so that

$$
\begin{aligned}
\delta\hat{\mathbf{q}}(\nabla T, \nu; \nabla\delta T) &= \underbrace{-\hat{\mathbf{k}}(\nu)}_{\frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}} \nabla\delta T \\
\delta\hat{\mathbf{q}}(\nabla T, \nu; \delta\nu) &= -\frac{\partial\hat{\mathbf{k}}(\nu)}{\partial\nu} \nabla T\,\delta\nu \\
&= \underbrace{-p\,\nu^{p-1}(1-\epsilon)\,k_o\,\nabla T}_{\frac{\partial\hat{\mathbf{q}}}{\partial\nu}}\,\delta\nu
\end{aligned} \tag{45}
$$

LiDO wlll provide SMITH with the Coefficient representations for both $\hat{\mathbf{q}}$ and its two derivatives. Or, preferably, SMITH will incorporate this functionality in its user material, i.e. "UMAT," option.

For the computations we will use an L_2 volume fraction field ν that is parameterized by Bezier spline volume fraction control point parameters $\mathbf{p}^\nu$ which is defined as

$$
\nu(\mathbf{x}) = \sum_i D_i^\nu\,\varphi_i(\mathbf{x}) \tag{46}
$$

where $\varphi_i : \Omega \to \{0, 1\}$ are the L_2 interpolations functions and the $D_i^\nu = \hat{d}(\mathbf{x}_i, \mathbf{p}^\nu)$ are the volume fraction values evaluated at the element Ω_i centroids via the explicit Bezier spline function $\hat{d} : \Omega \times \mathcal{R}^n \to [0, 1]$. LiDO will do the spline computations and generate the L_2 volume fraction field ν, i.e. Grid function, that will be used to define the constitutive response $\hat{\mathbf{q}}$.

For the SMITH implementation we parameterize the temperature via H^1 functions such that

$$
T(\mathbf{x}) = \sum_i T_i\,\phi_i(\mathbf{x}) \tag{47}
$$

where $\phi_i : \Omega \to \mathcal{R}$ are the H^1 interpolations functions and the T_i are the nodal temperature values.

Physics considerations: The PDE for this problem is presented in Section 9.3. The bilinear, load linear form and pseudo load definitions are as follows:

- residual:

$$
\begin{aligned}
r(u, \omega, d) &= \ell_1(\nabla\omega, d) + \ell_2(\omega, d) \\
&= \int_\Omega \nabla\omega \cdot \mathbf{q}\,dv + \int_\Omega \omega\,r\,dv \\
&\approx \sum_i \omega_i \left(\underbrace{\int_\Omega \nabla\phi_i \cdot \mathbf{q}\,dv}_{R_{1_i}} + \underbrace{\int_\Omega \phi_i\,r\,dv}_{R_{2_i}} \right)
\end{aligned} \tag{48}
$$

where R_{1_i} and R_{2_i} are the components of the *MFEM nonlinearform physics residual vector* associated with the load linear forms $\ell_1(\nabla\omega, d)$ and $\ell_2(\omega, d)$. The residual may also contain surface terms, which are omitted here for simplicity. We may also have a nonlinear constitutive response, e.g. if $\hat{\mathbf{k}}$ depends on the temperature T.

- tangent:

$$
\begin{aligned}
a_{11}(\nabla\Delta T, \nabla\omega) &= \delta\ell_1(\nabla\omega, d; \nabla\Delta T) \\
&= \int_\Omega \nabla\omega \cdot \delta\hat{\mathbf{q}}(\cdot; \nabla\Delta T)\, dv \\
&= \int_\Omega \nabla\omega \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}[\nabla\Delta T]\, dv \\
&\approx \sum_{i,j} \omega_i \underbrace{\int_\Omega \nabla\phi_i \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}\, \nabla\phi_j\, dv}_{K_{ij}}\, \Delta T_j
\end{aligned}
\tag{49}
$$

where K_{ij} is the component of the *MFEM nonlinearform physics tangent stiffness matrix.*

- pseudo load *operating on the known adjoint temperature field* ω:

$$
\begin{aligned}
\delta\ell_1(\nabla\omega, d; \delta d) &= \int_\Omega \nabla\omega \cdot \delta\hat{\mathbf{q}}(\cdot; \delta d)\, dv \\
&= \underbrace{\sum_j \int_\Omega \nabla\omega \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nu}\, \varphi_j\, dv}_{\left(\left(\frac{\partial\mathbf{R}_1}{\partial\mathbf{D}}\right)^T \mathbf{W}\right)_j}\, \delta D_j^\nu
\end{aligned}
\tag{50}
$$

where $\left((\partial\mathbf{R}_1/\partial\mathbf{D})^T\,\mathbf{W}\right)_j$ is a component of the pseudo *vector* $\{\ (\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}\ \}$. LiDO will supply SMITH with the L_2 Grid function representation of the volume fraction field ν. Recall that T, ω and ν are interpolated via the basis functions ϕ_i, ϕ_i and φ_i. Also recall that the adjoint response is computed from, cf Figure **??**,

$$
\mathbf{K}^T\,\mathbf{W} = \frac{\partial\Theta}{\partial\mathbf{T}}
\tag{51}
$$

where $\mathbf{W}$ is the Grid function representation of ω and where the computation of $\frac{\partial\Theta}{\partial\mathbf{T}}$ is described in (53).

QoI considerations: The QoI should take advantage of the analysis modules if at all possible. In this compliance QoI we see how the heat flux module that encodes the constitutive relation (44) and its variations (45) that is used to evaluate the residual (48) tangent (49) and pseudo load (50) is also utilized to evaluate the QoI and its variations.

- thermal compliance QoI:

$$\begin{aligned}\theta(T,d) &= \int_\Omega \pi(\mathbf{q},\nabla T,\nu)\,dv \\ &= \underbrace{\int_\Omega \frac{1}{2}\,\mathbf{q}\cdot\nabla T\,dv}_{\Theta} \end{aligned} \tag{52}$$

  LiDO will compute the QoI Θ via a nonlinearform integrator.

- adjoint load:

$$\begin{aligned}\delta\theta(T,d;\delta T) &= \int_\Omega \delta\pi(\cdot;\nabla\delta T)\,dv \\ &= \int_\Omega \frac{1}{2}\left(\left(\frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}\right)^T \frac{\partial\pi}{\partial\mathbf{q}}\cdot\nabla\delta T + \frac{\partial\pi}{\partial\nabla T}\cdot\nabla\delta T\right)\,dv \\ &= \int_\Omega \frac{1}{2}\left(\left(\frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}\right)^T \nabla T\cdot\nabla\delta T + \mathbf{q}\cdot\nabla\delta T\right)\,dv \\ &= \sum_i \underbrace{\int_\Omega \frac{1}{2}\left(\left(\frac{\partial\hat{\mathbf{q}}}{\partial\nabla T}\right)^T \nabla T + \mathbf{q}\right)\cdot\nabla\phi_i\,dv}_{\frac{\partial\Theta}{\partial T_i}}\,\delta T_i \end{aligned} \tag{53}$$

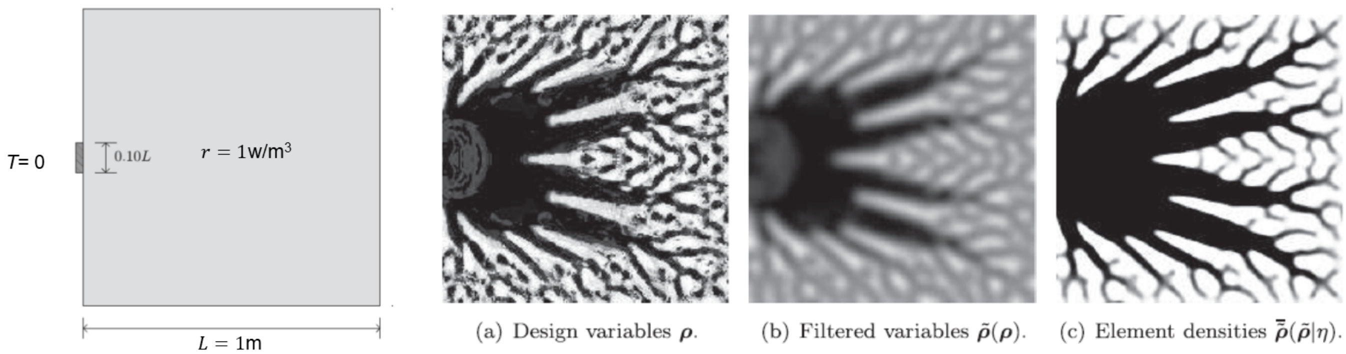
  where $\partial\Theta/\partial T_i$ is a component of the *MFEM nonlinearform physics adjoint load vector* $\partial\Theta/\partial\mathbf{T}$. LiDO will supply $\partial\pi/\partial\nabla T$ to SMITH in Vector Coefficient form. Note that this load is akin to an initial stress.

- explicit sensitivity:

$$\begin{aligned}\delta\theta(T,d;\delta d) &= \int_\Omega \delta\pi(\cdot;\delta\nu)\,dv \\ &= \int_\Omega \frac{\partial\pi}{\partial\mathbf{q}}\cdot\frac{\partial\hat{\mathbf{q}}}{\partial\nu}\,\delta\nu\,dv \\ &= \sum_j \underbrace{\int_\Omega \frac{1}{2}\,\nabla T\cdot\frac{\partial\hat{\mathbf{q}}}{\partial\nu}\varphi_j\,dv}_{\frac{\partial\Theta}{\partial\mathbf{D}_j^\nu}}\,\delta D_j^\nu \end{aligned} \tag{54}$$

  $\partial\Theta/\partial D_j^\nu$ is a component of the *MFEM nonlinearform explicit sensitivity vector* $\partial\theta/\partial\mathbf{D}^\nu$. LiDO will compute $\partial\Theta/\partial\mathbf{D}^\nu$ via a nonlinearform integrator.

The graph for the computations of the thermal compliance QoI and its sensitivity appear in Figure 5. LiDO will perform the computations of the first block that concerns the Bezier function and the last block which concerns the QoI, although LiDO will rely on the MFEM nonlinearform operator for some of these computations. SMITH will perform the operations in the middle block concerning the physics. DOES SEEM REASONABLE?



(a) Design variables $\boldsymbol{\rho}$. (b) Filtered variables $\tilde{\boldsymbol{\rho}}(\boldsymbol{\rho})$. (c) Element densities $\bar{\tilde{\boldsymbol{\rho}}}(\tilde{\boldsymbol{\rho}}|\eta)$.

Figure 4: Heat sink topology optimization problem: Domain, loads and boundary conditions (left), optimized volume fraction (a), filtered volume fraction (b) and thresholded filtered volume fraction (d).

5.2.2 Orthotropic material

In a second more complicated test we replace the isotropic conductivity tensor with the orthotropic tensor

$$\hat{\mathbf{k}}(\nu, \alpha) = \mathbf{k} = (\nu^p + (1 - \nu^p)\,\epsilon)\,\mathbf{R}(\alpha)\,\mathbf{k}_o\,\mathbf{R}^T(\alpha) \tag{55}$$

where $\mathbf{k}_o = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ W/(m · C) is the conductivity of the base material and $\mathbf{R}(\alpha) = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$ is a rotation tensor about the $\mathbf{e}_3$ axis. The constitutive relation is still that of Fourier's law, but now we have

$$\begin{aligned} \delta\hat{\mathbf{q}}(\nabla T, \nu, \alpha; \delta\nu) &= -\frac{\partial \hat{\mathbf{k}}(\nu,\alpha)}{\partial \nu}\,\nabla T\,\delta\nu \\ &= \underbrace{-p\,\nu^{p-1}(1-\epsilon)\;\mathbf{R}(\alpha)\,\mathbf{k}_o\,\mathbf{R}^T(\alpha)\,\nabla T}_{\frac{\partial \mathbf{q}}{\partial \nu}}\;\delta\nu \\ \delta\hat{\mathbf{q}}(\nabla T, \nu, \alpha; \delta\alpha) &= -\frac{\partial \hat{\mathbf{k}}(\nu,\alpha)}{\partial \alpha}\,\nabla T\,\delta\alpha \\ &= \underbrace{-(\nu^p + (1-\nu^p)\,\epsilon)\;\big(\mathbf{R}'(\alpha)\,\mathbf{k}_o\,\mathbf{R}^T(\alpha) + \mathbf{R}(\alpha)\,\mathbf{k}_o\,\mathbf{R}'^T(\alpha)\big)\;\nabla T}_{\frac{\partial \mathbf{q}}{\partial \alpha}}\;\delta\alpha \end{aligned} \tag{56}$$

We discretize the orientation α analogous to the volume fraction ν, i.e. α is parameterized by Bezier spline basis function and the orientation control point parameters $\mathbf{p}^\alpha$. We use the same basis functions $\varphi_i : \Omega \to \{0, 1\}$ to interpolate both ν and α, i.e. we also have

$$\alpha(\mathbf{x}) = \sum_i D_i^\alpha\,\varphi_i(\mathbf{x}) \tag{57}$$

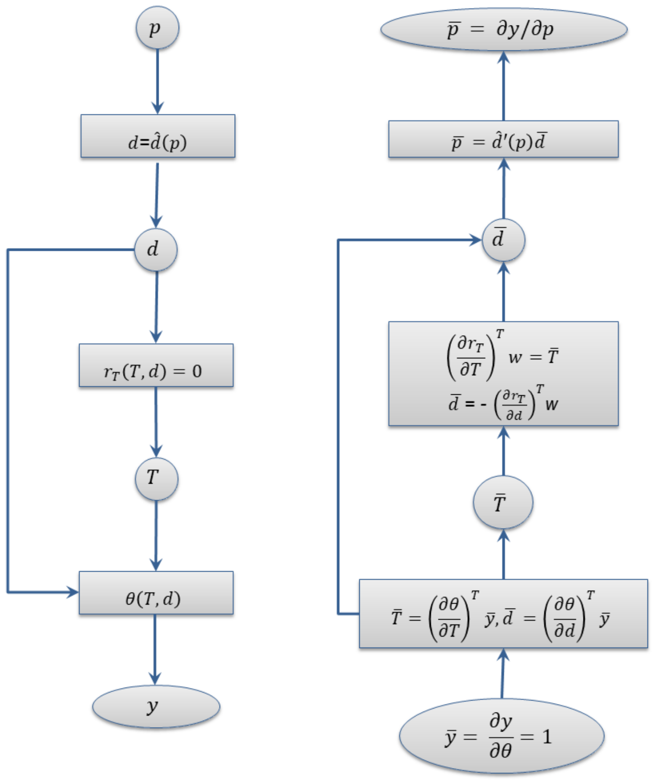


Figure 5: Heat sink topology optimization problem AD graph.

where $D_i^\alpha = \hat{d}(\mathbf{x}_i, \mathbf{p}^\alpha)$ are the orientation values evaluated at the element Ω_i centroids via the same explicit Bezier spline function $\hat{d} : \Omega \times \mathcal{R}^n \to [0,1]$. LiDO will supply SMITH with the L_2 Grid function representations of the volume fraction ν and orientation α. WOULD IT BE BETTER TO PASS THIS ALONG AS 1 VECTOR VALUED FIELD RATHER THAN 2 SCALAR VALUED FIELDS? IS THIS POSSIBLE? IT SEEMS LIKE THIS IS POSSIBLE AS IT IS NO DIFFERENT THAN REPRESENTING THE DISPLACEMENT VECTOR.

The residual and tangent remain unchanged, except for the form of $\mathbf{q}$ and $\partial\mathbf{q}/\partial\nabla T$ which is supplied above. Since we have two inputs ν and α we have two pseudo loads operating on the adjoint temperature ω

- Volume fraction:

$$
\begin{aligned}
\delta\ell_1(\nabla\omega, d; \delta d^\nu) &= \int_\Omega \nabla\omega \cdot \delta\hat{\mathbf{q}}(\cdot;\delta\nu)\, dv \\
&= \underbrace{\int_\Omega \nabla\omega \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nu}\, \varphi_j\, dv}_{\left(\left(\frac{\partial\mathbf{R}_1}{\partial\mathbf{D}^\nu}\right)^T \mathbf{W}\right)_j} \delta D_j^\nu
\end{aligned} \tag{58}
$$

- Orientation:

$$
\begin{aligned}
\delta\ell_1(\nabla\omega, d; \delta d^\alpha) &= \int_\Omega \nabla\omega \cdot \delta\hat{\mathbf{q}}(\cdot;\delta\alpha)\, dv \\
&= \underbrace{\int_\Omega \nabla\omega \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\alpha}\, \varphi_j\, dv}_{\left(\left(\frac{\partial\mathbf{R}_1}{\partial\mathbf{D}^\alpha}\right)^T \mathbf{W}\right)_j} \delta D_j^\alpha
\end{aligned} \tag{59}
$$

where the $\left((\partial\mathbf{R}_1/\partial\mathbf{D}^\nu)^T\, \mathbf{W}\right)_j$ and $\left((\partial\mathbf{R}_1/\partial\mathbf{D}^\alpha)^T\, \mathbf{W}\right)_j$ are components of the *vectors* $\left\{\ (\partial\mathbf{R}/\partial\mathbf{D}^\nu)^T\mathbf{W}\ \right\}$ and $\left\{\ (\partial\mathbf{R}/\partial\mathbf{D}^\alpha)^T\mathbf{W}\ \right\}$.

The QoI and adjoint load also remain unchanged, except for the form of $\mathbf{q}$ which is supplied above. For the explicit sensitivity we now have two terms for the two design fields

- Volume fraction:

$$
\begin{aligned}
\delta\theta(T, d; \delta d^\nu) &= \int_\Omega \delta\pi(\cdot;\delta\nu)\, dv \\
&= \int_\Omega \frac{\partial\pi}{\partial\mathbf{q}} \cdot \frac{\partial\mathbf{q}}{\partial\nu}\, \delta\nu\, dv \\
&= \int_\Omega \frac{1}{2}\, \nabla T \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nu}\, \delta\nu\, dv \\
&= \sum_j \underbrace{\int_\Omega \frac{1}{2}\, \nabla T \cdot \frac{\partial\hat{\mathbf{q}}}{\partial\nu}\, \varphi_j\, dv}_{\partial\Theta/\partial D_j^\nu} \delta D_j^\nu
\end{aligned} \tag{60}
$$

- Orientation:

$$
\begin{aligned}
\delta\theta(T, d; \delta d^{\alpha}) &= \int_{\Omega} \delta\pi(\cdot\,; \delta\alpha)\, dv \\
&= \int_{\Omega} \frac{\partial \pi}{\partial \mathbf{q}} \cdot \frac{\partial \mathbf{q}}{\partial \alpha}\, \delta\alpha\, dv \\
&= \int_{\Omega} \frac{1}{2}\, \nabla T \cdot \frac{\partial \hat{\mathbf{q}}}{\partial \alpha}\, \delta\alpha\, dv \\
&= \sum_{j} \underbrace{\int_{\Omega} \frac{1}{2}\, \nabla T \cdot \frac{\partial \hat{\mathbf{q}}}{\partial \alpha}\, \varphi_j\, dv}_{\partial\Theta/\partial D_j^{\alpha}}\, \delta D_j^{\alpha}
\end{aligned} \tag{61}
$$

where $\partial\Theta/\partial D_j^{\nu}$ and $\partial\Theta/\partial D_j^{\alpha}$ are components of the *MFEM nonlinearform explicit sensitivity vectors* $\partial\Theta/\partial\mathbf{D}^{\nu}$ and $\partial\Theta/\partial\mathbf{D}^{\alpha}$.

The graph for this system appears in Figure 6.

5.3 Compliance problem

This section is a specialization of that presented in Section 4. Here the physics is governed by the equations of linear elasticity for a heterogeneous isotropic material over the domain Ω. As such the symmetric elasticity tensor that describes the linear elastic isotropic material response is

$$
\mathbb{C} = 2\,\mu\,\mathbb{I}_s + \lambda\,\mathbf{I}\otimes\mathbf{I} \tag{62}
$$

where the pair of Lame parameters (μ, λ) constitutes the *design field* $\mathbf{d}$ which changes wrt. position, i.e. $\mathbf{d}(\mathbf{x}) = (\mu(\mathbf{x}), \lambda(\mathbf{x}))$ and $\mathbb{I}_s$ is the symmetrizer defined such that $\mathbb{I}_s[\mathbf{A}] = \frac{1}{2}\,(\mathbf{A} + \mathbf{A}^T)$ for any 2-tensor $\mathbf{A}$. To complete the physics description we also need the data which defines the loads and boundary conditions. For this problem we have prescribed zero displacement over the boundary region $A_u \subset \partial\Omega$, nonzero prescribed traction $\mathbf{t}^p$ over the disjoint boundary region $A_t \subset \partial\Omega$ and zero prescribed traction over the remainder of the boundary.

Having the domain Ω, elasticity tensor $\mathbb{C}$ and load data we solve for the displacement $\mathbf{u}$, i.e. we solve the primal problem: Find $\mathbf{u} \in \mathcal{H}$ such that

$$
\begin{aligned}
r(\mathbf{u}, \mathbf{w}, \mathbf{d}) = 0 &= \int_{\Omega} \nabla\mathbf{w} \cdot \mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]\, dv - \int_{A_t} \mathbf{w} \cdot \mathbf{t}^p\, da \\
&= a(\mathbf{u}, \mathbf{w}, \mathbf{d}) - \ell(\mathbf{w}, \mathbf{d})
\end{aligned} \tag{63}
$$

for all $\mathbf{w} \in \mathcal{H}$ where

$$
\mathcal{H} = \{\mathbf{u} \in H^1 : \mathbf{u}(\mathbf{x}) = \mathbf{0} \text{ for } \mathbf{x} \in A_u\} \tag{64}
$$

is the set of kinematically admissible displacement fields, a is the bilinear form wrt. $\mathbf{u}$ and $\mathbf{w}$ and ℓ is the load linear form wrt. $\mathbf{w}$.

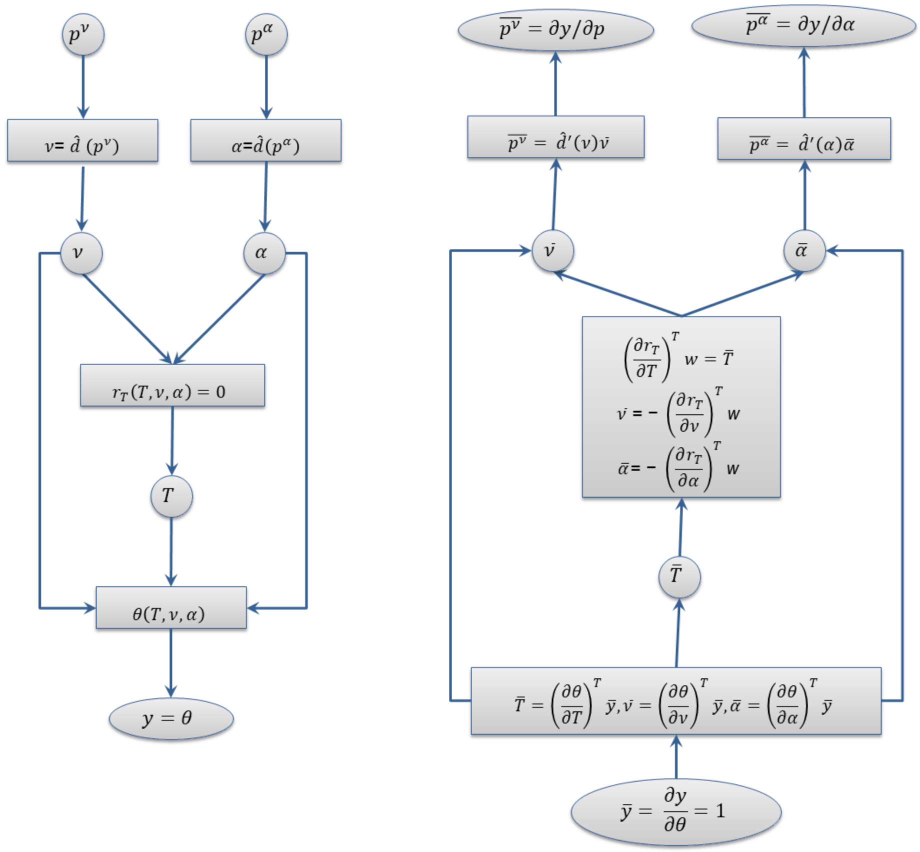


Figure 6: Orthotropic heat sink topology optimization problem AD graph.

In MFEM we represent the displacements $\mathbf{u}$ and $\mathbf{w}$ Grid Functions so that,

$$\begin{aligned}\mathbf{u}(\mathbf{x}) &= \sum_i U_i\, \boldsymbol{\phi}_i(\mathbf{x}) \\ \mathbf{w}(\mathbf{x}) &= \sum_i W_i\, \boldsymbol{\phi}_i(\mathbf{x})\end{aligned} \tag{65}$$

where $\boldsymbol{\phi}_i$ are interpolation functions that most likely are elements of H^1, i.e. $\boldsymbol{\phi}_i \in H^1$. Inserting these approximations into (63) gives

$$\begin{aligned} r(\mathbf{u},\mathbf{w},\mathbf{d}) &= \sum_j \sum_k W_k \overbrace{\int_\Omega \nabla\phi_k \cdot \mathbb{C}(\mathbf{d})[\nabla\phi_k]\, dv}^{K_{kj}} U_j - \sum_k W_k \overbrace{\int_{A_t} \phi_k\, \mathbf{t}^p\, da}^{P_k} \\ &= \mathbf{W}^T (\mathbf{K}\,\mathbf{U} - \mathbf{P}) \end{aligned} \tag{66}$$

The arbitrariness of $\mathbf{W}$ provides the usual linear equation $\mathbf{K}\,\mathbf{U} = \mathbf{P}$ that we solve for $\mathbf{U}$.

Knowing the displacement field $\mathbf{u}$ allows us to compute other response quantities such as the strain and stress

$$\begin{aligned} \boldsymbol{\epsilon} &= \frac{1}{2}\left(\nabla\mathbf{u} + \nabla^T\mathbf{u}\right) \\ \boldsymbol{\sigma} &= \mathbb{C}(\mathbf{d})[\nabla\mathbf{u}] \end{aligned} \tag{67}$$

Having the displacement $\mathbf{u}$ we also evaluate the compliance QoI as [2]

$$\begin{aligned} \tilde{\theta}(\mathbf{d}) &= \theta(\mathbf{u},\mathbf{d}) \\ &= \int_\Omega \pi(\nabla\mathbf{u},\mathbf{d})\, dv \\ &= \int_\Omega \nabla\mathbf{u} \cdot \underbrace{\mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]}_{\boldsymbol{\sigma}}\, dv \end{aligned} \tag{68}$$

The explicit and implicit variations of the above are required for the sensitivity analysis. They are evaluated as

$$\begin{aligned} \delta_d\theta(\mathbf{u},\mathbf{d};\delta\mathbf{d}) &= \int_\Omega \frac{\partial\pi}{\partial\mathbf{d}} \cdot \delta\mathbf{d}\, dv = \int_\Omega \nabla\mathbf{u} \cdot \frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}][\nabla\mathbf{u}]\, dv \\ \delta_u\theta(\mathbf{u},\mathbf{d};\delta\mathbf{u}) &= \int_\Omega \frac{\partial\pi}{\partial\nabla\mathbf{u}} \cdot \delta\nabla\mathbf{u}\, dv = \int_\Omega 2\, \underbrace{\mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]}_{\boldsymbol{\sigma}} \cdot \nabla\delta\mathbf{u}\, dv \end{aligned} \tag{69}$$

where

$$\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}] = 2\,\delta\mu\mathbb{I} + \delta\lambda\,\mathbf{I}\otimes\mathbf{I} \tag{70}$$

The variations $\delta\mathbf{d}$ and $\nabla\delta\mathbf{u}$ are interpolated fields like the displacement $\mathbf{u}$, i.e. they are to be treated as MFEM grid functions and interpolated accordingly. Most likely, $\mathbf{d} = (\mu,\lambda)$ will be an L_2 function and $\mathbf{u}$ an H^1 function and consequently $\delta\mathbf{d} = (\delta\mu,\delta\lambda)$ will be an L_2 function while $\nabla\delta\mathbf{u}$ will be the spatial derivative of the H^1 function

[2]Note: If we equate the arbitrary $\mathbf{w}$ in (63) to $\mathbf{u}$, then we see that we can equally express the compliance as $\tilde{\theta}(\mathbf{d}) = \int_{A_t} \mathbf{w}\cdot\mathbf{t}^p\, da$. However, we do not opt for this definition.

$\delta\mathbf{u}$. The variations themselves are linear forms, e.g. $\delta_d\theta(\mathbf{u},\mathbf{d};\cdot)$ and $\delta_u\theta(\mathbf{u},\mathbf{d};\cdot)$ are linear forms that are dual to $\delta\mathbf{d}$ and $\delta\mathbf{u}$. To be clear, let us interpolate

$$\mathbf{d}(\mathbf{x}) = \sum_i D_i\,\boldsymbol{\varphi}_i(\mathbf{x}) \tag{71}$$

where, with out loss of generality we assume $\boldsymbol{\varphi}_i \in L_2$. Then

$$\begin{aligned}
\delta_d\theta(\mathbf{u},\mathbf{d};\delta\mathbf{d}) &= \sum_i \overbrace{\int_\Omega \frac{\partial\pi}{\partial\mathbf{d}}\cdot\boldsymbol{\varphi}_i\,dv}^{\frac{\partial\Theta}{\partial D_i}}\,\delta D_i = \sum_i \int_\Omega \nabla\mathbf{u}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\boldsymbol{\varphi}_i][\nabla\mathbf{u}]\,dv\,\delta D_i = \frac{\partial\Theta}{\partial\mathbf{D}}\,\delta\mathbf{D} \\
\delta_u\theta(\mathbf{u},\mathbf{d};\delta\mathbf{u}) &= \sum_j \overbrace{\int_\Omega \frac{\partial\pi}{\partial\nabla\mathbf{u}}\cdot\delta\nabla\mathbf{u}\,dv}^{\frac{\partial\Theta}{\partial U_j}}\,\delta U_j = \sum_j \int_\Omega 2\,\mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]\cdot\nabla\boldsymbol{\phi}_j\,dv\,\delta U_j = \frac{\partial\Theta}{\partial\mathbf{U}}\,\delta\mathbf{U}
\end{aligned} \tag{72}$$

To perform the sensitivity analysis we now solve the adjoint problem: Find $\mathbf{w}\in\mathcal{H}$ such that

$$\begin{aligned}
0 &= \delta_u\theta(\mathbf{u},\mathbf{d};\delta\mathbf{u}) - a(\delta\mathbf{u},\mathbf{w},\mathbf{d}) \\
&= \int_\Omega \nabla\delta\mathbf{u}\cdot 2\,\mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]\,dv - \int_\Omega \nabla\mathbf{w}\cdot\mathbb{C}(\mathbf{d})[\nabla\delta\mathbf{u}]\,dv
\end{aligned} \tag{73}$$

for all $\delta\mathbf{u}\in\mathcal{H}$. This is the same bilinear form $a(\delta\mathbf{u},\mathbf{w},\mathbf{d})$, however, the positions of the arbitrary weighting function, $\delta\mathbf{u}$ and the field to be evaluated $\mathbf{w}$ have changed positions versus that in the primal problem (63) where we have $a(\mathbf{u},\mathbf{w},\mathbf{d})$ wherein $\mathbf{w}$ is the arbitrary weighting function and $\mathbf{u}$ is the field to be evaluated. So we use the transpose (or *adjoint*) of the bilinear form a that is used in the primal problem here in the adjoint problem. But for this linear elasticity case, a is symmetric so we need not worry about this transposition. For reasons that are hopefully self evident, the implicit variation $\delta_u\theta(\mathbf{u},\mathbf{d};\cdot)$ is usually called the *adjoint load linear form.*

Ultimately after MFEM discretization we have

$$0 = \delta\mathbf{U}^T\left(\frac{\partial\Theta}{\partial\mathbf{U}}^T - \mathbf{K}^T\,\mathbf{W}\right) \tag{74}$$

which, accounting for the arbitrariness of $\delta\mathbf{U}$ renders the linear adjoint problem $\mathbf{K}^T\,\mathbf{W} = \frac{\partial\Theta}{\partial\mathbf{U}}^T$ that we solve for $\mathbf{W}$. And again, we note that $\mathbf{K}^T = \mathbf{K}$. As seen above, the stiffness matrix for the adjoint analysis is the same one that is used in the primal analysis and thus we need not recompute it. Furthermore, we can possibly gain efficiency by developing better preconditioners for the primal, and hence adjoint, simulations. [3]

[3] Note: If we equate the arbitrary $\mathbf{w}$ in (63) to $\delta\mathbf{u}$, then we see that $\int_\Omega \nabla\delta\mathbf{u}\cdot 2\,\mathbb{C}(\mathbf{d})[\nabla\mathbf{u}]\,dv = \int_{A_t}\delta\mathbf{u}\cdot 2\,\mathbf{t}^p\,da$ and hence $\mathbf{w} = 2\,\mathbf{u}$. This known relationship between the primal and adjoint solutions is a QoI-specific result that does not generalize to other QoIs. Please do not use this simplification under any circumstance.

All that remains is to evaluate the sensitivity, i.e. we now have

$$\begin{aligned}
\delta\tilde{\theta}(\mathbf{d};\delta\mathbf{d}) &= \delta\theta(\mathbf{u},\mathbf{d};\delta\mathbf{d}) - \delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\delta\mathbf{d}) \\
&= \delta\theta(\mathbf{u},\mathbf{d};\delta\mathbf{d}) - \delta_d a(\mathbf{u},\mathbf{w},\mathbf{d};\delta\mathbf{d}) \\
&= \int_\Omega \nabla\mathbf{u}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}][\nabla\mathbf{u}]\,dv - \int_\Omega \nabla\mathbf{w}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}][\nabla\mathbf{u}]\,dv \qquad (75)
\end{aligned}$$

Similar to above, $\delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\cdot)$ is a one form that is dual to the variation $\delta\mathbf{d}$. [4] For discretization, all that remains is the discretization of $\delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\delta\mathbf{d})$, which, upon using the above interpolations becomes

$$\begin{aligned}
\delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\delta\mathbf{d}) &= \int_\Omega \nabla\mathbf{w}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}][\nabla\mathbf{u}]\,dv \\
&= \sum_i \underbrace{\int_\Omega \nabla\mathbf{w}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\varphi_i][\nabla\mathbf{u}]\,dv}_{\frac{\partial\mathbf{R}}{\partial D_i}}\,\delta D_i \qquad (76)
\end{aligned}$$

and hence the discretized sensitivity is

$$\frac{\partial\tilde{\Theta}}{\partial\mathbf{D}} = \frac{\partial\Theta}{\partial\mathbf{D}} - \frac{\partial\mathbf{R}}{\partial\mathbf{D}} \qquad (77)$$

The graph of the QoI evaluation and its adjoint sensitivity analysis is depicted in Figure 7.

Here we did an adjoint, i.e. backward, sensitivity analysis. If we had instead performed a direct, i.e. forward sensitivity problem, then instead of solving the adjoint problem of (78) for $\mathbf{w}$ we would differentiate the primal problem (63) wrt. D_i and solve the resulting pseudo problems: Find $\partial\mathbf{u}/\partial D_i \in \mathcal{H}$ such that

$$0 = a(\frac{\partial\mathbf{u}}{\partial D_i},\mathbf{w},\mathbf{d}) + \delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\frac{\partial\mathbf{d}}{\partial D_i}) \qquad (78)$$

for all $\mathbf{w}\in\mathcal{H}$. And hence we call $\delta_d r(\mathbf{u},\cdot,\mathbf{d};\frac{\partial\mathbf{d}}{\partial D_i})$ the *pseudo load linear form* and hence we also call $\delta_d r(\mathbf{u},\mathbf{w},\mathbf{d};\cdot)$ the pseudo load linear form. Back to the task at hand, upon evaluating $\partial\mathbf{d}/\partial D_i$ in the pseudo analysis, we evaluate the derivative

$$\frac{\partial\tilde{\theta}}{\partial D_i} = \delta_d\theta(\mathbf{u},\mathbf{d};\frac{\partial\mathbf{d}}{\partial D_i}) + \delta_u\theta(\mathbf{u},\mathbf{d};\frac{\partial\mathbf{u}}{\partial D_i}) \qquad (79)$$

[4] Note: If we utilize the fact that $\mathbf{w} = 2\,\mathbf{u}$ from the Footnote , then $\delta\tilde{\theta}(\mathbf{d};\delta\mathbf{d}) = -\int_\Omega \nabla\mathbf{u}\cdot\frac{\partial\mathbb{C}(\mathbf{d})}{\partial\mathbf{d}}[\delta\mathbf{d}][\nabla\mathbf{u}]\,dv$. Again, this simplification is a QoI-specific result that does not generalize to other QoIs.

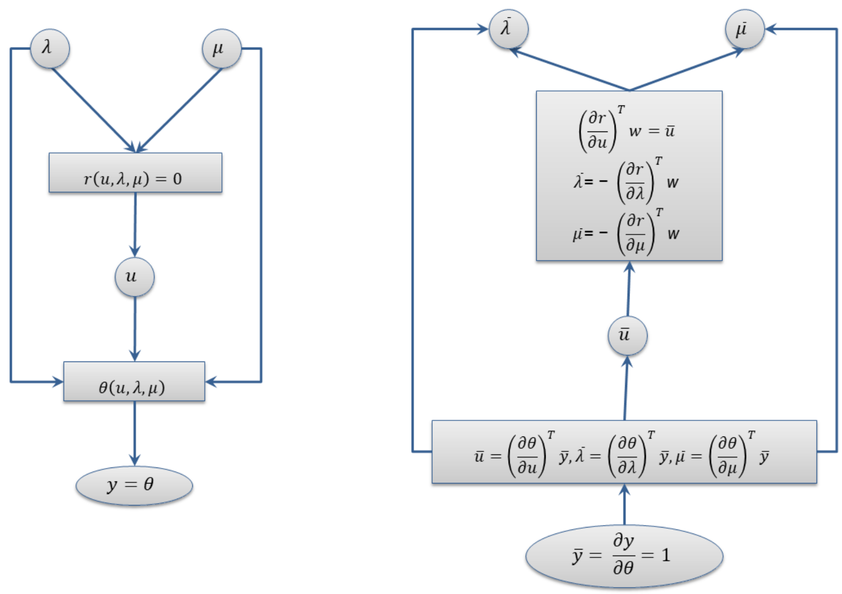


Figure 7: Compliance problem illustration.

6 Possible LiDO 2.0/MFEM/SMITH sensitivity implementation for nonlinear elasticity

Consider the nonlinear elasticity problem where we are to find the displacement $\mathbf{u}$ for the given design d, First Piola Kirchhoff constitutive relation $\mathbf{P}(\nabla\mathbf{u},\mathbf{d})$, body load $\mathbf{b}$, prescribed surface traction $\mathbf{t}^p$ over A^t and the prescribed displacement $\mathbf{u}^p$ over $A^u=\partial\Omega\backslash A^t$ such that

$$r(\mathbf{u},\mathbf{w},\mathbf{d})=0=\int_\Omega(\nabla\mathbf{w}\cdot\mathbf{P}(\nabla\mathbf{u},\mathbf{d})-\mathbf{w}\cdot\mathbf{b}(\mathbf{d}))\,dv-\int_{A^t}\mathbf{w}\cdot\mathbf{t}^p(\mathbf{d})\,da \tag{80}$$

for all kinematically admissible $\mathbf{w}$. In MFEM we parameterize the displacements $\mathbf{u}$ and $\mathbf{w}$ so that,

$$\begin{aligned}\mathbf{u}(\mathbf{x})&=\sum_i\mathbf{U}_i\,\phi_i(\mathbf{x})\\ \mathbf{w}(\mathbf{x})&=\sum_i\mathbf{W}_i\,\phi_i(\mathbf{x})\end{aligned} \tag{81}$$

Inserting these approximations into (80) gives

$$r(\mathbf{u},\mathbf{w},\mathbf{d})=\sum_k\mathbf{W}_k^T\overbrace{\left(\int_\Omega(\nabla\phi_k\cdot\mathbf{P}(\nabla\mathbf{u},\mathbf{d})-\phi_k\cdot\mathbf{b}(\mathbf{d}))\,dv-\int_{A^t}\phi_k\cdot\mathbf{t}^p(\mathbf{d})\,da\right)}^{\mathbf{R}_k} \tag{82}$$

In the above $\mathbf{R}_k$ is a (block) component of the *MFEM nonlinearform physics residual vector.*

To solve the nonlinear (80) we use Newton's method and find the kinematically admissible incremental displacement $\Delta\mathbf{u}$ such that

$$a(\Delta\mathbf{u},\mathbf{w},\mathbf{d}) = \delta r(\mathbf{u},\mathbf{w},\mathbf{d};\Delta\mathbf{u}) = \int_\Omega \nabla\mathbf{w}\cdot\frac{\partial\mathbf{P}}{\partial\nabla\mathbf{u}}[\nabla\Delta\mathbf{u}]\,dv = -r(\mathbf{w},\mathbf{u},\mathbf{d}) \tag{83}$$

for all kinematicallt admissible $\mathbf{w}$. Having $\Delta\mathbf{u}$ we update $\mathbf{u}\leftarrow\mathbf{u}+\Delta\mathbf{u}$. Iterations continue until $r(\mathbf{u},\mathbf{w},\mathbf{d})\approx 0$. Inserting the (81) approximation into (83) gives

$$a(\Delta\mathbf{u},\mathbf{w},\mathbf{d}) = \delta r(\mathbf{u},\mathbf{w},\mathbf{d};\Delta\mathbf{u}) = \sum_{k,j}\mathbf{W}_k^T\overbrace{\int_\Omega \nabla\phi_k\cdot\frac{\partial\mathbf{P}}{\partial\nabla\mathbf{u}}[\nabla\phi_j]\,dv}^{\mathbf{K}_{kj}}\,\Delta\mathbf{U}_j \tag{84}$$

In the above $\mathbf{K}_{kj}$ is a (block) component of the *MFEM nonlinearform physics tangent stiffness matrix*. Using these MFEM vectors and matrix we solve

$$\mathbf{K}\,\Delta\mathbf{U} = -\mathbf{R} \tag{85}$$

and update $\mathbf{U}\leftarrow\mathbf{U}+\Delta\mathbf{U}$ until $|\mathbf{R}|\approx 0$.

Having obtained the response $\mathbf{u}$ we now evaluate the pseudo load, i.e. the derivative of the residual wrt. the design $\mathbf{d}$. But the design is a field, i.e. a function of position, so we instead take the variation, i.e. we evaluate

$$\delta r(\mathbf{u},\mathbf{w},\mathbf{d};\Delta\mathbf{d}) = 0 = \int_\Omega (\nabla\mathbf{w}\cdot\frac{\partial\mathbf{P}}{\partial\mathbf{d}}[\delta\mathbf{d}] - \mathbf{w}\cdot\frac{\partial\mathbf{b}}{\partial\mathbf{d}}[\delta\mathbf{d}])\,dv - \int_{A^t}\mathbf{w}\cdot\frac{\partial\mathbf{t}^p}{\partial\mathbf{d}}[\delta\mathbf{d}]\,da \tag{86}$$

At this point, we again emphasize that the design $\mathbf{d}$ is a *field* which can be interpolated as any other MFEM Coefficient/GridFunction etc. As such, we can express

$$\mathbf{d}(\mathbf{x}) = \sum_i \mathbf{D}_i\,\varphi_i(\mathbf{x}) \tag{87}$$

Note that the basis functions ϕ_i that are used to interpolate $\mathbf{u}$ and $\mathbf{w}$ may be different than the φ_i that are used to interpolate $\mathbf{d}$. With this parameterization we now express the pseudo load as

$$\delta r(\mathbf{u},\mathbf{w},\mathbf{d};\Delta\mathbf{u}) = \sum_{k,j}\mathbf{W}_k^T\overbrace{\left(\int_\Omega (\nabla\phi_k\cdot\frac{\partial\mathbf{P}}{\partial\mathbf{d}}\,\varphi_j - \phi_k\cdot\frac{\partial\mathbf{b}}{\partial\mathbf{d}}\,\varphi_j)\,dv - \int_{A^t}\phi_k\cdot\frac{\partial\mathbf{t}^p}{\partial\mathbf{d}}\,\varphi_j\,da\right)}^{\frac{\partial\mathbf{R}_k}{\partial\mathbf{D}_j}}\delta\mathbf{D}_j \tag{88}$$

In the above $\partial\mathbf{R}_k/\partial\mathbf{D}_j$ is a (block) component of the *MFEM nonlinearform physics pseudo load* ***matrix***.

At this point we can evaluate the QoI and its derivatives. The generic QoI i is expressed as

$$\tilde{\theta}_i(\mathbf{d}) = \theta_i(\mathbf{u},\mathbf{d}) = \int_\Omega \pi_i(\nabla\mathbf{u},\mathbf{u},\mathbf{d})\,dv + \int_{\partial\Omega}\gamma_i(\mathbf{u},\mathbf{t},\mathbf{d})\,da \tag{89}$$

In the above $\tilde{\Theta}_i = \tilde{\theta}_i(\mathbf{d})$ is the *MFEM nonlinearform QoI i*.

The adjoint load is obtained from the variation of θ_i wrt. $\mathbf{u}$, i.e.

$$\delta\theta_i(\mathbf{u},\mathbf{d};\delta\mathbf{u}) = \int_\Omega \left(\frac{\partial \pi_i}{\partial \nabla \mathbf{u}} \cdot \nabla \delta \mathbf{u} + \frac{\partial \pi_i}{\partial \mathbf{u}} \cdot \delta \mathbf{u} \right) dv + \int_\Omega \frac{\partial \gamma_i}{\partial \mathbf{u}} \cdot \delta \mathbf{u}\, da \tag{90}$$

where the adjoint prescribe displacement satisfies $\mathbf{w} = -\partial\gamma_i/\partial\mathbf{t}$ on A^u. We discretize the above as

$$\delta\theta_i(\mathbf{u},\mathbf{d};\delta\mathbf{u}) = \sum_j \overbrace{\left(\int_\Omega \left(\frac{\partial \pi_i}{\partial \nabla \mathbf{u}} \cdot \nabla \phi_j + \frac{\partial \pi_i}{\partial \mathbf{u}} \cdot \phi_j \right) dv + \int_\Omega \frac{\partial \gamma_i}{\partial \mathbf{u}} \cdot \phi_j\, da \right)}^{\frac{\partial \Theta_i}{\partial \mathbf{U}_j}} \delta \mathbf{U}_j \tag{91}$$

and adjoint prescribed displacement satisfies $\mathbf{w}(\mathbf{x}) = -\sum_i \left(\partial\gamma_i/\partial\mathbf{t}\right)_i \phi_i(\mathbf{x})$ on A^u where $\left(\partial\gamma_i/\partial\mathbf{t}\right)_i = \phi_i^*(\partial\gamma_i/\partial\mathbf{t})$ with ϕ_i^* being the dual basis functional corresponding to the basis element ϕ_i. In the above $\frac{\partial \Theta_i}{\partial \mathbf{U}_j}$ is a (block) component of the *MFEM nonlinearform QoI i adjoint load vector*. We next express the variation of θ_i wrt. $\mathbf{d}$ and subsequently discretize it as

$$\begin{aligned} \delta\theta_i(\mathbf{u},\mathbf{d};\delta\mathbf{d}) &= \int_\Omega \frac{\partial \pi_i}{\partial \mathbf{d}} \delta \mathbf{d}\, dv + \int_{\partial\Omega} \frac{\partial \gamma_i}{\partial \mathbf{d}}\, \delta \mathbf{d}\, da \\ &= \sum_j \overbrace{\left(\int_\Omega \frac{\partial \pi_i}{\partial \mathbf{d}} \varphi_j\, dv + \int_{\partial\Omega} \frac{\partial \gamma_i}{\partial \mathbf{d}}\, \varphi_j\, da \right)}^{\frac{\partial \Theta_i}{\partial \mathbf{D}_j}} \delta \mathbf{D}_j \end{aligned} \tag{92}$$

In the above $\partial\Theta_i/\partial\mathbf{D}_j$ is a component of the *MFEM nonlinearform QoI i explicit sensitivity vector*.

Having computed all of the above MFEM vectors and matrices we now proceed with the adjoint sensitivity analysis. We solve the adjoint problem

$$\mathbf{K}^T\, \mathbf{W} = \frac{\partial \Theta_i}{\partial \mathbf{U}} \tag{93}$$

for $\mathbf{W}$ and then evaluate the sensitivity

$$\frac{\partial \tilde{\Theta}_i}{\partial \mathbf{D}} = \frac{\partial \Theta_i}{\partial \mathbf{D}} - \left(\frac{\partial \mathbf{R}}{\partial \mathbf{D}} \right)^T \mathbf{W} \tag{94}$$

The process is summarized in Figure **??**.

Notes:

- The algorithm can easily accommodate multiple QoIs Θ_i by placing a loop over the bottom blocks, cf. Figure **??**. Notably, the pseudo load matrix is only computed once; albeit storing the $\partial\mathbf{R}/\partial\mathbf{D}$ matrix may be problematic. Another option is to eliminate the $\partial\mathbf{R}/\partial\mathbf{D}$ block and insert a $(\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}$ block after the

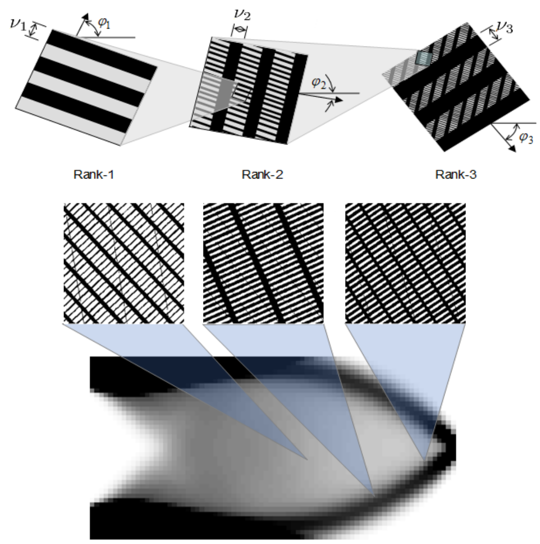


Figure 8: Rank 3 laminate optimization of a cantilever beam.

adjoint solve, i.e. replace the matrix computation of $\partial\mathbf{R}/\partial\mathbf{D}$ with the vector computation $\left\{\ (\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}\ \right\}$ where

$$\delta r(\mathbf{u},\mathbf{w},\mathbf{d};\delta\mathbf{d}) = \sum_j \overbrace{\left(\int_\Omega (\nabla\mathbf{w}\cdot\frac{\partial\mathbf{P}}{\partial\mathbf{d}}\,\varphi_j - \mathbf{w}\cdot\frac{\partial\mathbf{b}}{\partial\mathbf{d}}\,\varphi_j)\,dv - \int_{A^t}\mathbf{w}\cdot\frac{\partial\mathbf{t}^p}{\partial\mathbf{d}}\,\varphi_j\,da\right)}^{((\frac{\partial\mathbf{R}}{\partial\mathbf{D}})^T\mathbf{W})_j}\delta\mathbf{D}_j \tag{95}$$

is a block vector contribution to the *vector* $\left\{\ (\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}\ \right\}$. Obviously the vector $\left\{\ (\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}\ \right\}$ requires much less storage than the matrix $\left[\ \partial\mathbf{R}/\partial\mathbf{D}\ \right]$; however the vector $(\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}$ must be computed for each QoI as opposed to computing the matrix $\left[\ \partial\mathbf{R}/\partial\mathbf{D}\ \right]$ once and performing a matrix-vector product $(\partial\mathbf{R}/\partial\mathbf{D})^T\mathbf{W}$ for each QoI, cf. Figures **??** and 9.

- Multiphysics simulations can be accommodated by partitioning the response as $\mathbf{U} = \left[\ \mathbf{U}_1^T\quad \mathbf{U}_2^T\cdots\ \right]^T$. The subsequent partitioning of $\mathbf{R}$ and $\mathbf{K}$ follow from that of $\mathbf{U}$. The coupling may be strong or weak; the latter will simplify the analysis procedure.

- In the above we assume $\mathbf{d} = \left[\ d_1\quad d_2\quad \cdots\ \right]^T = \sum\mathbf{D}_i\,\varphi_i$ is a vector field whose components represent, e.g. the layer volume fractions ν_i and orientations ϕ_i in a ranked laminate, cf. Figure 8, or the volume fraction in a SIMP material, or the fiber orientations in a fiber reinforced composite, or ... This design vector field $\mathbf{d}$ is akin to a displacement field $\mathbf{u}$ and hence should be codeable via MFEM's vectorCoefficients/VectorGridFunctions.

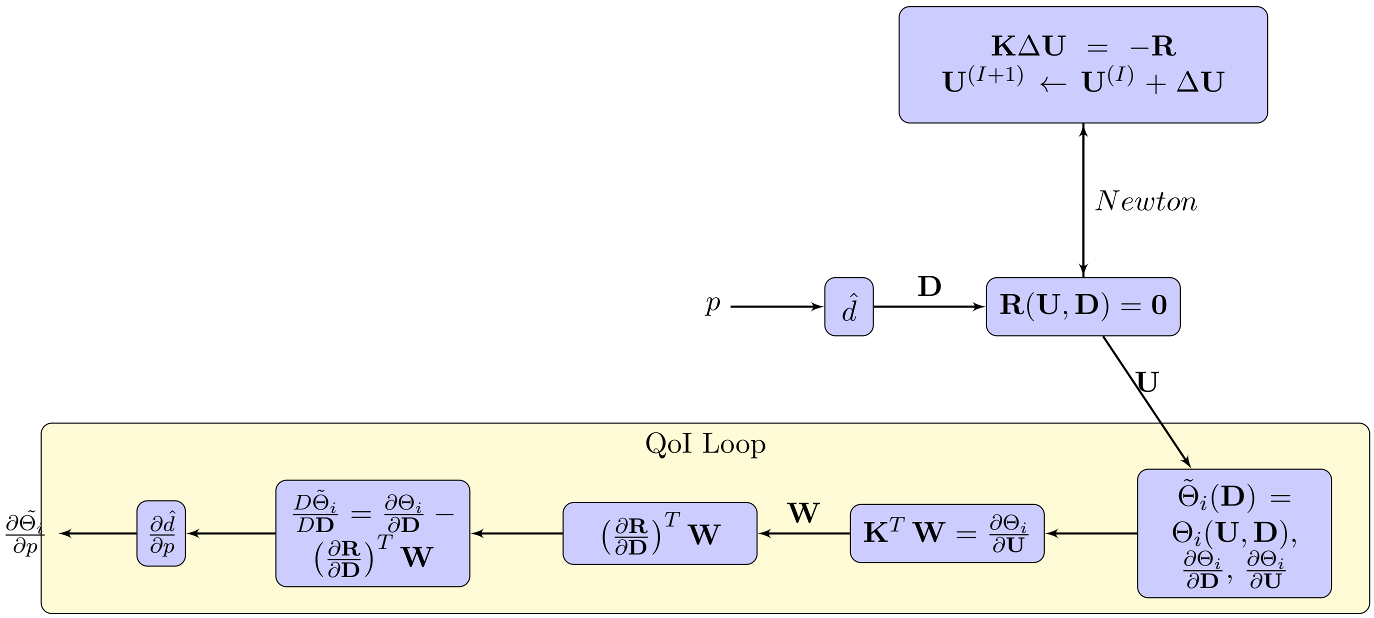


Figure 9: Low storage MFEM analysis and sensitivity analysis option.

- In some regions the design is fixed, i.e. prescribed. We can enforce these constraints by treating them as we do Dirichlet boundary conditions in MFEM.

- If different design fields use different parameterizations, e.g. $\mathbf{d}_1(\mathbf{x}) = \sum \mathbf{D}_{1_i}\varphi_i(\mathbf{x})$ and $\mathbf{d}_2(\mathbf{x}) = \sum \mathbf{D}_{2_i}\hat{\varphi}_i(\mathbf{x})$, then the pseudo load matrix $\frac{\partial\mathbf{R}}{\partial\mathbf{D}}$ and explicit derivative vector $\frac{\partial\Theta_i}{\partial\mathbf{D}}$ will have to be evaluated multiple times. For example, consider a design with the SIMP material $\mathbf{P} = \nu^p\,\mathbb{C}[\nabla\mathbf{u}]$ where $\mathbb{C} = 2\,\mu\mathbb{I} + \lambda\,\mathbf{I}\otimes\mathbf{I}$. We might have both the volume fraction ν and Lame parameters (μ,λ) be design fields. The volume fraction may change over the domain so we would express it as $\nu(\mathbf{x}) = \sum_i N_i\,\varphi_i(\mathbf{x})$, but the Lame parameters may be uniform so we would simply express them as $(\mu(\mathbf{x}),\lambda(\mathbf{x})) = (\mu,\lambda)$; to obtain the MFEM parameterization $(\mu,\lambda) = \sum \left\{ \begin{array}{c} M_i \\ \Lambda_i \end{array} \right\} \varphi_i(\mathbf{x})$ we can use an MFEM projection operation. This multiple basis function usage should be possible, as MFEM can already solve incompressible elasticity problems in which displacement and pressure are interpolated with different basis functions.

- Having $\frac{\partial\Theta_i}{\partial\mathbf{D}}$ we can then use the chain-rule to obtain the sensitivity wrt. higher-level parameters, e.g. b-spline coefficients and/or nonfiltered parameters (which will require another adjoint sensitivity analysis), cf. Section 7.

- While it is possible to “mix” parameterization levels, e.g. by using filtered and unfiltered volume fractions in the analysis and QoI definitions, I personally don’t see the need for it especially in light of the complications in the sensitivity analysis and chain rule applications that such a mixture will produce.

- For some design fields certain derivatives will be zero and others will not. E.g. for a volume fraction field ν we expect $\partial \mathbf{P}/\partial\nu \neq \mathbf{0}$ and $\partial \mathbf{b}/\partial\nu \neq \mathbf{0}$. But for a fiber orientation field α we expect $\partial \mathbf{P}/\partial\alpha \neq \mathbf{0}$ but $\partial \mathbf{b}/\partial\alpha = \mathbf{0}$. Can these zero derivatives be identified a priori to hasten computations? Would the computational savings be worth the trouble of doing this hastening?

  For example, Figure 10 shows the topology optimization of the wheel. The wheel consists of 2 regions, the green region which consists of a fixed isotropic steel material where $\mathbf{P} = \mathbb{C}[\nabla \mathbf{u}]$ does not depend on the design so that $\partial \mathbf{P}/\partial\nu = \mathbf{0}$ and the magenta region which consists of the SIMP isotropic steel material $\mathbf{P} = \nu^p\, \mathbb{C}[\nabla \mathbf{u}]$ which does depend on the design field ν so that $\partial \mathbf{P}/\partial\nu \neq \mathbf{0}$. Note that five fold cyclic symmetry on ν is enforced in the magenta region. HOW DO WE DO THIS IN MFEM?

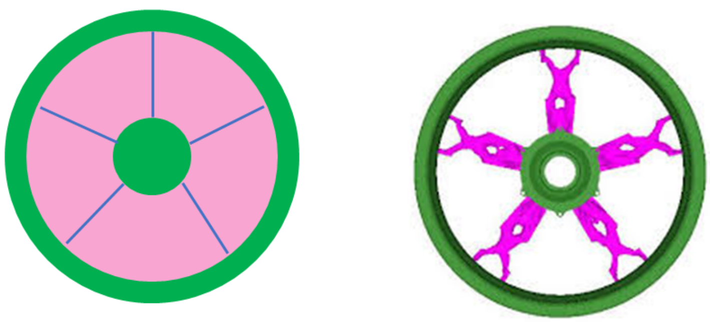


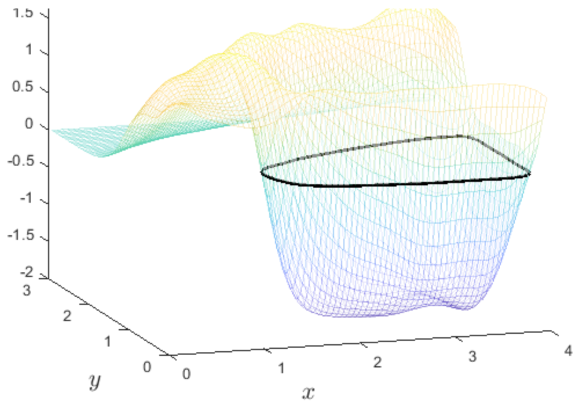
Figure 10: Multmaterial optimization model, initial design (left) and optimized design (right).

7 Design parameterization

Here we expound the types of design parameterization $\hat{d} : \Omega \times \mathcal{R}^n \to \mathcal{R}$.

7.1 Level set: Fourier series, b-splines, bezier splines, r-functions, ...

A great many of parameterizations can be viewed as level set functions, e.g. Fourier series, b-splines, bezier splines, geometric projection, r-functions, ... In this approach, $\hat{d} : \Omega \times \mathcal{R}^n \to \mathcal{R}$ is viewed as a level set function in that its contours, i.e. $\hat{d}(x) = c_k$ take on specific meaning. For example in reinforced composite structures the level set value may correspond to the fiber orientation angle and in metamaterial structures it may correspond to the strut radii. This is illustrated in Figure 11 where the function $\hat{d}$ is obtained from a b-spline surface defined over a 4×5 grid of control points. The heights of the 20 control points are the design parameters p.



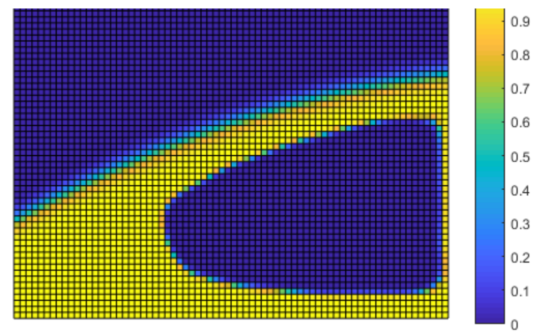


Figure 11: Level set function $\hat{d}$ with $\hat{d}(x) = 0$ level set (left) and volume fraction function $\nu = H_{k,z} \circ \hat{d}$ (right).

7.2 Element-based design parameterization and filtering

As an example of an element-based design parameterization, a volume fraction design parameter p_e is assigned to each element Ω_e in the mesh. The *nominal, or base, design field* $d : \Omega \times \mathcal{R}^n \to \mathcal{R}$ from this *parameterization* is piecewise uniform over the elements such that

$$d(x) = \hat{d}(x, p) = p_e \text{ for } x \in \Omega_e \tag{96}$$

As such $d \in L^2$.

It is well known that such parameterizations produce anomalies in the optimized design and often leads to ill-posed optimization problems. For this reason it is common to restrict the design space by imposing some amount of smoothness. We obtain a smoothed design field via a two step process. To begin we obtain the smooth H^1 field ν_f (where the subscript f denotes the filtration of the nominal field d) by solving the "*filter PDE*" equation for the kinematically admissible $\nu_f \in H^1$ that satisfies

$$\begin{aligned} r_f(\nu_f, d) = 0 \quad &= \quad \int_\Omega \left(r^2 \, \nabla\alpha \cdot \nabla\nu_f + \alpha \left(\nu_f - d\right)\right) \, dv \\ &= \quad \underbrace{\int_\Omega \left(r^2 \, \nabla\alpha \cdot \nabla\nu_f + \alpha \, \nu_f\right) \, dv}_{a_f(\nu_f, \alpha)} - \underbrace{\int_\Omega \alpha \, d \, dv}_{l_f(\alpha)} \end{aligned} \tag{97}$$

for all kinematically admissible $\alpha \in H^1$. Now we *project* the H^1 field ν_f onto the L^2 space to define design the filtered projected field $\nu_p \in L^2$. In this way, the filtered-projected volume fraction $\nu_p \in L^2$ is the smoothed version of the nominal design field $d \in L^2$; it is uniform over the finite elements so it is piecewise uniform over the domain Ω, and it exhibits smoother spatial variations versus d. Back to the task at hand. To project $\nu_f \in H_1$ onto the L^2

space to compute $\nu_p \in L^2$ we solve the minimization problem:

$$\min_{\nu_p \in L^2} \quad \tfrac{1}{2} \int_\Omega \left(\nu_p - \nu_f\right)^2 \, dv \tag{98}$$

Stationarity of the above tells us that $\nu_p \in L^2$ must satisfy

$$\begin{aligned} r_p(\nu_p, \beta) = 0 \quad &= \quad \int_\Omega \beta \ (\nu_p - \nu_f) \ dv \\ &= \quad \underbrace{\int_\Omega \beta \, \nu_p \, dv}_{a_p(\nu_p,\beta)} - \underbrace{\int_\Omega \beta \, \nu_f \, dv}_{\ell_p(\beta)} \end{aligned} \tag{99}$$

for all $\beta \in L^2$. Note that in the above the stiffness matrix associated with the bilinear form a_p is diagonal.

The filtered and projected volume fraction $\nu_p \in L^2$ is used to define the Poisson's ratio, i.e. $\mathsf{v} = \mathsf{v}(\nu_p)$ and shear modulus $\mu = \mu(\nu_p)$ which in turn define the elasticity tensor $\mathbb{C}$. Without loss of generality, we assume that the body force $\mathbf{b}$ and applied traction $\mathbf{t}^p$ are *not* functions of the design.

Having the elasticity tensor $\mathbb{C}$, body load $\mathbf{b}$ and applied traction $\mathbf{t}^p$ we can compute the displacement $\mathbf{u}$ by solving the elasticity equation of finding the admissible $\mathbf{u}$ such that

$$\begin{aligned} r_u(\mathbf{u}, \mathbf{w}) = 0 \quad &= \quad \int_\Omega (\nabla \mathbf{w} \cdot \mathbb{C}[\nabla \mathbf{u}] - \mathbf{w} \cdot \mathbf{b}) \, dv - \int_{A^t} \mathbf{w} \cdot \mathbf{t}^p \, da = 0 \\ &= \quad \underbrace{\int_\Omega \nabla \mathbf{w} \cdot \mathbb{C}[\nabla \mathbf{u}] \, dv}_{a_u(\mathbf{u},\mathbf{w})} - \underbrace{\int_\Omega \mathbf{w} \cdot \mathbf{b}) \, dv - \int_{A^t} \mathbf{w} \cdot \mathbf{t}^p \, da}_{\ell_u(\mathbf{w})} = 0 \end{aligned} \tag{100}$$

for all kinematically admissible $\mathbf{w}$.

Finally, having the displacement $\mathbf{u}$, volume fraction fields d, ν_f and ν_p and material properties v and μ we can evaluate any QoI of the form

$$\theta(\mathbf{u}, d, \nu_f, \nu_p, \mathsf{v}, \mu) = \int_\Omega \pi(\mathbf{u}, d, \nu_f, \nu_p, \mathsf{v}, \mu) \, dv \tag{101}$$

Now we commence with the sensitivity analysis via the graphical approach. First we evaluate the one forms:

$$
\begin{aligned}
\bar{\mathbf{u}}(\delta\mathbf{u}) &= \int_\Omega \frac{\partial\pi}{\partial\mathbf{u}}\,\delta\mathbf{u}\,dv \\
\bar{d}(\delta d) &= \int_\Omega \frac{\partial\pi}{\partial d}\,\delta d\,dv \\
\bar{\nu}_f(\delta\nu_f) &= \int_\Omega \frac{\partial\pi}{\partial\nu_f}\,\delta\nu_f\,dv \\
\bar{\nu}_p(\delta\nu_p) &= \int_\Omega \frac{\partial\pi}{\partial\nu_p}\,\delta\nu_p\,dv \\
\bar{\boldsymbol{\nu}}(\delta\boldsymbol{\nu}) &= \int_\Omega \frac{\partial\pi}{\partial\boldsymbol{\nu}}\,\delta\boldsymbol{\nu}\,dv \\
\bar{\mu}(\delta\mu) &= \int_\Omega \frac{\partial\pi}{\partial\mu}\,\delta\mu\,dv
\end{aligned}
\tag{102}
$$

The discretized one forms are MFEM linear forms. E.g. for $\delta\mathbf{u}(x) = \sum^I \delta\mathbf{u}^I\,\phi^I(x)$, the coefficient of $\delta\mathbf{u}^I$ is a component of the MFEM $\bar{\mathbf{u}}$ linear form.

Next we solve an adjoint problem for the kinematically admissible $\mathbf{w}_u$

$$
a_u(\delta\mathbf{u}, \mathbf{w}_u) = \bar{\mathbf{u}}(\delta\mathbf{u}) \tag{103}
$$

for all kinematically admissible $\delta\mathbf{u}$. Subsequently we evaluate the pseudo load contribution to the one forms $\bar{\boldsymbol{\nu}}$ and $\bar{\mu}$, i.e.

$$
\begin{aligned}
\bar{\boldsymbol{\nu}}(\delta\boldsymbol{\nu}) &= -\int_\Omega \nabla\mathbf{w}_u \cdot \frac{\partial\mathbb{C}}{\partial\boldsymbol{\nu}}[\nabla\mathbf{u}]\,\delta\boldsymbol{\nu}\,dv \\
\bar{\mu}(\delta\mu) &= -\int_\Omega \nabla\mathbf{w}_u \cdot \frac{\partial\mathbb{C}}{\partial\mu}[\nabla\mathbf{u}]\,\delta\mu\,dv
\end{aligned}
\tag{104}
$$

As described above, for $\delta\boldsymbol{\nu}(x) = \sum_I \delta\boldsymbol{\nu}^I\,\psi^I(x)$, the coefficient of $\delta\boldsymbol{\nu}^I$ is a component of the MFEM $\bar{\boldsymbol{\nu}}$ linear form.

Two more contributions to the one form $\bar{\nu}_p$ follow from the direct computations

$$
\begin{aligned}
\bar{\nu}_p(\delta\nu_p) &= \int_\Omega \left(\frac{\partial\boldsymbol{\nu}}{\partial\nu_p}\right)^T \bar{\boldsymbol{\nu}}\,\delta\nu_p\,dv \\
\bar{\nu}_p(\delta\nu_p) &= \int_\Omega \left(\frac{\partial\mu}{\partial\nu_p}\right)^T \bar{\mu}\,\delta\mu\,dv
\end{aligned}
\tag{105}
$$

Note that I wrote these as integrals to be consistent with the MFEM form language.

Next we solve another adjoint problem for the kinematically admissible w_p

$$
a_p(\delta\nu_p, w_p) = \bar{\nu}_p(\delta\nu_p) \tag{106}
$$

for all kinematically admissible $\delta\nu_p$. Subsequently we evaluate the pseudo load contribution to the one form $\bar{\nu}_f$, i.e.

$$\bar{\nu}_f(\delta\nu_f) = \int_\Omega w_p \, \delta\nu_f \, dv \tag{107}$$

As described above, $\bar{\nu}_f$ forms another MFEM linear form.

And this is followed by yet another another adjoint problem for the kinematically admissible w_f

$$a_f(\delta\nu_f, w_f) = \bar{\nu}_f(\delta\nu_f) \tag{108}$$

for all kinematically admissible $\delta\nu_f$. Subsequently we evaluate the pseudo load contribution to the one form $\bar{d}$, i.e.

$$\bar{d}(\delta d) = \int_\Omega w_f \, \delta d \, dv \tag{109}$$

where $\bar{d}$ forms yet another MFEM linear form.

We are now done! Figure 12 depicts the analysis and sensitivity analysis.

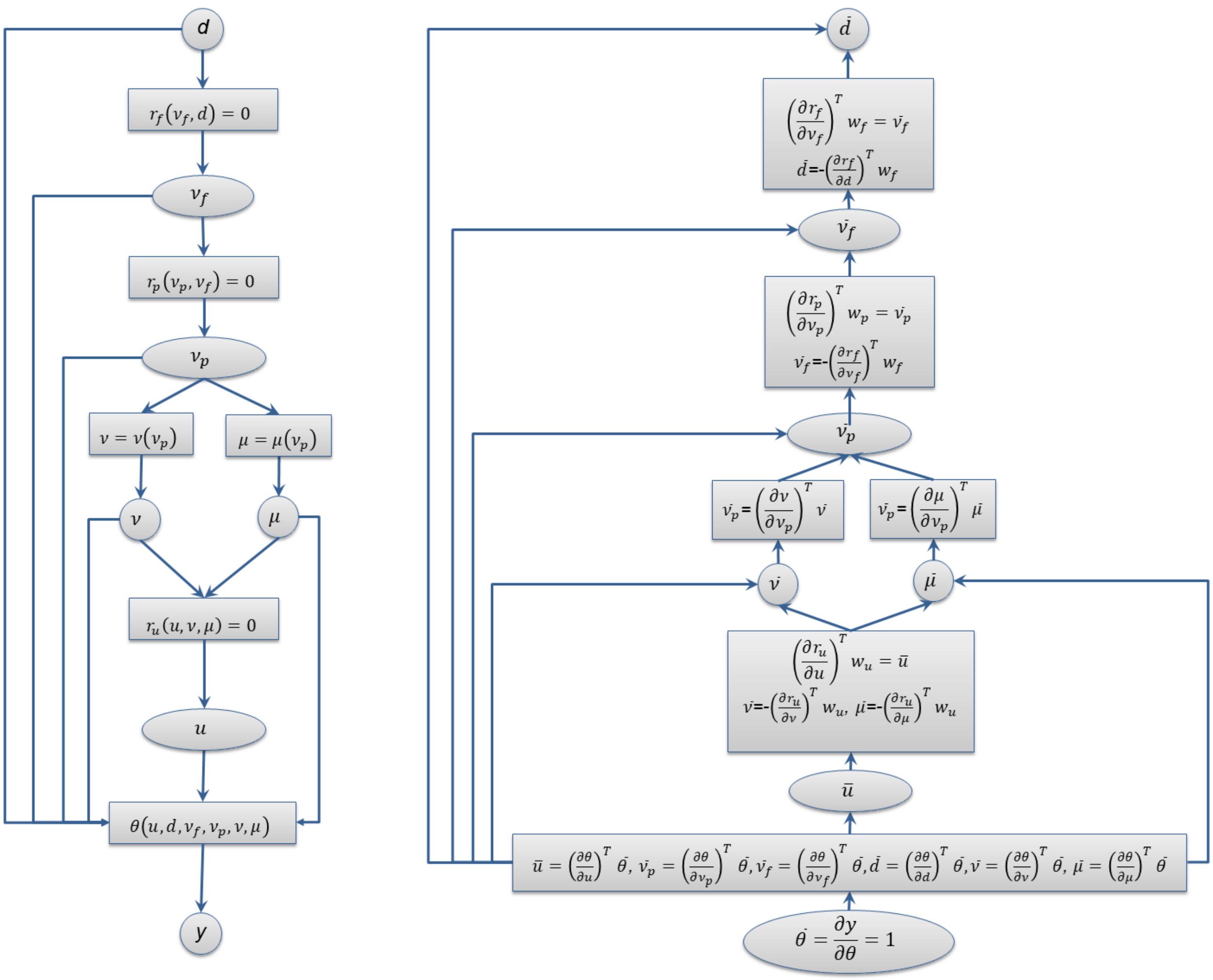


Figure 12: Analysis and adjoint sensitivity analysis with filtering and projection.

8 Example QoIs

The QoIs that are used to define the cost and constraint functions of item **??** in Section **??** can take on very general forms. Here we provide some example QoIs. We assume that only the material properties are functions of the design. For clarity, here and henceforth, Greek letters denote scalar fields, Latin lowercase letters denote vector fields and Latin uppercase letters denote 2-tensor fields.

1. Linear elasticity, compliance (version 1)
   - QoI: $\tilde{\theta}_1(d) = \theta_1(\mathbf{u}, d) = \int \mathbf{u} \cdot \mathbf{b} \, dv + \int_{A^n} \mathbf{u} \cdot \mathbf{t}^p \, da$ [5]
   - Adjoint load: [6] $\delta\theta_1(\mathbf{u}, d; \delta\mathbf{u}) = \int \delta\mathbf{u} \cdot \mathbf{b} \, dv + \int_{A^n} \delta\mathbf{u} \cdot \mathbf{t}^p \, da$
   - Explicit derivative: $\delta\theta_1(\mathbf{u}, d; \delta d) = \int \mathbf{u} \cdot \delta\mathbf{b}(d; \delta d) \, dv$
2. Linear elasticity, compliance (version 2)
   - QoI: $\tilde{\theta}_2(d) = \theta_2(\mathbf{u}, d) = \int \nabla\mathbf{u} \cdot \mathbb{C}[\nabla\mathbf{u}] \, dv$ where $\mathbb{C} = \partial^2\psi/\partial(\nabla\mathbf{u})^2)$ is the elasticity tensor.
   - Adjoint load: [7] $\delta\theta_2(\mathbf{u}, d; \delta\mathbf{u}) = \int \nabla\delta\mathbf{u} \cdot 2\,\mathbb{C}[\nabla\mathbf{u}] \, dv$
   - Explicit derivative: $\delta\theta_2(\mathbf{u}, d; \delta d) = \int \nabla\mathbf{u} \cdot \frac{\partial\mathbb{C}}{\partial d}[\nabla\mathbf{u}] \, \delta d \, dv$
3. Displacement magnitude
   - QoI: $\tilde{\theta}_3(d) = \theta_3(\mathbf{u}, d) = \int \|\mathbf{u}\| \, dv = \int (\mathbf{u} \cdot \mathbf{u})^{\frac{1}{2}} \, dv$
   - Adjoint load: $\delta\theta_3(\mathbf{u}, d; \delta\mathbf{u}) = \int \delta\mathbf{u} \cdot \frac{1}{\|\mathbf{u}\|}\mathbf{u} \, dv$
   - Explicit derivative: $\delta\theta_3(\mathbf{u}, d; \delta d) = 0$
4. Linear elasticity: VonMises stress measure
   - The vonMises stress $\sigma_{VM} = \sqrt{3\,J_2}$ where $J_2 = 1/2\,\mathbf{P}^D \cdot \mathbf{P}^D$ where $\mathbf{P} = \mathbb{C}[\nabla\mathbf{u}]$ is the stress, $\mathbf{P}^D = \mathbb{I}_D[\mathbf{P}] = \mathbf{P} - \frac{1}{3}\mathrm{tr}\mathbf{P}\,\mathbf{I}$ is the deviatoric stress and $\mathbb{I}_D = \mathbb{I} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}$ is the deviator projection tensor defined such that $\mathbb{I}_D[\mathbf{A}] = \mathbf{A} - \frac{1}{3}\mathrm{tr}\mathbf{A}\,\mathbf{I}$ for any tensor $\mathbf{A}$. Yielding occurs when $\sigma_{VM} > \bar{\sigma}$ where $\bar{\sigma}$ is the uniaxial yield stress. Note that

     $$\begin{aligned} \frac{\partial\sigma_{VM}}{\partial\mathbf{P}} &= \sqrt{\frac{3}{2}} \frac{\mathbf{I}_D[\mathbf{P}]}{\sqrt{\mathbf{P}^D.\mathbf{P}^D}} = \frac{\sqrt{3}}{2} \frac{\mathbf{P}_D}{\sqrt{J_2}} \\ \frac{\partial\mathbf{P}}{\partial\nabla\mathbf{u}} &= \mathbb{C} \end{aligned} \tag{110}$$

     where we assume $\mathbb{C}$ is symmetric. Note that the product $f'(\sigma_{VM})\,\frac{\partial\sigma_{VM}}{\partial\mathbf{P}}$ is the chain-rule derivative of $f(\sigma_{VM}(\mathbf{P}))$ wrt. $\mathbf{P}$.

[5] We assume the body load is a function of the design feild, but the applied traction is not.
[6] Note $\theta_1(\mathbf{u}, d; \delta\mathbf{u})$ equals the primal load linear form $\ell(\mathbf{w}) = \int \mathbf{w} \cdot \mathbf{b} \, dv + \int_{A^n} \mathbf{w} \cdot \mathbf{b} \, da$ and hence $\mathbf{w} = \mathbf{u}$.
[7] Note that $\int \nabla\delta\mathbf{u} \cdot 2\,\mathbb{C}[\nabla\mathbf{u}] \, dv = 2\left(\int \mathbf{u} \cdot \mathbf{b} \, dv + \int_{A^n} \mathbf{u} \cdot \mathbf{b} \, da\right)$ equals twice the primal load linear form $\ell(\mathbf{w})$ and hence $\mathbf{w} = 2\,\mathbf{u}$.

- QoI: $\tilde{\theta}_4(d) = \theta_4(\mathbf{u}, d) = \int f(\sigma_{VM})\, dv$ where f is a differentiable function, e.g. $f(x) = x^p$ for the p-norm and $f(x) = \exp(k\, x)$ for the Kreisselmeier-Steinhauser (KS) function.
- Adjoint load: $\delta\theta_4(\mathbf{u}, d; \delta\mathbf{u}) = \int \nabla\delta\mathbf{u} \cdot \mathbb{C}[f'(\sigma_{VM})\, \frac{\partial \sigma_{VM}}{\partial \mathbf{P}}]\, dv$
- Explicit derivative (wrt. material): $\delta\theta_5(\mathbf{u}, d; \delta d) = \int f'(\sigma_{VM})\, \frac{\partial \sigma_{VM}}{\partial \mathbf{P}} \cdot \frac{\partial \mathbb{C}}{\partial d}[\nabla\mathbf{u}]\, dv$
- Explicit derivative (wrt. shape): $\delta\theta_5(\mathbf{u}, d; \delta d) = -\int \nabla^T\mathbf{u}\, \mathbb{C}[f'(\sigma_{VM})\, \frac{\partial \sigma_{VM}}{\partial \mathbf{P}}] \cdot \nabla\mathbf{v}\, dv$

5. Linear elasticity: Drucker–Prager stress measure

- The Drucker–Prager stress $\sigma_{DP} = \sqrt{J_2} - B\, I_1$ where $J_2 = 1/2\, \mathbf{P}^D \cdot \mathbf{P}^D$ and $I_1 = \mathbf{I} \cdot \mathbf{P} = \mathrm{tr}\mathbf{P}$ where $\mathbf{P} = \mathbb{C}[\nabla\mathbf{u}]$ is the stress, $\mathbf{P}^D = \mathbb{I}_D[\mathbf{P}] = \mathbf{P} - \frac{1}{3}\mathrm{tr}\mathbf{P}\,\mathbf{I}$ is the deviatoric stress and $\mathbb{I}_D = \mathbb{I} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}$ is the deviator projection tensor defined such that $\mathbb{I}_D[\mathbf{A}] = \mathbf{A} - \frac{1}{3}\mathrm{tr}\mathbf{A}\,\mathbf{I}$ for any tensor $\mathbf{A}$. Failure occurs when $\sigma_{DP} > A$. The constants $A = \frac{2}{\sqrt{3}} \frac{\sigma_c\, \sigma_t}{\sigma_c + \sigma_t}$ and $B = \frac{1}{\sqrt{3}} \frac{\sigma_t - \sigma_c}{\sigma_c + \sigma_t}$ are defined by the uniaxial failure stress under compression $\sigma_c > 0$ and uniaxial failure stress under tension $\sigma_t > 0$. Note that

$$
\begin{aligned}
\frac{\partial \sigma_{DP}}{\partial \mathbf{P}} &= \frac{1}{\sqrt{2}} \frac{\mathbf{I}_D[\mathbf{P}]}{\sqrt{\mathbf{P}^D \cdot \mathbf{P}^D}} - B\,\mathbf{I} = \frac{1}{2} \frac{\mathbf{P}_D}{\sqrt{J_2}} - B\,\mathbf{I} \\
\frac{\partial \mathbf{P}}{\partial \nabla\mathbf{u}} &= \mathbb{C}
\end{aligned}
\tag{111}
$$

  where we assume $\mathbb{C}$ is symmetric. Note that the product $f'(\sigma_{DP})\, \frac{\partial \sigma_{DP}}{\partial \mathbf{P}}$ is the chain-rule derivative of $f(\sigma_{DP}(\mathbf{P}))$ wrt. $\mathbf{P}$.
- QoI: $\tilde{\theta}_5(d) = \theta_5(\mathbf{u}, d) = \int f(\sigma_{DP})\, dv$ where f is a differentiable function, e.g. $f(x) = x^p$ for the p-norm and $f(x) = \exp(k\, x)$ for the Kreisselmeier-Steinhauser (KS) function.
- Adjoint load: $\delta\theta_5(\mathbf{u}, d; \delta\mathbf{u}) = \int \nabla\delta\mathbf{u} \cdot \mathbb{C}[f'(\sigma_{DP})\, \frac{\partial \sigma_{DP}}{\partial \mathbf{P}}]\, dv$
- Explicit derivative (wrt. material): $\delta\theta_5(\mathbf{u}, d; \delta d) = \int f'(\sigma_{DP})\, \frac{\partial \sigma_{DP}}{\partial \mathbf{P}} \cdot \frac{\partial \mathbb{C}}{\partial d}[\nabla\mathbf{u}]\, dv$
- Explicit derivative (wrt. shape): $\delta\theta_5(\mathbf{u}, d; \delta d) = -\int \nabla^T\mathbf{u}\, \mathbb{C}[f'(\sigma_{DP})\, \frac{\partial \sigma_{DP}}{\partial \mathbf{P}}] \cdot \nabla\mathbf{v}\, dv$

9 Bilinear, load linear and pseudo load forms for SMITH

For future reference we denote the bilinear and load linear forms that will be needed for different physics and QoIs, i.e. cost and constraint functions.

You will notice that the pseudo load forms can be obtained using the same module that is used to compute the load linear forms for the primal analysis. However, rather than feeding in, e.g. the stress constitutive relation for $\mathbf{P}$, body load $\mathbf{b}$ and prescribed traction $\mathbf{t}^p$ we feed in their derivatives, e.g. $\delta\mathbf{P}$, $\delta\mathbf{b}$ and $\delta\mathbf{t}^p$. Depending on the discretization of d there may be many derivatives which are only nonzero over small subregions of Ω, e.g. element-wise volume fraction parameterization or only a few derivatives which are nonzero over the entirety of Ω, e.g. r-function parameters.

Unfortunately, this simplification does not carry over for shape derivatives. That said, the shape variation problem is not too hard, all things considered.

9.1 Steady-state solid mechanics (Equilibrium)

Symbol definition:

- $\mathbf{u}$: displacement
- $\mathbf{w}$: virtual displacement, adjoint displacement
- $\mathbf{b}$: Body load
- $\mathbf{t}^p$: prescribed traction
- $\mathbf{P}$: Piola Kirchhoff I stress
- $\mathbf{p} = \mathbf{P}\,\mathbf{n}$: traction
- $\mathbf{F} = \mathbf{I} + \nabla\mathbf{u}$: deformation gradient
- Ω: undeformed configuration
- A^n: undeformed configuration surface with prescribed Neumann BCs
- A^d: undeformed configuration surface with prescribed Dirichlet BCs
- $\mathbf{n}$ normal vector to undeformed configuration surface

Bilinear, load linear form and pseudo load definitions:

- residual: $r(\mathbf{u}, \mathbf{w}, d) = \ell_1(\nabla\mathbf{w}, d) + \ell_2(\mathbf{w}, d) + \ell_3(\mathbf{w}, d) = \int_\Omega \nabla\mathbf{w} \cdot \mathbf{P}\, dv - \int_\Omega \mathbf{w} \cdot \mathbf{b}\, dv - \int_{A^n} \mathbf{w} \cdot \mathbf{t}^p\, dv$
- tangent: $a_1(\nabla\Delta\mathbf{u}, \nabla\mathbf{w}) = \int_\Omega \nabla\mathbf{w} \cdot \frac{\partial\mathbf{P}}{\partial\mathbf{F}}[\nabla\Delta\mathbf{u}]\, dv$
- pseudo load linear form: $\delta r(\mathbf{u}, \mathbf{w}, d; \delta d) = \delta\ell_1(\nabla\mathbf{w}, d; \delta d) + \delta\ell_2(\mathbf{w}, d; \delta d) + \delta\ell_3(\mathbf{w}, d; \delta d) =$ $\int_\Omega \nabla\mathbf{w} \cdot \delta\mathbf{P}(\nabla\mathbf{u}, d; \delta d)\, dv - \int_\Omega \mathbf{w} \cdot \delta\mathbf{b}(d; \delta d)\, dv - \int_{A^n} \mathbf{w} \cdot \delta\mathbf{t}^p(d; \delta d)\, dv$
- Uniform pressure load:
  - load linear form $\ell_3(\mathbf{w}, d) = -\int_{A^n} \mathbf{w} \cdot \mathbf{t}^p\, da = -\int_{A^n} \mathbf{w} \cdot p \det\mathbf{F}\,\mathbf{F}^{-T}\,\mathbf{n}\, da$
  - tangent: $a_3(\nabla\Delta\mathbf{u}, \mathbf{w}, d) = -\int_{A^n} \mathbf{w} \cdot p \det\mathbf{F}\left((\mathbf{F}^{-T} \cdot \nabla\Delta\mathbf{u})\mathbf{F}^{-T} - \mathbf{F}^{-T}\,\nabla\Delta\mathbf{u}^T\,\mathbf{F}^{-T}\right)\mathbf{n}\, da$
  - pseudo load linear form $\delta\ell_3(\mathbf{w}, d; \delta d) = -\int_{A^n} \mathbf{w} \cdot \delta p \det\mathbf{F}\,\mathbf{F}^{-T}\,\mathbf{n}\, da$

QoI:

- QoI: $\theta(\mathbf{u}, d) = \int_\Omega \pi(\mathbf{u}, \nabla\mathbf{u})\, dv + \int_{A^n} \beta(\mathbf{u})\, da + \int_{A^d} \beta(\mathbf{p})\, da$
- adjoint load linear form: $\delta\theta(\mathbf{u}, d : \delta\mathbf{u}) = \int_\Omega \left(\frac{\partial\pi}{\partial\mathbf{u}} \cdot \delta\mathbf{u} + \frac{\partial\pi}{\partial\nabla\mathbf{u}} \cdot \nabla\delta\mathbf{u}\right) dv + \int_{A^n} \frac{\partial\beta}{\partial\mathbf{u}} \cdot \delta\mathbf{u}\, da + \int_{A^d} \frac{\partial\beta}{\partial\mathbf{p}} \cdot \delta\mathbf{p}\, da$

Maximum stress QoI: In this QoI we capture the "maximum" von Mises stress in the domain Ω. The word maximum is in quotes, because this is not a differentiable quantity and hence we approximate the maximum by the p-norm. To begin we note that for linear elasticity the stress is given by

$$\boldsymbol{\sigma} = \mathbb{C}[\nabla \mathbf{u}]$$

As a first step towards evaluating the Von Mises stress we compute the deviatoric (i.e. pressureless) part of the stress

$$\mathbf{s} = \boldsymbol{\sigma} - \frac{1}{3}\operatorname{tr}\boldsymbol{\sigma}\,\mathbf{I} = \left(\mathbb{I} - \frac{1}{3}\mathbf{I}\otimes\mathbf{I}\right)[\boldsymbol{\sigma}] = \mathbb{I}_D[\boldsymbol{\sigma}]$$

where $\mathbb{I}_D = \left(\mathbb{I} - \frac{1}{3}\mathbf{I}\otimes\mathbf{I}\right)$ is the deviatoric projector which is symmetric and has the property that $\mathbb{I}_D\,\mathbb{I}_D = \mathbb{I}_D$ and it is easily shown that $\operatorname{tr}\mathbf{s} = 0$. Having the deviatoric stress we now define the von Mises stress

$$\sigma_{vm} = \sqrt{\frac{3}{2}}\,(\mathbf{s}\cdot\mathbf{s})^{\frac{1}{2}} = \sqrt{\frac{3}{2}}\,(s_{ij}\,s_{ij})^{\frac{1}{2}}$$

For the sensitivity analysis we obtain the variation of σ_{vm} with respect to the stress $\boldsymbol{\sigma}$, i.e.

$$\sigma_{vm}(\boldsymbol{\sigma};\delta\boldsymbol{\sigma}) = \sqrt{\frac{3}{2}}\,(\mathbf{s}\cdot\mathbf{s})^{-\frac{1}{2}}\,\mathbf{s}\cdot\mathbb{I}_D[\delta\boldsymbol{\sigma}] = \sqrt{\frac{3}{2}}\,\mathbf{n}\cdot\delta\boldsymbol{\sigma}$$

where we define $\mathbf{n} = \mathbf{s}/\,(\mathbf{s}\cdot\mathbf{s})^{\frac{1}{2}}$ and utilize the projection property so that $\mathbb{I}_D[\mathbf{s}] = \mathbf{s}$.

We take two steps to evaluate the maximum stress QoI. In the first step we compute the the p QoI θ_p, and in the second step we compute the maximum stress QoI θ_{max}.

- p QoI:

$$\theta_p(\mathbf{u}, d) = \int_\Omega \sigma_{vm}^p\,dv$$

- adjoint load linear form: $\delta\boldsymbol{\sigma} \to \mathbb{C}[\nabla\delta\mathbf{u}]$

$$\begin{aligned}\delta\theta_p(\mathbf{u}, d:\delta\mathbf{u}) &= \int_\Omega p\,\sigma_{vm}^{p-1}\sqrt{\frac{3}{2}}\,\mathbf{n}\cdot\mathbb{C}[\nabla\delta\mathbf{u}]\,dv \\ &= \int_\Omega p\,\sigma_{vm}^{p-1}\sqrt{\frac{3}{2}}\,\mathbb{C}[\mathbf{n}]\cdot\nabla\delta\mathbf{u}\,dv\end{aligned}$$

- explicit derivative wrt. $\delta\mathbb{C}$: $\delta\boldsymbol{\sigma} \to \delta\mathbb{C}[\nabla\mathbf{u}]$

$$\begin{aligned}\delta\theta_p(\mathbf{u}, d:\delta d) &= \int_\Omega p\,\sigma_{vm}^{p-1}\sqrt{\frac{3}{2}}\,\mathbf{n}\cdot\delta\mathbb{C}[\nabla\mathbf{u}]\,dv \\ &= \int_\Omega p\,\sigma_{vm}^{p-1}\sqrt{\frac{3}{2}}\,(\mathbf{n}\otimes\nabla\mathbf{u})\cdot\delta\mathbb{C}\,dv\end{aligned}$$

- explicit derivative wrt. shape with *design* velocity $\mathbf{v}$:

$$\delta\theta_p(\mathbf{u}, d : \delta d) \;\; = \;\; \int_\Omega \left(-p\, \sigma_{vm}^{p-1} \sqrt{\frac{3}{2}}\, \mathbf{n} \cdot \mathbb{C}[\nabla \mathbf{u}\, \nabla \mathbf{v}] + \sigma_{vm}^{p}\, \mathrm{div} \mathbf{v} \right) dv$$

  Upon completing the analysis we compute the p QoI $\theta_p(d)$ and upon completing the above described adjoint sensitivity analysis we compute its sensitivity $\delta\theta_p(d; \delta d)$. This completes the first step. We now start the second step wherein define the maximum stress QoI $\theta_{max}(d)$ and obtain its sensitivity $\theta_{max}(d; \delta d)$

  - maximum stress QoI:
$$\theta_{max}(d) = (\theta_p)^{\frac{1}{p}}$$
  - Sensitivity:
$$\delta\theta_{max}(d; \delta d) = \frac{1}{p}\, (\theta_p)^{\frac{1-p}{p}}\; \delta\theta_p(d; \delta d)$$

CVaR stress QoI: In this QoI we quantify the average of the top X percent of the von Mises stress in the domain Ω. Please refer to the Maximum stress QoI above to see the computation of the von Mises stress and its variation. This QoI is evaluated via 3 steps. In step 1 we obtain the von Mises stress σ_X level corresponding to the top $X\%$ by

- For each element Ω_e evaluate its
  1. volume $\mathrm{vol}(\Omega_e) = \int_{\Omega_e} dv$
  2. element integrated stress [8] $\mathrm{vol}(\sigma_{vm}^e) = \int_{\Omega_e} \sigma_{vm}\, dv$
- Evaluate the total volume $TOTAL\_VOL = \sum_e \mathrm{vol}(\Omega_e)$.
- Create 2 arrays of length n_{elem}, $SORT\_STRESS$ and $VOLUME$
- Sort the $\mathrm{vol}(\sigma_{vm}^e)$ in ascending order to create $SORT\_STRESS$.
- Place the $\mathrm{vol}(\Omega_e)$ in the associated index of the $VOLUME$ array, e.g. if $STRESS(52) = \mathrm{vol}(\sigma_{vm}^4)$ then $VOLUME(52) = \mathrm{vol}(\Omega_4)$.
- Find the entry IX corresponding to $1-X\%$ of the volume, i.e. $\sum_I VOLUME(I)$ until $\sum_I VOLUME(I) = (1 - X\%)\, TOTAL\_VOL$ and then equate $IX = I$.
- Evaluate the $X\%$ stress, i.e. $\sigma_X = SORT\_STRESS(IX)/VOLUME(IX)$.

[8] In many cases this may be evaluated via a one point quadrature in which case $\mathrm{vol}(\sigma_{vm}^e) = \mathrm{vol}(\Omega_e)\, \sigma_{vm}^e(\bar{\mathbf{x}})$ where $\bar{\mathbf{x}}$ is the element cetroid.

In step 2 we evaluate 2 QoIs, the percentile integrated volume $\theta_{per-vol}$ and the percentile integrated stress θ_{per-vm}. To do this we integrate over the entire domain and use a heaviside, i.e. unit step, function H to omit the regions we do not want to integrate over. And to obtain a differentiable QoI we replace the heaviside with a smooth approximation H_ϵ defined such that $H_\epsilon(x) = 1/2 + 1/2\,\tanh(x/\epsilon)$ so that $\lim_{\epsilon\to 0} H_\epsilon = H$.

- QoIs
  - percentile integrated volume: $\theta_{per-vol}(\mathbf{u}, d) = \int_\Omega H_\epsilon(\sigma_{vm} - \sigma_X)\, dv$
  - percentile integrated stress: $\theta_{per-vm}(\mathbf{u}, d) = \int_\Omega H_\epsilon(\sigma_{vm} - \sigma_X)\, \sigma_{vm}\, dv$
- adjoint load linear forms: $\delta\boldsymbol{\sigma} \to \mathbb{C}[\nabla\delta\mathbf{u}]$
  - percentile integrated volume:

$$\delta\theta_{per-vol}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega H'_\epsilon(\sigma_{vm} - \sigma_X)\sqrt{\frac{3}{2}}\,\mathbb{C}[\mathbf{n}]\cdot\nabla\delta\mathbf{u}\, dv$$

  - percentile integrated stress:

$$\delta\theta_{per-vm}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega \big(H'_\epsilon(\sigma_{vm} - \sigma_X)\,\sigma_{vm} + H_\epsilon(\sigma_{vm} - \sigma_X)\big)\,\sqrt{\frac{3}{2}}\,\mathbb{C}[\mathbf{n}]\cdot\nabla\delta\mathbf{u}\, dv$$

- explicit derivatives wrt. $\delta\mathbb{C}$: $\delta\boldsymbol{\sigma} \to \delta\mathbb{C}[\nabla\mathbf{u}]$
  - percentile integrated volume:

$$\delta\theta_{per-vol}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega H'_\epsilon(\sigma_{vm} - \sigma_X)\sqrt{\frac{3}{2}}\,\mathbf{n}\otimes\nabla\mathbf{u}\cdot\delta\mathbb{C}\, dv$$

  - percentile integrated stress:

$$\delta\theta_{per-vm}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega \big(H'_\epsilon(\sigma_{vm} - \sigma_X)\,\sigma_{vm} + H_\epsilon(\sigma_{vm} - \sigma_X)\big)\,\sqrt{\frac{3}{2}}\,\mathbf{n}\otimes\nabla\mathbf{u}\cdot\delta\mathbb{C}\, dv$$

- explicit derivatives wrt. shape with *design* velocity $\mathbf{v}$:
  - percentile integrated volume:

$$\delta\theta_{per-vol}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega \left[-H'_\epsilon(\sigma_{vm} - \sigma_X)\,\sqrt{\frac{3}{2}}\,\mathbf{n}\cdot\mathbb{C}[\nabla\mathbf{u}\,\nabla\mathbf{v}] + H_\epsilon(\sigma_{vm} - \sigma_X)\,\mathrm{div}\mathbf{v}\right] dv$$

  - percentile integrated stress:

$$\delta\theta_{per-vm}(\mathbf{u}, d : \delta\mathbf{u}) \;=\; \int_\Omega \Bigg[-\big(H'_\epsilon(\sigma_{vm} - \sigma_X)\,\sigma_{vm} + H_\epsilon(\sigma_{vm} - \sigma_X)\big)\,\sqrt{\frac{3}{2}}\,\mathbf{n}\cdot\mathbb{C}[\nabla\mathbf{u}\,\nabla\mathbf{v}] + H_\epsilon(\sigma_{vm} - \sigma_X)\;\sigma_{vm}\,\mathrm{div}\mathbf{v}\Bigg]\, dv$$

Upon completing the analysis we compute $\theta_{per-vol}(d)$ and $\theta_{per-vm}(d)$ and upon completing the above described adjoint sensitivity analysis we compute their sensitivities $\delta\theta_{per-vol}(d;\delta d)$ and $\delta\theta_{per-vm}(d;\delta d)$. This completes the second step. We now start the third step wherein we define the Von Mises stress CVaR QoI $\theta_{CVaR}(d;\delta d)$ and obtain its sensitivity $\theta_{CVaR}(d;\delta d)$

- Von Mises stress CVaR QoI :
$$\theta_{CVaR}(d) = \frac{\theta_{per-vm}(d)}{\theta_{per-vol}(d)}$$
- Sensitivity:
$$\delta\theta_{CVaR}(d;\delta d) = \frac{1}{\theta_{per-vol}(d)}\,\delta\theta_{per-vm}(d;\delta d) - \frac{\theta_{per-vm}(d)}{(\theta_{per-vol}(d))^2}\,\delta\theta_{per-vol}(d;\delta d)$$

9.2 Constitutive functions:

- SIMP:
  - Notation: Volume fraction ν, nominal stiffness tensor $\mathbb{C}_o$.
  - Stress: $\mathbf{P} = \nu^p\,\mathbf{C}_o[\nabla\mathbf{u}]$
  - Tangent stiffness: $\partial\mathbf{P}/\partial\nabla\mathbf{u} = \nu^p\,\mathbf{C}_o$
  - Stress variation: $\delta\mathbf{P} = p\,\nu^{p-1}\,\delta\nu\,\mathbf{C}_o[\nabla\mathbf{u}]$
- Fiber reinforced composite:
  - Notation: Fiber orientation θ, nominal stiffness tensor $\mathbb{C}_o$.
  - Stress: $\mathbf{P} = \mathbf{R}(\theta)\,\mathbf{C}_o\,\mathbf{R}^T(\theta)[\nabla\mathbf{u}]$
  - Tangent stiffness: $\partial\mathbf{P}/\partial\nabla\mathbf{u} = \mathbf{R}(\theta)\,\mathbf{C}_o\,\mathbf{R}^T(\theta)$
  - Stress variation: $\delta\mathbf{P} = \delta\theta\,\left(\mathbf{R}'(\theta)\,\mathbf{C}_o\,\mathbf{R}^T(\theta) + \mathbf{R}(\theta)\,\mathbf{C}_o\,(\mathbf{R}')^T(\theta)[\nabla\mathbf{u}]\right)$
- Meta material:
  - Notation: Material fields d_1, $d_2, \cdots$, stiffness tensor $\mathbb{C}$.
  - Stress: $\mathbf{P} = \mathbb{C}(d_1, d_2, \cdots)[\nabla\mathbf{u}]$
  - Tangent stiffness: $\partial\mathbf{P}/\partial\nabla\mathbf{u} = \mathbb{C}(d_1, d_2, \cdots)$
  - Stress variation: $\delta\mathbf{P} = \delta d_1\,\frac{\partial\mathbb{C}}{\partial d_1}[\nabla\mathbf{u}] + \delta d_2\,\frac{\partial\mathbb{C}}{\partial d_2}[\nabla\mathbf{u}] + \cdots$
- Nonlinear elasticity:
  - Notation: Material fields d_1, $d_2, \cdots$, deformation gradient $\mathbf{F} = \mathbf{I} + \nabla\mathbf{u}$, internal energy function $\psi(\mathbf{F}, d_1, d_2, \cdots)$.
  - Stress: $\mathbf{P} = \partial\psi/\partial\mathbf{F}$
  - Tangent stiffness: $\partial\mathbf{P}/\partial\nabla\mathbf{u} = \partial^2\psi/\partial\mathbf{F}^2$
  - Stress variation: $\delta\mathbf{P} = \delta d_1\,\frac{\partial^2\psi}{\partial\mathbf{F}\partial d_1} + \delta d_2\,\frac{\partial^2\psi}{\partial\mathbf{F}\partial d_2} + \cdots$

9.3 Steady-state heat conduction

Symbol definition:

- T: temperature
- ω: virtual temperature, adjoint temperature
- r: source
- q^p: prescribed flux
- $\mathbf{q}$: heat flux
- $q^n = \mathbf{q} \cdot \mathbf{n}$: surface flux
- T_o: ambient temperature
- Ω: undeformed configuration
- A^n: undeformed configuration surface with prescribed Neumann BCs
- A^d: undeformed configuration surface with prescribed Dirichlet BCs
- $\mathbf{n}$ normal vector to undeformed configuration surface

Strong form

$$\begin{array}{rcll} \mathrm{div}\mathbf{q} + r & = & 0 & \text{in } \Omega \\ \mathbf{q} & = & \hat{\mathbf{q}}(T, \nabla T) = -\mathbf{k}(T)\, \nabla T & \text{in}\Omega \\ \mathbf{q} \cdot \mathbf{n} & = & q^p & \text{on } A^n \\ T & = & T^p & \text{on } A^d \end{array}$$

Bilinear, load linear form and pseudo load definitions:

- residual: $r(u, d) = \ell_1(\nabla\omega, d) + \ell_2(\omega, d) + \ell_3(\omega, d) = \int_\Omega \nabla\omega \cdot \mathbf{q}\, dv - \int_\Omega \omega\, r\, dv - \int_{A^n} \omega\, q^p\, da$
- tangent: $a_{11}(\nabla\Delta T, \nabla\omega) + a_{12}(\Delta T, \nabla\omega) = \int_\Omega \nabla\omega \cdot \frac{\partial \mathbf{q}}{\partial \nabla\theta}[\nabla\Delta T]\, dv + \int_\Omega \nabla\omega \cdot \frac{\partial \mathbf{q}}{\partial \theta}[\Delta T]\, dv$
- pseudo load: $\ell_1(\nabla\omega, d; \delta d) + \ell_2(\omega, d; \delta d) + \ell_3(\omega, d; \delta d) = \int_\Omega \nabla\omega \cdot \delta\mathbf{q}(d; \delta d)\, dv - \int_\Omega \omega\, \delta r(d; \delta d)\, dv - \int_{A^n} \omega\, \delta q^p(d; \delta d)\, da$
- Robin BC load:
  - load linear form: $\ell_3(\omega, d) = -\int_{A^n} \omega\, q^p\, dv = -\int_{A^n} \omega\, h\, (T_o - T)\, da$
  - tangent: $a_3(\Delta T, \omega) = -\int_{A^n} \omega\, \left(\frac{\partial h}{\partial T}\, (T_o - T) - h\right)\, \Delta T\, da$

- pseudo load: [9] $\delta\ell_3(\omega, d; \delta d) = -\int_{A^n} \omega \, (\delta h(d;\delta d)\,(T_o(d) - T) + h(d)\,\delta T_o(d,\delta d))\; da$

QoI:

- QoI: $\theta(T, d) = \int_\Omega \pi(T, \nabla T)\, dv + \int_{A^n} \beta(T)\, da + \int_{A^d} \beta(q^n)\, da$
- adjoint load: $\delta\theta(T, d : \delta T) = \int_\Omega \left(\frac{\partial \pi}{\partial T}\, \delta T + \frac{\partial \pi}{\partial \nabla T} \cdot \nabla \delta T\right)\, dv + \int_{A^n} \frac{\partial \beta}{\partial T}\, \delta T\, da + \int_{A^d} \frac{\partial \beta}{\partial q^n}\, \delta q^n\, da$

9.4 Steady-state thermo-mechanics

Symbol definition: same as above, but we place superscripts u and T to denote the respective mechanical and thermal linear and bilinear forms. The only coupling is due to stress, specifically $\mathbf{P}$ is a function of temperature T.

Solve heat conduction first (as above)
Mechanical response: same as above except

- tangent: $a^u_{11}(\nabla \Delta \mathbf{u}, \nabla \mathbf{w}) + a^u_{12}(\Delta T, \nabla \mathbf{w}) = \int_\Omega \nabla \mathbf{w} \cdot \frac{\partial \mathbf{P}}{\partial \mathbf{F}}[\nabla \Delta \mathbf{u}]\, dv + \int_\Omega \nabla \mathbf{w} \cdot \frac{\partial \mathbf{P}}{\partial T} \Delta T\, dv$

Assume the QOIs are the same as in the mechanical case, with the caveat that now $\mathbf{P}$ is dependent on T.

- QoI: $\theta(\mathbf{u}, T, d) = \int_\Omega \pi(\mathbf{u}, \nabla \mathbf{u}, T)\, dv$
- adjoint loads:
  - $\delta\theta(\mathbf{u}, T, d; \delta \mathbf{u}) = \int_\Omega \left(\frac{\partial \pi}{\partial \mathbf{u}} \cdot \delta \mathbf{u} + \frac{\partial \pi}{\partial \nabla \mathbf{u}} \cdot \nabla \delta \mathbf{u}\right)\, dv$
  - $\delta\theta(\mathbf{u}, T, d; \delta T) = \int_\Omega \frac{\partial \pi}{\partial T}\, \delta T\, dv$

As seen in Figure 13, in the primal analysis we first solve the thermal problem for θ and then the mechanical problem for u. However in the adjoint analysis we first solve the mechanical problem for $\mathbf{w}$ and then the thermal problem for ω.

9.5 Brinkman flow $\sim$ steady incompressible Navier Stokes with a velocity dependent source

Symbol definition:

- $\mathbf{v}$: velocity
- $\mathbf{w}$: virtual velocity, adjoint velocity
- $\mathbf{b}$: Body load
- p^p: prescribed pressure

[9] This is an exception in that we cannot merely replace h and θ_o in the ℓ_3 evaluation with their variations δh and $\delta\theta_o$ in the $\delta\ell_3$ evaluation. However, this is the case if we only vary the film coefficient, i.e. if $\delta\theta_o = 0$.

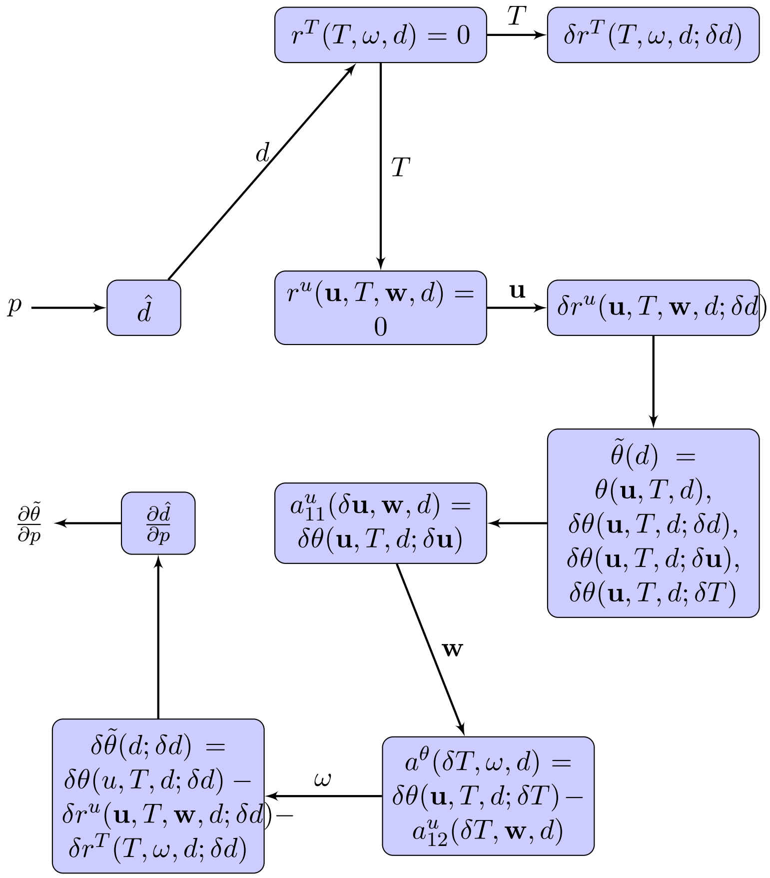


Figure 13: Analysis sensitivity analysis for thermo-mechanical response.

- p: pressure
- ω: virtual pressure, adjoint pressure
- $\boldsymbol{\sigma} = 2\,\mu\mathbf{D} - p\,\mathbf{I}$: Cauchy stress
- μ: Shear viscosity
- α: inverse permeability
- $\mathbf{D} = 1/2(\nabla\mathbf{v} + \nabla^T\mathbf{v})$: stretching tensor
- $\mathbf{t} = \boldsymbol{\sigma}\,\mathbf{n}$: traction
- Ω: deformed configuration

- A^n: deformed configuration surface with prescribed Neumann BCs
- A^d: deformed configuration surface with prescribed Dirichlet BCs
- $\mathbf{n}$ normal vector to deformed configuration surface

Bilinear, load linear form and pseudo load definitions:

- residual equation, primal problem: Find kinematically admissible $(\mathbf{v}, p)$ such that

$$\begin{aligned}
r(\mathbf{v}, \mathbf{w}, d) = 0 &= \ell_1(\nabla \mathbf{w}, d) + \ell_2(\mathbf{w}, d) + \ell_3(\mathbf{w}, d) + \ell_4(\mathbf{w}, d) + \ell_5(\omega, d) \\
&= \int_\Omega \nabla \mathbf{w} \cdot \boldsymbol{\sigma} \, dv + \int_\Omega \mathbf{w} \cdot (\rho \, \nabla \mathbf{v} \, \mathbf{v} + \alpha \, \mathbf{v}) \, dv \\
&\qquad - \int_\Omega \mathbf{w} \cdot \mathbf{b} \, dv - \int_{A^n} \mathbf{w} \cdot p^p \, \mathbf{n} \, dv + \int_\Omega \omega \, \mathrm{div} \mathbf{v} \, dv \\
&= \int_\Omega \nabla \mathbf{w} \cdot (\mu(\nabla^T \mathbf{v} + \nabla \mathbf{v}) - p \, \mathbf{I}) \, dv + \int_\Omega \mathbf{w} \cdot (\rho \, \nabla \mathbf{v} \, \mathbf{v} + \alpha \, \mathbf{v}) \, dv \\
&\qquad - \int_\Omega \mathbf{w} \cdot \mathbf{b} \, dv - \int_{A^n} \mathbf{w} \cdot p^p \, \mathbf{n} \, dv + \int_\Omega \omega \, \mathrm{div} \mathbf{v} \, dv
\end{aligned}$$

  for all kinematically admissible $(\mathbf{w}, q)$

- tangent:

$$\begin{aligned}
&a_1^{(v)}(\nabla \Delta \mathbf{v}, \nabla \mathbf{w}) + a_1^{(p)}(\Delta p, \nabla \mathbf{w}) + a_2(\mathbf{v}, \mathbf{w}) + a_5(\Delta \mathbf{v}, \omega) \\
&\quad = \int_\Omega \nabla \mathbf{w} \cdot \frac{\partial \boldsymbol{\sigma}}{\partial \nabla \mathbf{v}} [\nabla \Delta \mathbf{v}] \, dv + \int_\Omega \nabla \mathbf{w} \cdot \frac{\partial \boldsymbol{\sigma}}{\partial p} [\Delta p] \, dv + \\
&\qquad \int_\Omega \mathbf{w} \cdot (\rho \, \nabla \Delta \mathbf{v} \, \mathbf{v} + \rho \, \nabla \mathbf{v} \, \Delta \mathbf{v} + \mathbf{w} \cdot \alpha \, \Delta \mathbf{v}) \, dv + \int_\Omega \omega \, \mathrm{div} \Delta \mathbf{v} \, dv \\
&\quad = \int_\Omega \nabla \mathbf{w} \cdot \mu(\nabla^T \Delta \mathbf{v} + \nabla \Delta \mathbf{v}) \, dv + \int_\Omega \nabla \mathbf{w} \cdot \mathbf{I} \Delta p \, dv + \\
&\qquad \int_\Omega \mathbf{w} \cdot (\rho \, \nabla \Delta \mathbf{v} \, \mathbf{v} + \rho \, \nabla \mathbf{v} \, \Delta \mathbf{v} + \mathbf{w} \cdot \alpha \, \Delta \mathbf{v}) \, dv + \int_\Omega \omega \, \mathrm{div} \Delta \mathbf{v} \, dv
\end{aligned}$$

- pseudo load linear form:

$$\begin{aligned}
\delta r(\mathbf{v}, p, \mathbf{w}, \omega, d; \delta d) &= \delta \ell_1(\nabla \mathbf{w}, d; \delta d) + \delta \ell_2(\mathbf{w}, d; \delta d) + \delta \ell_3(\mathbf{w}, d; \delta d) + \delta \ell_4(\mathbf{w}, d; \delta d) \\
&= \int_\Omega \nabla \mathbf{w} \cdot \frac{\partial \boldsymbol{\sigma}}{\partial \mu} \, \delta \mu(d; \delta d) \, dv + \int_\Omega \mathbf{w} \cdot (\delta \rho(d; \delta d) \, \nabla \mathbf{v} \, \mathbf{v} + \delta \alpha(d; \delta d) \, \mathbf{v}) \, dv \\
&\qquad - \int_\Omega \mathbf{w} \cdot \, \delta \mathbf{b}(d; \delta d) \, dv - \int_{A^n} \mathbf{w} \cdot \delta p^p(d; \delta d) \, \mathbf{n} \, dv
\end{aligned}$$

- pseudo load linear form with only the Brinkman parameter:

$$\begin{aligned}\delta r(\mathbf{v}, p, \mathbf{w}, \omega, d; \delta d) &= \delta \ell_3(\mathbf{w}, d; \delta d) \\ &= \int_\Omega \mathbf{w} \cdot \delta\alpha(d; \delta d)\, \mathbf{v}\, dv\end{aligned}$$

QoI:

  - QoI: $\theta(\mathbf{v}, p, d) = \int_\Omega \pi(\mathbf{v}, \nabla\mathbf{v}, p)\, dv + \int_{A^n} \beta(\mathbf{v})\, da + \int_{A^d} \beta(\mathbf{t})\, da$
  - adjoint load linear form: $\delta\theta((\mathbf{v}, p), d; (\delta\mathbf{v}, \delta v)) = \int_\Omega \left(\frac{\partial\pi}{\partial\mathbf{v}} \cdot \delta\mathbf{v} + \frac{\partial\pi}{\partial\nabla\mathbf{v}} \cdot \nabla\delta\mathbf{v} + \frac{\partial\pi}{\partial p} \cdot \delta p \right) dv + \int_{A^n} \frac{\partial\beta}{\partial\mathbf{v}} \cdot \delta\mathbf{v}\, da + \int_{A^d} \frac{\partial\beta}{\partial\mathbf{t}} \cdot \delta\mathbf{t}\, da$

Specialized QoI:

  - QoI: $\theta(\mathbf{v}, p, d) = \int_{A^d} \beta(\mathbf{t})\, da$
  - adjoint load linear form: $\delta\theta((\mathbf{v}, p), d; (\delta\mathbf{v}, \delta v)) = \int_{A^d} \frac{\partial\beta}{\partial\mathbf{t}} \cdot \delta\mathbf{t}\, da$

9.6 Steady-state linear solid mechanics (frequency response)

Symbol definition:

- Ω: undeformed configuration
- $\mathbf{n}$ normal vector to undeformed configuration surface
- $\mathbf{u}$: displacement
- $\mathbb{C}$: Elasticity tensor
- ρ density
- $\mathbf{t}^{\mathbf{n}} = \mathbb{C}[\nabla \mathbf{u}]\,\mathbf{n}$: traction
- $\mathbf{w}$: virtual displacement, adjoint displacement
- $\mathbf{b}$: Body load
- $\mathbf{t}^p$: prescribed traction
- A^n: undeformed configuration surface with prescribed Neumann BCs
- A^d: undeformed configuration surface with prescribed Dirichlet BCs
- ω: forcing frequency
- $(\omega_i, \boldsymbol{\phi}_i)$ eigen pairs

Bilinear form, load linear form and residual definitions:

- stiffness: $a_1(\nabla \mathbf{u}, \nabla \mathbf{w}, d) - a_2(\mathbf{u}, \mathbf{w}, d) = \int_\Omega \nabla \mathbf{w} \cdot \mathbb{C}[\nabla \mathbf{u}]\,dv - \omega^2 \int_\Omega \mathbf{w} \cdot \rho\,\mathbf{u}\,dv = \mathbf{W}^T \left(\mathbf{K} - \omega^2\,\mathbf{M}\right)\,\mathbf{U}$
- load: $\ell_1(\mathbf{w}, d) + \ell_2(\mathbf{w}, d) = \int_\Omega \mathbf{w} \cdot \mathbf{b}\,dv + \int_{A^n} \mathbf{w} \cdot \mathbf{t}^p\,dv = \mathbf{W}^T\,\mathbf{P}$
- residual: $r(\mathbf{u}, \mathbf{w}, d) = a_1(\nabla \mathbf{u}, \nabla \mathbf{w}, d) - a_2(\mathbf{u}, \mathbf{w}, d) - \ell_1(\mathbf{w}, d) - \ell_2(\mathbf{w}, d) = \mathbf{W}^T \left(\left(\mathbf{K} - \omega^2\,\mathbf{M}\right)\,\mathbf{U} - \mathbf{P}\right)$

QoI:

- QoI: $\theta(\mathbf{u}, d) = \int_\Omega \pi(\mathbf{u}, \nabla \mathbf{u}, d)\,dv = \Theta(\mathbf{U}, d)$
- adjoint load linear form: $\delta\theta(\mathbf{u}, d; \delta\mathbf{u}) = \int_\Omega \left(\frac{\partial \pi}{\partial \mathbf{u}} \cdot \delta\mathbf{u} + \frac{\partial \pi}{\partial \nabla \mathbf{u}} \cdot \nabla \delta\mathbf{u}\right)\,dv = \delta\mathbf{U}^T\,\frac{\partial \Theta}{\partial \mathbf{U}}$
- Explicit derivative: $\delta\theta(\mathbf{u}, d; \delta d) = \int_\Omega \frac{\partial \pi}{\partial d}\,\delta d\,dv = \delta d\,\frac{\partial \Theta}{\partial d}$

Adjoint sensitivity analysis:

- Find $\mathbf{w}$ such that $a_1(\nabla\delta\mathbf{u}, \nabla\mathbf{w}, d) - a_2(\delta\mathbf{u}, \mathbf{w}, d) = \delta\theta(\mathbf{u}, d; \delta\mathbf{u})$ for all $\delta\mathbf{u}$. The FEM problem is, find $\mathbf{W}$ such that $\left(\mathbf{K} - \omega^2\,\mathbf{M}\right)^T\,\mathbf{W} = \frac{\partial\Theta}{\partial\mathbf{U}}$
- pseudo load linear form: $\delta r(\mathbf{u}, \mathbf{w}, d; \delta d) = \delta a_1(\nabla\mathbf{u}, \nabla\mathbf{w}, d; \delta d) - \delta a_2(\mathbf{u}, \mathbf{w}, d; \delta d) - \delta\ell_1(\mathbf{w}, d; \delta d) - \delta\ell_2(\mathbf{w}, d; \delta d) = \int_\Omega \nabla\mathbf{w}\cdot\delta\mathbb{C}(d; \delta d)[\nabla\mathbf{u}]\,dv - \omega^2\,\int_\Omega \mathbf{w}\cdot\delta\rho(d; \delta d)\,\mathbf{u}\,dv - \int_\Omega \mathbf{w}\cdot\delta\mathbf{b}(d; \delta d)\,dv - \int_{A^n} \mathbf{w}\cdot\delta\mathbf{t}^p(d; \delta d)\,dv = \mathbf{W}^T\left(\left(\delta\mathbf{K} - \omega^2\,\delta\mathbf{M}\right)\,\mathbf{U}\right.$
- Sensitivity computation: $\frac{D\theta}{Dd}\,\delta d = \delta\theta(\mathbf{u}, d; \delta d) - \delta r(\mathbf{u}, \mathbf{w}, d; \delta d) = \frac{\partial\Theta}{\partial d}\,\delta d - \mathbf{W}^T\left(\left(\delta\mathbf{K} - \omega^2\,\delta\mathbf{M}\right)\,\mathbf{U} - \delta\mathbf{P}\right)$.

Modal analysis

1. Solve eigenvalue problem

$$(\mathbf{K} - \omega_i^2\,\mathbf{M})\,\mathbf{\Phi}_i = 0 \tag{112}$$

   for the eigenpairs $(\omega_i, \mathbf{\Phi}_i)$ arranged in ascending order so that $\omega_1 \leq \omega_2 \leq \Omega_3 \cdots$ and normalized such that $\mathbf{\Phi}_i^T\mathbf{M}\,\mathbf{\Phi}_i = 1$. Here $i = 1, 2, \cdots n_{modes}$ where $n_{modes} \leq n_{dof}$.

2. Form the "modal matrix" $\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{\Phi}_2 & \cdots & \mathbf{\Phi}_{n_{modes}} \end{bmatrix}$.

3. Form the modal stiffness matrix, modal mass matrix and modal force vector

$$\begin{aligned} \mathbf{K}_\phi &= \mathbf{\Phi}^T\,\mathbf{K}\,\mathbf{\Phi} = \mathrm{diag}\left(\omega_1^2, \omega_2^2, \cdots, \omega_{n_{mode}}^2\right) \\ \mathbf{M}_\phi &= \mathbf{\Phi}^T\,\mathbf{M}\,\mathbf{\Phi} = \mathbf{I} \\ \mathbf{P}_\phi &= \mathbf{\Phi}^T\,\mathbf{P} \end{aligned} \tag{113}$$

4. Solve modal problem

$$(\mathbf{K}_\phi - \omega^2\,\mathbf{M}_\phi)\,\mathbf{U}_\phi = \mathbf{P}_\phi \tag{114}$$

5. Evaluate $\mathbf{U} = \mathbf{\Phi}\,\mathbf{U}_\phi$

6. Evaluate Θ, $\frac{\partial\Theta}{\partial\mathbf{U}}$, $\frac{\partial\Theta}{\partial d}$ as done above.

7. Solve adjoint problem

$$(\mathbf{K}_\phi - \omega^2\,\mathbf{M}_\phi)^T\,\mathbf{W}_\phi = \mathbf{\Phi}^T\,\frac{\partial\Theta}{\partial\mathbf{U}} \tag{115}$$

8. Evaluate the adjoint response $\mathbf{W} = \mathbf{\Phi}\,\mathbf{W}_\phi$

9. Compute the sensitivity as done above

$$\frac{D\theta}{Dd}\,\delta d = \mathbf{W}^T\left(\left(\delta\mathbf{K} - \omega^2\,\delta\mathbf{M}\right)\,\mathbf{U} - \delta\mathbf{P}\right) \tag{116}$$

10 Eigensystem sensitivity analysis

After discretization we have the generalized eigenvalue problem

$$\mathbf{K}\boldsymbol{\phi}_i = \lambda_i \, \mathbf{M} \, \boldsymbol{\phi}_i \tag{117}$$

which we solve for the eigenparis $(\lambda_i, \boldsymbol{\phi}_i)$, $i = 1, 2, \cdots, n$. It is common to mass normalize the eigenvectors by introducing the scale factor

$$\alpha_i = \left(\boldsymbol{\phi}_i^T \, \mathbf{M} \, \boldsymbol{\phi}_i\right)^{\frac{1}{2}} \tag{118}$$

and defining the mass normalized eigenvectors

$$\bar{\boldsymbol{\phi}}_i = \frac{1}{\alpha_i} \, \boldsymbol{\phi}_i \tag{119}$$

that satisfy

$$\bar{\boldsymbol{\phi}}_i^T \, \mathbf{M} \, \bar{\boldsymbol{\phi}}_i = 1 \tag{120}$$

Finally we can evaluate any generalized function of the eigenpair

$$\Theta = \Pi(\lambda_i, \bar{\boldsymbol{\phi}}_i) \tag{121}$$

Now we need the sensitivity of Θ, i.e.

$$\delta\Theta = \frac{\partial \Pi}{\partial \lambda_i} \, \delta\lambda_i + \frac{\partial \Pi}{\partial \boldsymbol{\phi}_i} \, \delta\boldsymbol{\phi}_i$$

where we have dropped the arguments in Π for conciseness. As a first step we assume $(\lambda_i, \boldsymbol{\phi}_i)$ is a simple eigenpair. It is then easily verified that

$$\delta\lambda_i = \frac{\boldsymbol{\phi}_i^T \, (\delta\mathbf{K} - \lambda_i \, \delta\mathbf{M}) \, \boldsymbol{\phi}_i}{\boldsymbol{\phi}_i^T \, \mathbf{M} \, \boldsymbol{\phi}_i} \tag{122}$$

We use the adjoint method to annihilate the $\delta\bar{\boldsymbol{\phi}}_i$ from the sensitivity $\delta\Theta$. To this end we note that

$$\Theta = \Pi(\lambda_i, \bar{\boldsymbol{\phi}}_i) + \bar{\mathbf{w}}_i^T \left(\bar{\boldsymbol{\phi}}_i - \frac{1}{\alpha_i} \, \boldsymbol{\phi}_i\right) + \mathbf{w}_i^T \left(\mathbf{K}\boldsymbol{\phi}_i - \lambda_i \, \mathbf{M} \, \boldsymbol{\phi}_i\right) \tag{123}$$

since the coefficients of the adjoint vectors $\mathbf{w}$ and $\bar{\mathbf{w}}$ equal zero. Taking the variation of the above and rearranging gives

$$\begin{aligned} \delta\Theta \;=\;& \delta\bar{\boldsymbol{\phi}}^T \left(\frac{\partial \Pi}{\partial \bar{\boldsymbol{\phi}}}^T + \bar{\mathbf{w}}\right) + \\ & \delta\boldsymbol{\phi}^T \left(\frac{1}{\alpha^3}(\mathbf{M}\,\boldsymbol{\phi})\boldsymbol{\phi}^T - \frac{1}{\alpha}\,\mathbf{I}\right) \bar{\mathbf{w}} + \delta\boldsymbol{\phi}^T \left(\mathbf{K} - \lambda \, \mathbf{M}\right) \mathbf{w} + \\ & \frac{\partial \Pi}{\partial \lambda} \, \delta\lambda + \bar{\mathbf{w}}^T \left(\frac{1}{2\,\alpha^3} (\boldsymbol{\phi}^T \, \delta\mathbf{M} \, \boldsymbol{\phi}) \, \boldsymbol{\phi}\right) + \mathbf{w}^T \left(\delta\mathbf{K} - \delta\lambda \, \mathbf{M} - \lambda \, \delta\mathbf{M}\right) \, \boldsymbol{\phi} \end{aligned} \tag{124}$$

where we have dropped the eigenpair subscript i for brevity. To annihilate the implicit $\delta\bar{\boldsymbol{\phi}}$ variation we assign

$$\bar{\mathbf{w}} = -\frac{\partial \Pi}{\partial \bar{\boldsymbol{\phi}}}^T \tag{125}$$

Having $\bar{\mathbf{w}}$ we can now annihilate the implicit $\delta\boldsymbol{\phi}$ variation by solving the linear equation

$$(\mathbf{K} - \lambda\,\mathbf{M})\,\mathbf{w} = -\left(\frac{1}{\alpha^3}(\mathbf{M}\,\boldsymbol{\phi})\boldsymbol{\phi}^T - \frac{1}{\alpha}\,\mathbf{I}\right)\bar{\mathbf{w}} \tag{126}$$

for $\mathbf{w}$. Unfortunately the coefficient matrix $\mathbf{K} - \lambda\,\mathbf{M}$ is singular. But this is not a problem because ϕ is arbitrary to within a scalar multiple which we use to our advantage. This means, e.g. that we could *require* $\boldsymbol{\phi}(1) = 1$ (assuming $\boldsymbol{\phi}(1) \neq 0$) or any other constant, say β which comes out of the eigenvalue solve of (117). And because of this arbitrariness there is also arbitrariness in $\delta\boldsymbol{\phi}$ which we resolve by *requiring* that $\delta\boldsymbol{\phi}(1) = 0$ so that $\boldsymbol{\phi}(1)$ always equals β. This allows us to arbitrarily assign the value of $\mathbf{w}(1)$ and to make things easy we equate $\mathbf{w}(1) = 0$. Eliminating the top row from the linear system (126) results in an $n-1$ dimensional linear system that can be solved for the remaining elements of $\mathbf{w}$. Upon computing $\bar{\mathbf{w}}$ and $\mathbf{w}$ the sensitivity reduces to

$$\delta\Theta \;=\; \frac{\partial \Pi}{\partial \lambda}\,\delta\lambda + \bar{\mathbf{w}}^T\left(\frac{1}{2\,\alpha^3}(\boldsymbol{\phi}^T\,\delta\mathbf{M}\,\boldsymbol{\phi})\,\boldsymbol{\phi}\right) + \mathbf{w}^T\left(\delta\mathbf{K} - \delta\lambda\,\mathbf{M} - \lambda\,\delta\mathbf{M}\right)\boldsymbol{\phi} \tag{127}$$

11 Automatic Differentiation

Here we present the *Automatic Differentiation* (AD) in the *reverse* mode for computing the sensitivities. To begin we consider the analysis and sensitivity illustrated in Figure 1 after 1) assuming for simplicity that $d = p$ and 2) after *discretizing* so that $d \in \mathcal{R}^d$, $u \in \mathcal{R}^u$, $r_u : \mathcal{R}^u \times \mathcal{R}^d \to \mathcal{R}^u$, and $\theta : \mathcal{R}^u \times \mathcal{R}^d \to \mathcal{R}$. First we perform the *primal* analysis and compute the QoI. The steps for this are detailed in the left column of Table 1. To be consistent with the AD community, the middle column introduces new variables, i.e. $v_0 \leftarrow d$, $v_1 \leftarrow u$, $v_2 \leftarrow \tilde{\theta}$. Zero and negative subscripts denote input variables whereas positive subscripts denote computed variables. Other than this, the middle column is mostly identical to the left column with the exceptions that a) we eliminate the trial function w from the residual (which follows from the discretization) and b) we introduce the superfluous output scalar y. The right column is identical to the middle column except that we replace the implicit equation $r(v_1, v_0) = 0$ with an explicit equation $v_1 = \phi(v_0)$ where $\phi : \mathcal{R}^d \to \mathcal{R}^u$. This is justified if the implicit function theorem holds, which we assume is the case. Roughly, the implicit function theorem states that if v_1 solves the implicit equation $r_u(v_1, v_0) = 0$ for a fixed v_0 and $\partial r_u/\partial v_1$ is not singular, then in the neighborhood of (v_1, v_0) we can express $v_1 = \phi(v_0)$. Moreover, $\partial v_1/\partial v_0 = \partial\phi/\partial v_0 = -(\partial r_u/\partial v_1)^{-1}\partial r_u/\partial v_0$ which follows by differentiating $r_u(v_1, v_0) = 0$ wrt. v_0.

The goal of the sensitivity analysis is to compute $\partial\theta/\partial d$, or for our AD case $\partial y/\partial v_0$, i.e. the Jacobian

$$J = \frac{\partial y}{\partial v_0} \tag{128}$$

d	v_0	v_0
$r_u(u, w, d) = 0$	$r_u(v_1, v_0) = 0$	$v_1 = \phi(v_0)$
$\tilde{\theta}(d) = \theta(u, d)$	$v_2 = \theta(v_1, v_0)$	$v_2 = \theta(v_1, v_0)$
	$y = v_2$	$y = v_2$

Table 1: Primal analysis and computation of the QoI.

where $J \in \mathrm{Lin}(\mathcal{R}^d, \mathcal{R})$ is the linear operator that eats vectors in $\mathcal{R}^d$ and spits out real numbers in $\mathcal{R}$; $J = (\nabla\theta)^T$. This must be the case as $\theta(d + \Delta d) \approx \theta(d) + \nabla\theta(d) \cdot \Delta d$ is a scalar, i.e. $\theta(d + \Delta d) \in \mathcal{R}$ and $\Delta d \in \mathcal{R}^d$ and hence to be dimensionally consistent $\nabla\theta(d) \in \mathcal{R}^d$.

In the AD reverse mode we examine the action of the transpose of the Jacobian on Δy, i.e. $J^T \Delta y$. To begin we define the *adjoint* of the variable v_i wrt. y as

$$\bar{v}_i = \frac{\partial y}{\partial v_i} \tag{129}$$

Next we work *backwards* through the computations to evaluate $J = Dy/Dv_0$, cf. Table 2. In the progression note that

1. The derivatives $\bar{v}_i = \frac{\partial y}{\partial v_i}$ corresponding to each primal block step are evaluated (if they are nonzero).
2. The derivatives are accumulated, i.e. summed, as we progress *backwards* through the analysis steps.
3. The *adjoint* ***displacement*** w is defined such that $(\partial r_u/\partial v_1)^T\, w = (\partial\theta/\partial v_1)^T = \bar{v}_1$ which is in agreement with the previous result, i.e. that w solves $a_u(\delta u, w, d) = \delta\theta(u, d; \delta u)$ for all kinematically admissible δu.
4. Ultimately we end up with the adjoint derivative formula $\partial y/\partial v_0 = \bar{v}_0 = \bar{v}_0 - (\partial r_u/\partial d)^T\, w = (\partial\theta/\partial d)^T - (\partial r_u/\partial d)^T\, w$ which is identical to our earlier result $\delta\tilde{\theta}(d; \delta d) = \delta\theta(u, d; \delta d) - \delta r_u(u, w, d; \delta d)$!
5. The primal analysis and reverse AD sensitivity analysis are illustrated in Figure 14. Two options of the implementation exist.
   (a) Option 1: Sum the contributions as you go, i.e. compute $\bar{v}_0 = \partial\theta/\partial v_0\, \bar{v}_2$ in the block 1 and then update it to $\bar{v}_0 = \bar{v}_0 - (\partial r/\partial v_0)^T\, w$ in block 2.
   (b) Option 2: Push all the contributions to their respective nodes where they are summed, e.g. push the block 1 and block 2 contributions $\bar{v}_0 = \partial\theta/\partial v_0\, \bar{v}_2$ and $\bar{v}_0 = -(\partial r/\partial v_0)^T\, w$ to the final $\bar{v}_0$ node and sum.

Primal step	Reverse mode derivative step
$y = v_2$	$\partial y/\partial v_2 = \bar{v}_2 = \partial y/\partial y = 1$
$v_2 = \theta(v_1, v_0)$	$\partial y/\partial v_1 = \bar{v}_1 = (\partial v_2/\partial v_1)^T \partial y/\partial v_2 = (\partial\theta/\partial v_1)^T \bar{v}_2$ $\partial y/\partial v_0 = \bar{v}_0 = (\partial v_2/\partial v_0)^T \partial y/\partial v_2 = (\partial\theta/\partial v_0)^T \bar{v}_2$
$v_1 = \phi(v_0)$	$\begin{aligned} \partial y/\partial v_0 = \bar{v}_0 &= \bar{v}_0 + (\partial v_1/\partial v_0)^T \partial y/\partial v_1 \\ &= (\partial\theta/\partial v_0)^T \bar{v}_2 + (\partial\phi/\partial v_0)^T \bar{v}_1 \\ &= (\partial\theta/\partial v_0)^T \bar{v}_2 - \left((\partial r_u/\partial v_1)^{-1} \partial r_u/\partial v_0\right)^T \bar{v}_1 \\ &= (\partial\theta/\partial v_0)^T \bar{v}_2 - (\partial r_u/\partial v_0)^T (\partial r_u/\partial v_1)^{-T} (\partial\theta/\partial v_1)^T \bar{v}_2 \\ &= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w\right) \bar{v}_2 \\ &= (\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w \end{aligned}$

Table 2: AD reverse mode sensitivity analysis for QoI of Table 1.

Now we examine the thermo-mechanical case of Figure 13, again assuming $d = p$. For the primal analysis we refer to Table 3 and for the AD reverse sensitivity analysis we refer to Table 4. Here d and u are defined the same as above, and now $T \in \mathcal{R}^T$, $r_u : \mathcal{R}^u \times \mathcal{R}^T \times \mathcal{R}^d \to \mathcal{R}^u$, $r_T : \mathcal{R}^T \times \mathcal{R}^d \to \mathcal{R}^T$ and $\theta : \mathcal{R}^u \times \mathcal{R}^T \times \mathcal{R}^d \to \mathcal{R}$. We define the adjoint displacement w such that $(\partial r_u/\partial v_2)^T w = (\partial\theta/\partial v_2)^T = \bar{v}_2$ and the adjoint temperature ω such that $(\partial r_T/\partial v_1)^T \omega = (\partial\theta/\partial v_1)^T - (\partial r_u/\partial v_1)^T w = \bar{v}_1$. A summary of the procedure is illustrated in and Figure 15. Again, amazingly, as you all have said, we get the same result! Can we make use of this reverse AD to help automate our computations?

The "graph" is a nice way to look at the flow of information, cf. Figures 14 and 15. In the primal analysis each block eats some $v_i, v_j, \cdots$ nodes and spits out a $v_m = \phi_m(v_i, v_j, \cdots)$ node contribution. For the sensitivity analysis the associated "sensitivity" block eats the $\bar{v}_m$ node and spits out the $\bar{v}_i = \partial\phi_m/\partial v_i\, \bar{v}_m, \bar{v}_j = \partial\phi_m/\partial v_j\, \bar{v}_m, \cdots$ node contributions. If an implicit function is used to evaluate v_m in the primal analysis, i.e. if v_m is a solution to $r_m(v_m; v_i, v_j, \cdots) = 0$ for fixed $v_i, v_j, \cdots$, then its "sensitivity" block eats the $\bar{v}_m$ node, computes the adjoint response w_m that solves $(\partial r_m/\partial v_m)^T w_m = \bar{v}_m$ and spits out the $\bar{v}_i = -\partial r_m/\partial v_i\, w_m, \bar{v}_j = -\partial r_m/\partial v_j\, w_m, \cdots$ node contributions.

Figure 16 illustrates the integration of the Section 7.2 filtering and projection operations with the mechanical analysis of Figure 14.

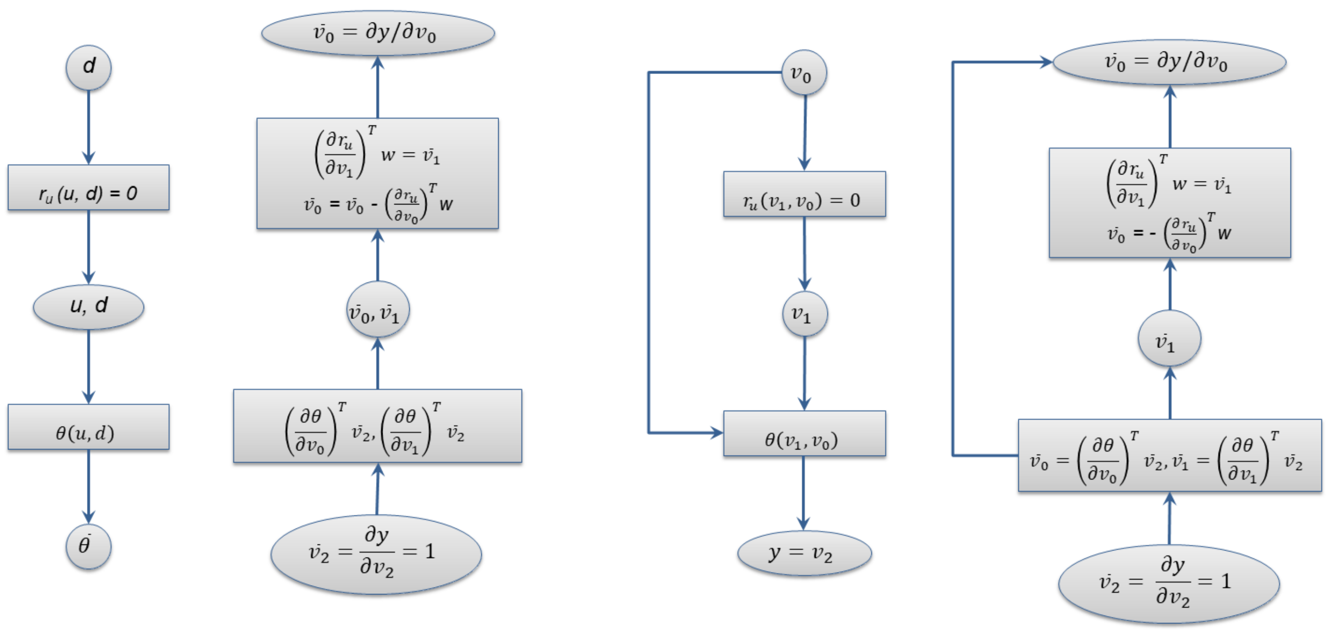


Figure 14: AD illustration. Top to bottom with summations (left two columns) and easily identifiable flow (right two columns). The mappings are $d \rightarrow v_0$, $u \rightarrow v_1$ and $\theta \rightarrow v_2$.

d	v_0	v_0
$r_T(T, \omega, d) = 0$	$r_T(v_1, v_0) = 0$	$v_1 = \phi_T(v_0)$
$r_u(u, T, w, d) = 0$	$r_u(v_2, v_1, v_0) = 0$	$v_1 = \phi_u(v_1, v_0)$
$\tilde{\theta}(d) = \theta(u, T, d)$	$v_2 = \theta(v_2, v_1, v_0)$	$v_2 = \theta(v_2, v_1, v_0)$
	$y = v_3$	$y = v_3$

Table 3: Primal analysis and computation of the thermo-mechanical QoI.

Primal step	Reverse mode derivative step
$y = v_3$	$\partial y/\partial v_3 = \bar{v}_3 = \partial y/\partial y = 1$
$v_3 = \theta(v_2, v_1, v_0)$	$\partial y/\partial v_2 = \bar{v}_2 = (\partial v_3/\partial v_2)^T \partial y/\partial v_3 = (\partial\theta/\partial v_2)^T \bar{v}_3$ $\partial y/\partial v_1 = \bar{v}_1 = (\partial v_3/\partial v_1)^T \partial y/\partial v_3 = (\partial\theta/\partial v_1)^T \bar{v}_3$ $\partial y/\partial v_0 = \bar{v}_0 = (\partial v_3/\partial v_0)^T \partial y/\partial v_3 = (\partial\theta/\partial v_0)^T \bar{v}_3$
$v_2 = \phi_u(v_1, v_0)$	$\partial y/\partial v_1 = \bar{v}_1 = \bar{v}_1 + (\partial v_2/\partial v_1)^T \partial y/\partial v_2$ $= (\partial\theta/\partial v_1)^T \bar{v}_3 + (\partial\phi_u/\partial v_1)^T \bar{v}_2$ $= (\partial\theta/\partial v_1)^T \bar{v}_3 - \left((\partial r_u/\partial v_2)^{-1} \partial r_u/\partial v_1\right)^T \bar{v}_2$ $= (\partial\theta/\partial v_1)^T \bar{v}_3 - \left((\partial r_u/\partial v_2)^{-1} \partial r_u/\partial v_1\right)^T (\partial\theta/\partial v_2)^T \bar{v}_3$ $= (\partial\theta/\partial v_1)^T \bar{v}_3 - (\partial r_u/\partial v_1)^T (\partial r_u/\partial v_2)^{-T} (\partial\theta/\partial v_2)^T \bar{v}_3$ $= \left((\partial\theta/\partial v_1)^T - (\partial r_u/\partial v_1)^T w\right) \bar{v}_3$ $\partial y/\partial v_0 = \bar{v}_0 = \bar{v}_0 + (\partial v_2/\partial v_0)^T \partial y/\partial v_2$ $= (\partial\theta/\partial v_0)^T \bar{v}_3 + (\partial\phi_u/\partial v_0)^T \bar{v}_2$ $= (\partial\theta/\partial v_0)^T \bar{v}_3 - \left((\partial r_u/\partial v_2)^{-1} \partial r_u/\partial v_0\right)^T \bar{v}_2$ $= (\partial\theta/\partial v_0)^T \bar{v}_3 - \left((\partial r_u/\partial v_2)^{-1} \partial r_u/\partial v_0\right)^T (\partial\theta/\partial v_2)^T \bar{v}_3$ $= (\partial\theta/\partial v_0)^T \bar{v}_3 - (\partial r_u/\partial v_0)^T (\partial r_u/\partial v_2)^{-T} (\partial\theta/\partial v_2)^T \bar{v}_3$ $= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w\right) \bar{v}_3$
$v_1 = \phi_T(v_0)$	$\partial y/\partial v_0 = \bar{v}_0 = \bar{v}_0 + (\partial v_1/\partial v_0)^T \partial y/\partial v_1$ $= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w\right) \bar{v}_3 + (\partial\phi_T/\partial v_0)^T \bar{v}_1 =$ $= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w\right) \bar{v}_3 - \left((\partial r_T/\partial v_1)^{-1} \partial r_T/\partial v_0\right)^T \bar{v}_1 =$ $= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w\right) \bar{v}_3 -$ $(\partial r_T/\partial v_0)^T (\partial r_T/\partial v_1)^{-T} \left((\partial\theta/\partial v_1)^T - (\partial r_u/\partial v_1)^T w\right) \bar{v}_3$ $= \left((\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w - (\partial r_T/\partial v_0)^T \omega\right) \bar{v}_3$ $= (\partial\theta/\partial v_0)^T - (\partial r_u/\partial v_0)^T w - (\partial r_T/\partial v_0)^T \omega$

Table 4: AD reverse mode sensitivity analysis for thermo-mechanical QoI of Table 3.

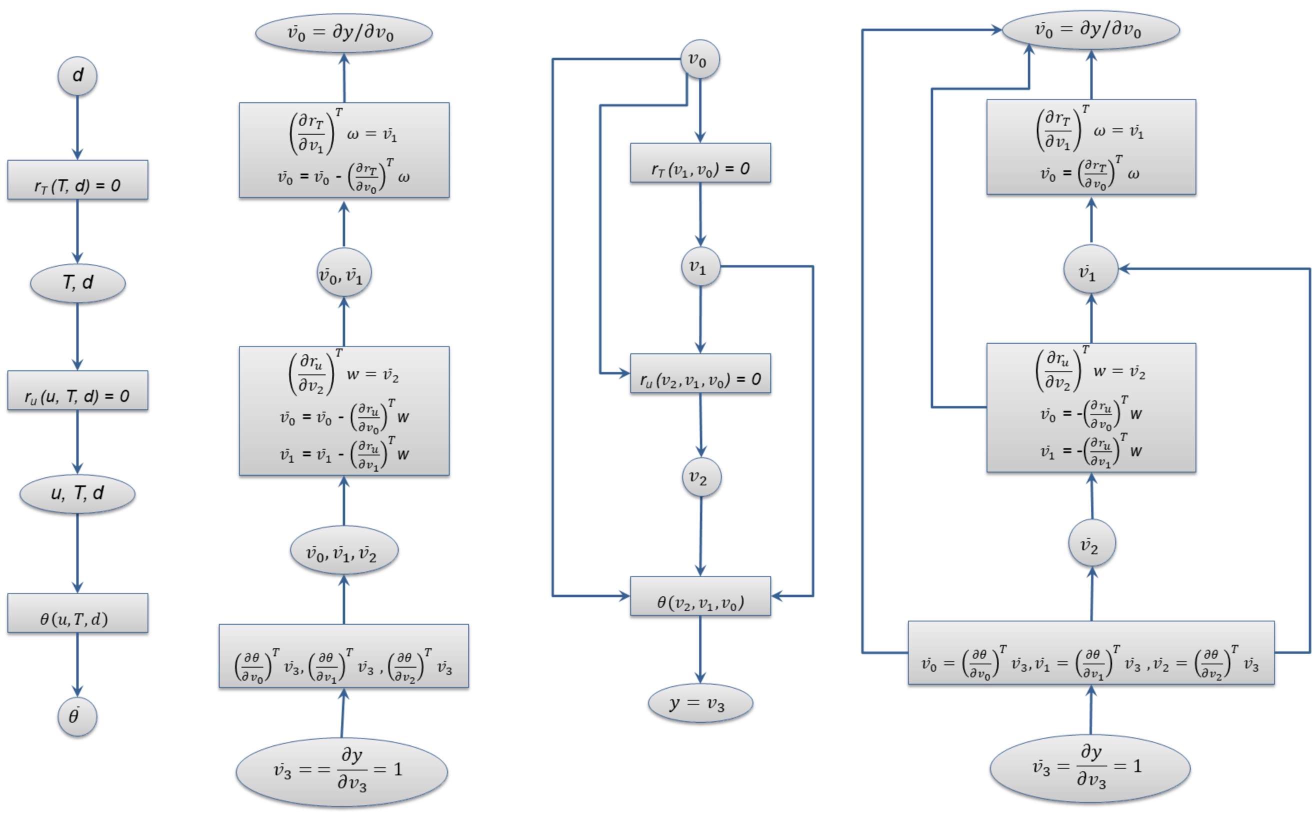


Figure 15: AD illustration for thermo-mechanical problem. Top to bottom with summations (left two columns) and easily identifiable flow (right two columns). The mappings are $d \to v_0$, $T \to v_1$, $u \to v_2$ and $\theta \to v_3$.

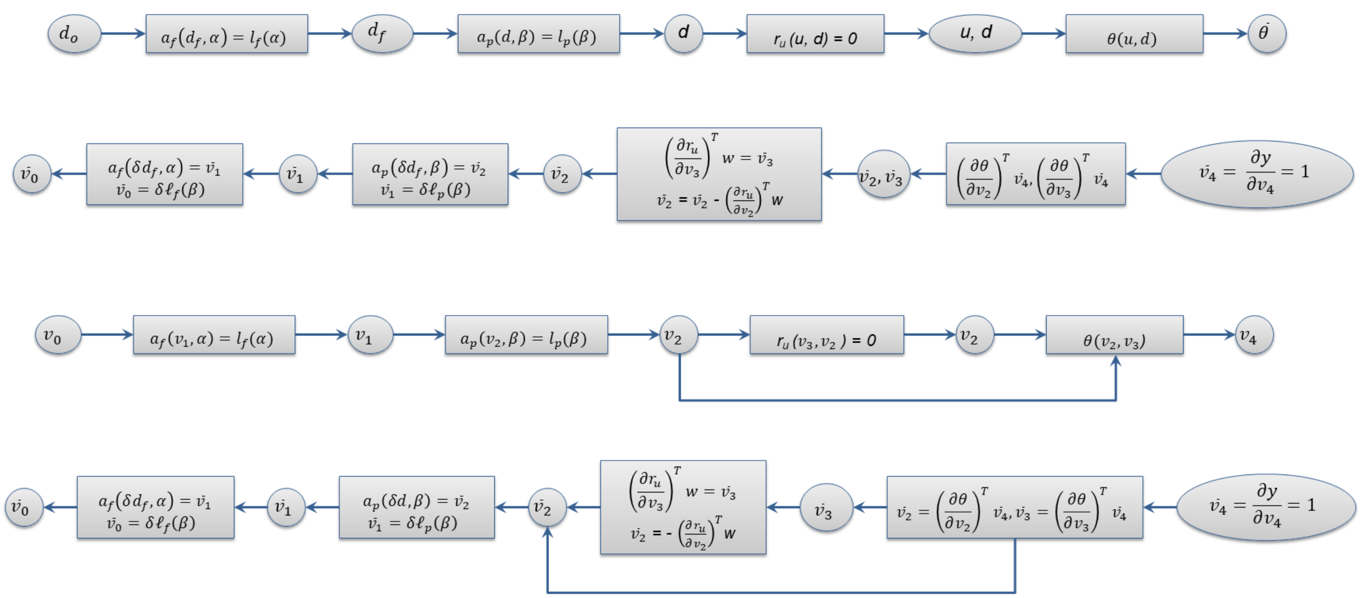


Figure 16: AD illustration for filter-projection-mechanical problem. Left to right with summations (top two rows) and easily identifiable flow (bottom two rows). The mappings are $d_o \to v_0$, $d_f \to v_1$, $d \to v_2$, $u \to v_3$ and $\theta \to v_4$.

12 Notation

We use the direct notation and the fairly standard "letters" as well as some less used tensor conjugate products. Also some other identities are noted

$$
\begin{aligned}
\mathbf{a}\cdot\mathbf{A}\,\mathbf{b} &= (\mathbf{A}^T\,\mathbf{a})\cdot\mathbf{b}\\
(\mathbf{a}\otimes\mathbf{b})\cdot\mathbf{A} &= \mathbf{a}\cdot\mathbf{A}\,\mathbf{b}\\
\mathbf{A}\cdot\mathbf{B} &= \mathrm{tr}(\mathbf{A}^T\mathbf{B})\\
\mathbf{A}\cdot\mathbb{A}[\mathbf{B}] &= \mathbb{A}^T[\mathbf{A}]\cdot\mathbf{B}\\
(\mathbf{A}\otimes\mathbf{B})[\mathbf{C}] &= (\mathbf{B}\cdot\mathbf{C})\,\mathbf{A}\\
(\mathbf{A}\otimes\mathbf{B})^T &= (\mathbf{B}\otimes\mathbf{A})\\
(\mathbf{A}\boxtimes\mathbf{B})[\mathbf{C}] &= \mathbf{A}\,\mathbf{C}\,\mathbf{B}^T\\
(\mathbf{A}\boxtimes\mathbf{B})^T &= (\mathbf{A}^T\boxtimes\mathbf{B}^T)\\
(\mathbf{A}\boxtimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D})(\mathbf{E}\boxtimes\mathbf{F}) &= \mathbf{A}\mathbf{C}\mathbf{B}^T\otimes\mathbf{E}^T\mathbf{D}\mathbf{F}\\
(\mathbf{A}\boxtimes\mathbf{B})\,\mathbb{T} &= \mathbb{T}\,(\mathbf{B}\boxtimes\mathbf{A})
\end{aligned}
\tag{130}
$$

where $\mathbf{a}$ and $\mathbf{b}$ are arbitrary vectors, $\mathbf{A}$, $\mathbf{B}$, ... are arbitrary 2-tensors and $\mathbb{A}$ is a 4-tensor. In the above $(\mathbf{A}\otimes\mathbf{B})$ and $(\mathbf{A}\boxtimes\mathbf{B})$ are a 4-tensors with components $(\mathbf{A}\otimes\mathbf{B})_{ijkl} = A_{ij}\,B_{kl}$ and $(\mathbf{A}\boxtimes\mathbf{B})_{ijkl} = A_{ik}\,B_{jl}$. 4-tensors linearly transforms 2-tensors into 2-tensors. In the above we make use of one of our special 4-tensors, the identity, transposer and symmetrizer defined such that

$$
\begin{aligned}
\mathbb{I}[\mathbf{A}] &= \mathbf{A}\\
\mathbb{T}[\mathbf{A}] &= \mathbf{A}^T\\
\mathbb{S}[\mathbf{A}] &= \tfrac{1}{2}\,(\mathbf{A}^T+\mathbf{A})
\end{aligned}
\tag{131}
$$

Note that $\mathbb{S} = \frac{1}{2}(\mathbb{I}+\mathbb{T})$.

To *ease* the numerical implementation we utilize the matrix Kronecker product defined as the $mr \times ns$ matrix

$$
\mathbf{A}\odot\mathbf{B} = \begin{bmatrix} A_{11}\,\mathbf{B} & A_{12}\,\mathbf{B} & \dots & A_{1n}\,\mathbf{B}\\ A_{21}\,\mathbf{B} & A_{22}\,\mathbf{B} & \dots & A_{2n}\,\mathbf{B}\\ \vdots & \vdots & \ddots & \vdots\\ A_{m1}\,\mathbf{B} & A_{m2}\,\mathbf{B} & \dots & A_{mn}\,\mathbf{B} \end{bmatrix}
\tag{132}
$$

for any $m\times n$ matrix $\mathbf{A}$ and $p\times q$ matrix $\mathbf{B}$. The i,j components of $\mathbf{A}\odot\mathbf{B}$ are given by

$$
(A\odot B)_{i,j} = A_{\mathrm{floor}((i-1)/p)+1,\,\mathrm{floor}((j-1)/q)+1}\;B_{i-\mathrm{floor}((i-1)/p)\,p,\,j-\mathrm{floor}((j-1)/q)\,q}
\tag{133}
$$

where the floor function rounds a number to the next smaller integer, e.g. $\mathrm{floor}(3/2)=1$.

It can be verified that

$$
\begin{aligned}
(\mathbf{A}\odot\mathbf{B})^T &= (\mathbf{A}^T\odot\mathbf{B}^T)\\
(\mathbf{A}\odot\mathbf{B})(\mathbf{D}\odot\mathbf{E}) &= (\mathbf{A}\,\mathbf{D})\odot(\mathbf{B}\,\mathbf{E})\\
\mathrm{vec}\,(\mathbf{A}\,\mathbf{X}\,\mathbf{B}) &= (\mathbf{B}^T\odot\mathbf{A})\,\mathrm{vec}\,(\mathbf{X})
\end{aligned}
\tag{134}
$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{D}$ and $\mathbf{E}$ are conforming matrices and the $\mathrm{vec}\,()$ operator stacks the columns of a matrix into a vector, e.g. for the 2-tensor $\mathbf{A}$

$$\mathrm{vec}\,(\mathbf{A}) = \begin{bmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{12} \\ A_{22} \\ A_{32} \\ A_{13} \\ A_{23} \\ A_{33} \end{bmatrix} \tag{135}$$

We also define the $\mathrm{mat}\,()$ operator which transforms 4-tensors $\mathbb{A}$ into matrices such that

$$\mathrm{mat}\,(\mathbb{A}) = \begin{bmatrix} A_{1111} & A_{1112} & A_{1113} & A_{1112} & A_{1122} & A_{1123} & A_{1132} & A_{1132} & A_{1133} \\ A_{2111} & A_{2112} & A_{2113} & A_{2112} & A_{2122} & A_{2123} & A_{2132} & A_{2132} & A_{2133} \\ A_{3111} & A_{3112} & A_{3113} & A_{3112} & A_{3122} & A_{3123} & A_{3132} & A_{3132} & A_{3133} \\ A_{1211} & A_{1212} & A_{1213} & A_{1212} & A_{1222} & A_{1223} & A_{1232} & A_{1232} & A_{1233} \\ A_{2211} & A_{2212} & A_{2213} & A_{2212} & A_{2222} & A_{2223} & A_{2232} & A_{2232} & A_{2233} \\ A_{3211} & A_{3212} & A_{3213} & A_{3212} & A_{3222} & A_{3223} & A_{3232} & A_{3232} & A_{3233} \\ A_{1311} & A_{1312} & A_{1313} & A_{1312} & A_{1322} & A_{1323} & A_{1332} & A_{1332} & A_{1333} \\ A_{2311} & A_{2312} & A_{2313} & A_{2312} & A_{2322} & A_{2323} & A_{2332} & A_{2332} & A_{2333} \\ A_{3311} & A_{3312} & A_{3313} & A_{3312} & A_{3322} & A_{3323} & A_{3332} & A_{3332} & A_{3333} \end{bmatrix} \tag{136}$$

It can then be verified that

$$\begin{aligned} \mathbf{A} \cdot \mathbf{B} &= \mathrm{vec}\,(\mathbf{A})^T \,\mathrm{vec}\,(\mathbf{B}) \\ \mathrm{mat}\,(\mathbf{A} \otimes \mathbf{B}) &= \mathrm{vec}\,(\mathbf{A})\,\mathrm{vec}\,(\mathbf{B})^T \\ \mathrm{mat}\,(\mathbf{A} \boxtimes \mathbf{B}) &= \mathbf{B} \odot \mathbf{A} \\ \mathrm{mat}\left(\mathbb{A}^T\right) &= \mathrm{mat}\,(\mathbb{A})^T \\ \mathrm{mat}\,(\mathbb{A}\,\mathbb{B}) &= \mathrm{mat}\,(\mathbb{A})\ \mathrm{mat}\,(\mathbb{B}) \end{aligned} \tag{137}$$

12.1 Notation use in hyperelasticity

I use this kronecker product and vector and matrix notations to “simplify” solid mechanics derivations. Let us consider hyperelasticity wherein in the relation between the First Piola-Kirchhoff stress $\mathbf{P}$, second Piola-Kirchhoff $\mathbf{S}$ and Cauchy stress $\boldsymbol{\sigma}$ is such that

$$\begin{aligned} \mathbf{P} &= \det \mathbf{F}\, \boldsymbol{\sigma} \mathbf{F}^{-T} \\ \mathbf{S} &= \det \mathbf{F}\, \mathbf{F}^{-1}\, \boldsymbol{\sigma} \mathbf{F}^{-T} \\ \mathbf{P} &= \mathbf{F}\, \mathbf{S} \end{aligned} \tag{138}$$

where $\mathbf{F}$ is the deformation gradient.

In hyperelasticity it is know that

$$\mathbf{P}(\nabla \mathbf{u}) = \nabla \psi(\mathbf{F}) \tag{139}$$

where ψ is the strain energy density. Objectivity requires

$$\psi(\mathbf{F}) = \tilde{\psi}(\mathbf{C}) \tag{140}$$

where $\mathbf{C} = \mathbf{F}^T \mathbf{F}$ is the right Green Cauchy strain tensor. Using the fact that

$$\begin{aligned} \delta \mathbf{C} &= \delta \mathbf{F}^T \mathbf{F} + \mathbf{F}^T \delta \mathbf{F} = \mathbf{I} \, \delta \mathbf{F}^T \mathbf{F} + \mathbf{F}^T \delta \mathbf{F} \, \mathbf{I} \\ &= (\mathbf{I} \boxtimes \mathbf{F}^T)[\delta \mathbf{F}^T] + (\mathbf{F}^T \boxtimes \mathbf{I})[\delta \mathbf{F}] = (\mathbf{I} \boxtimes \mathbf{F}^T) \, \mathbb{T}[\delta \mathbf{F}] + (\mathbf{F}^T \boxtimes \mathbf{I})[\delta \mathbf{F}] = (\mathbb{T} + \mathbb{I})(\mathbf{F}^T \boxtimes \mathbf{I})[\delta \mathbf{F}] \\ &= 2 \, \mathbb{S} \, (\mathbf{F}^T \boxtimes \mathbf{I})[\delta \mathbf{F}] \end{aligned} \tag{141}$$

we now replace (139) with

$$\begin{aligned} \mathbf{P}(\nabla \mathbf{u}) &= 2 \, \left(\mathbb{S} \, (\mathbf{F}^T \boxtimes \mathbf{I})\right)^T [\nabla \tilde{\psi}(\mathbf{C})] = 2 \, (\mathbf{F}^T \boxtimes \mathbf{I})^T \, \mathbb{S}^T [\nabla \tilde{\psi}(\mathbf{C})] = 2 \, (\mathbf{F} \boxtimes \mathbf{I}) \, \mathbb{S}[\nabla \tilde{\psi}(\mathbf{C})] = 2 \, (\mathbf{F} \boxtimes \mathbf{I})[\nabla \tilde{\psi}(\mathbf{C})] \\ &= 2 \, \mathbf{F} \, \nabla \tilde{\psi}(\mathbf{C}) \end{aligned} \tag{142}$$

where we use the fact that $\nabla \tilde{\psi}(\mathbf{C})$ is symmetric so that $\mathbb{S}[\nabla \tilde{\psi}(\mathbf{C})] = \nabla \tilde{\psi}(\mathbf{C})$. Comparing (138) to (142) we see that

$$\mathbf{S} = 2 \, \nabla \tilde{\psi}(\mathbf{C}) \tag{143}$$

To evaluate the incremental stress $\delta \mathbf{P}$ we differentiate the above, i.e.

$$\delta \mathbf{P} = D^2 \psi(\mathbf{F})[\delta \mathbf{F}] \tag{144}$$

or equivalently that

$$\begin{aligned} \delta \mathbf{P}(\nabla \mathbf{u}; \nabla \delta \mathbf{u}) &= 2 \, \delta \mathbf{F} \, \nabla \tilde{\psi}(\mathbf{C}) + 2 \, \mathbf{F} \, D^2 \tilde{\psi}(\mathbf{C})[\delta \mathbf{C}] = 2 \, \mathbf{I} \, \delta \mathbf{F} \, \nabla \tilde{\psi}(\mathbf{C}) + 4 \, \mathbf{F} \, D^2 \tilde{\psi}(\mathbf{C}) \, \mathbb{S} \, (\mathbf{F}^T \boxtimes \mathbf{I})[\delta \mathbf{F}] \, \mathbf{I} \\ &= \underbrace{\left(2 \, (\mathbf{I} \boxtimes \nabla \tilde{\psi}(\mathbf{C})) + 4 \, (\mathbf{F} \boxtimes \mathbf{I}) \, D^2 \tilde{\psi}(\mathbf{C}) \, \mathbb{S} \, (\mathbf{F} \boxtimes \mathbf{I})^T\right)}_{\mathbb{A}} [\delta \mathbf{F}] \end{aligned} \tag{145}$$

We can readily put the stress and incremental stress into vector form for finite element computation as

$$\begin{aligned} \operatorname{vec}(\mathbf{P}(\nabla \mathbf{u})) &= 2 \, (\mathbf{I} \odot \mathbf{F}) \operatorname{vec}\left(\nabla \tilde{\psi}(\mathbf{C})\right) \\ \operatorname{vec}(\delta \mathbf{P}(\nabla \mathbf{u}; \nabla \delta \mathbf{u})) &= \underbrace{\left(2 \, (\nabla \tilde{\psi}(\mathbf{C}) \odot \mathbf{I}) + 4 \, (\mathbf{I} \odot \mathbf{F}) \operatorname{mat}\left(D^2 \tilde{\psi}(\mathbf{C})\right) \operatorname{mat}(\mathbb{S}) \, (\mathbf{I} \odot \mathbf{F})^T\right)}_{\operatorname{mat}(\mathbb{A})} \operatorname{vec}(\delta \mathbf{F}) \end{aligned} \tag{146}$$

Now we consider the internal force vector in the FEM, i.e.

$$r(\mathbf{u}) = \int_\Omega \nabla\mathbf{w}\cdot\mathbf{P}(\nabla\mathbf{u})\,dv = \sum_e \mathbf{W}_e^T \underbrace{\int_{\Omega_e} \mathbf{G}_e^T\,\mathrm{vec}\,(\mathbf{P})\ dv}_{\mathbf{R}_e} \tag{147}$$

where $\mathbf{u}$ is the kinematically admissible displacement, $\mathbf{w}$ is the kinematically admissible weighting function, $\mathbf{R}_e$ is the element Ω_e internal force vector, $\mathbf{W}_e$ is the vector of element Ω_e displacement dofs and $\mathbf{G}_e$ is the element Ω_e basis function gradient operator defined such that $\mathrm{vec}\,(\nabla\mathbf{w}) = \mathbf{G}_e\,\mathbf{W}_e$ for $\mathbf{x}\in\Omega_e$. For the tangent matrix we have

$$\delta r(\mathbf{u};\delta\mathbf{u}) = \int_\Omega \nabla\mathbf{w}\cdot\delta\mathbf{P}(\nabla\mathbf{u};\nabla\delta\mathbf{u})\,dv = \sum_e \mathbf{W}_e^T \underbrace{\int_{\Omega_e} \mathbf{G}_e^T\,\mathrm{mat}\,(\mathbb{A})\ \mathbf{G}_e\,dv}_{\mathbf{K}_e}\,\mathrm{vec}\,(\mathbf{P}) \tag{148}$$

wherein we see that the element Ω_e tangent stiffness matrix $\mathbf{K}_e$ is identical in form to that in linear elasticity.

13 Nomenclature

The following nomenclature is somewhat standard.

When we have functions of functions, e.g. $f:\mathcal{U}\to\mathcal{R}$ where $\mathcal{U}$ is some function space, we define the *variation* $f(u;\delta u)$ of the function f at $u\in\mathcal{U}$ in the direction $\delta u\in\mathcal{U}$, if it exists, such that

$$\begin{aligned}\delta f(u;\delta u) &= \lim_{\epsilon\to 0}\frac{1}{\epsilon}(f(u+\epsilon\,\delta u)-f(u))\\ &= \frac{d}{d\,\epsilon}f(u+\epsilon\,\delta u)\bigg|_{\epsilon=0}\end{aligned} \tag{149}$$

Note that $\delta f(u;\delta u)$ operates linearly on δu and hence $\delta f(u;\cdot)$ is a *linear form.* The semicolon is used to denote the direction in which the variation is taken and hence the argument which is treated in a linear manner. Extending this concept we define the *second variation* $\delta^2 f(u;\delta u,\delta v)$ of the function f at $u\in\mathcal{U}$ in the directions $\delta u\in\mathcal{U}$ and $\delta v\in\mathcal{U}$, if it exists, such that

$$\delta^2 f(u;\delta u,\delta v) = \frac{d^2}{d\,\epsilon\,d\,\beta}f(u+\epsilon\,\delta u+\beta\,\delta v)\bigg|_{\epsilon=\beta=0} \tag{150}$$

Note that $\delta^2 f(u;\delta u,\delta v)$ operates linearly on δu and δv and hence $\delta^2 f(u;\cdot,\cdot)$ is a *bilinear form.*

A PDE can be expressed in a *strong form*, e.g. for elastostatics we find the displacement $\mathbf{u}$ that satisfies

$$\begin{aligned}\mathrm{div}\mathbf{P}(\nabla\mathbf{u},\mathbf{d})+\mathbf{b}(\mathbf{d}) &= \mathbf{0} && \text{for } \mathbf{x}\in\Omega\\ \mathbf{P}(\nabla\mathbf{u})\mathbf{n} &= \mathbf{t}^p(\mathbf{d}) && \text{for } \mathbf{x}\in A^t\\ \mathbf{u} &= \mathbf{u}^p(\mathbf{d}) && \text{for } \mathbf{x}\in A^u\end{aligned} \tag{151}$$

where $\mathbf{P}$, $\mathbf{b}$, $\mathbf{t}^p$, $\mathbf{u}^p$ and $\mathbf{d} \in \mathcal{D}$ are the First Piola Kirchhoff stress, body load, prescribed surface traction $\mathbf{t}^p$ over A^t prescribed displacement $\mathbf{u}^p$ over $A^u = \partial\Omega \backslash A^t$ and design fields; the last belongs to the set of admissible designs $\mathcal{D}$. For conciseness we suppress the position in the arguments of the above fields.

We can state the above PDE in its *weak form*: Find the $\mathbf{u} \in \mathcal{U}$ such that

$$r(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}) = 0 = \int_\Omega (\nabla\delta\mathbf{w} \cdot \mathbf{P}(\nabla\mathbf{u}, \mathbf{d}) - \delta\mathbf{w} \cdot \mathbf{b}(\mathbf{d}))\, dv - \int_{A^t} \delta\mathbf{w} \cdot \mathbf{t}^p(\mathbf{d})\, da \tag{152}$$

for all $\delta\mathbf{w} \in \mathcal{U}$ where $\mathcal{U}$ is the set of kinematically admissible functions. In (152), $r : \mathcal{U} \times \mathcal{D} \times \mathcal{U} \to \mathcal{R}$ is the *residual*. The residual is linear with respect to the weighting function $\delta\mathbf{w}$ and hence $r(\mathbf{u}, \mathbf{d}, \cdot) : \mathcal{U} \to \mathcal{R}$ is a *linear form* ; it may be nonlinear wrt its other arguments. For this reason, we denote the weighting function with $\delta\mathbf{w}$ rather than $\mathbf{w}$ to emphasize the linearity and the similarity to the variation.

To solve (152) we use Newton's method wherein we find the kinematically admissible $\Delta\mathbf{u}$ such that

$$-r(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}) = \delta_u r(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}; \Delta\mathbf{u}) = \int_\Omega \nabla\delta\mathbf{w} \cdot \frac{\partial\mathbf{P}}{\partial\nabla\mathbf{u}}[\nabla\Delta\mathbf{u}]\, dv \tag{153}$$

for all kinematically admissible $\delta\mathbf{w}$. In the above, $\delta_u r(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}; \Delta\mathbf{u})$ is the *tangent*. The tangent operates linearly on the weighting function $\delta\mathbf{w}$ and the differential $\Delta\mathbf{u}$ and hence $\delta_u r(\mathbf{u}, \mathbf{d}, \cdot; \cdot) : \mathcal{U} \times \mathcal{U} \to \mathcal{R}$ is a *bilinear form*; it may be nonlinear wrt its other arguments. In (153) we see the *material tangent* $\partial\mathbf{P}/\partial\nabla\mathbf{u}$, i.e. the derivative of the constitutive relation wrt. the response.

We may wish to partition the residual and tangent into their *internal* and *external* parts, e.g. the *internal residual* is $r_i(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}) = \int_\Omega \nabla\delta\mathbf{w} \cdot \mathbf{P}(\nabla\mathbf{u}, \mathbf{d})\, dv$ and the *external residual* is $r_e(\mathbf{u}, \mathbf{d}, \delta\mathbf{w}) = -\int_\Omega \delta\mathbf{w} \cdot \mathbf{b}(\mathbf{d})\, dv - \int_{A^t} \delta\mathbf{w} \cdot \mathbf{t}^p(\mathbf{d})\, da$.

Having obtained the displacement $\mathbf{u}$ for the given design $\mathbf{d}$ we evaluate *Quantities of Interest (QoIs)* $\theta : \mathcal{D} \times \mathcal{U} \to \mathcal{R}$ which are used to define optimization cost and constraint functions. Each QoI is of the form [10]

$$\theta(\mathbf{d}, \mathbf{u}) = \int_\Omega \pi(\mathbf{u}, \nabla\mathbf{u}, \mathbf{d})\, dv \tag{154}$$

and hence is a *functional*, i.e. a mapping from the design and displacement spaces into the real numbers; the mapping can be nonlinear. In the above we say that $\int_\Omega \pi(\mathbf{u}, \nabla\mathbf{u}, \mathbf{d})\, dv$ is an *integral* and $\pi(\mathbf{u}, \nabla\mathbf{u}, \mathbf{d})$ is an *integrand*.

For the sensitivity analysis we require the *adjoint load linear form*

$$\delta_u\theta(\mathbf{d}, \mathbf{u}; \delta\mathbf{u}) = \int_\Omega \left(\frac{\partial\pi}{\partial\mathbf{u}} \cdot \delta\mathbf{u} + \frac{\partial\pi}{\partial\nabla\mathbf{u}} \cdot \delta\nabla\mathbf{u} \right) dv \tag{155}$$

and the *explicit derivative linear form*

$$\delta_d\theta(\mathbf{d}, \mathbf{u}; \delta\mathbf{d}) = \int_\Omega \frac{\partial\pi}{\partial\mathbf{d}} \cdot \delta\mathbf{d}\, dv \tag{156}$$

[10] The QoIs may also be defined via surface integrals over $\partial\Omega$.

Note that $\delta_u\theta(\mathbf{d},\mathbf{u};\delta_d\mathbf{u})$ and $\delta\theta(\mathbf{d},\mathbf{u};\delta\mathbf{d})$ are linear wrt $\delta\mathbf{u}$ and $\delta\mathbf{d}$, respectively, however they can be nonlinear wrt $\mathbf{u}$ and $\mathbf{d}$. And hence $\delta_u\theta(\mathbf{d},\mathbf{u};\cdot):\mathcal{U}\to\mathcal{R}$ and $\delta_d\theta(\mathbf{d},\mathbf{u};\cdot):\mathcal{D}\to\mathcal{R}$ are linear forms.

For the sensitivity analysis may also require the *pseudo load bilinear form*

$$\delta_{w,d}r(\mathbf{u},\mathbf{d},\delta\mathbf{w};\delta\mathbf{d}) = 0 = \int_\Omega (\nabla\delta\mathbf{w}\cdot\frac{\partial\mathbf{P}}{\partial\mathbf{d}}[\mathbf{d}] - \delta\mathbf{w}\cdot\frac{\partial\mathbf{b}}{\partial\mathbf{d}}\cdot\delta\mathbf{d})\,dv - \int_{A^t}\delta\mathbf{w}\cdot\frac{\partial\mathbf{t}^p}{\partial\mathbf{d}}\cdot\delta\mathbf{d}\,da \tag{157}$$

It is linear wrt. the weighting function $\delta\mathbf{w}$ and design variation $\delta\mathbf{d}$; it may be nonlinear wrt $\mathbf{u}$ and $\mathbf{d}$; and hence $\delta_{w,d}r(\mathbf{u},\mathbf{d},\cdot;\cdot):\mathcal{U}\times\mathcal{D}\to\mathcal{R}$ is a bilinear form. If we operate on the pseudo load bilinear form with $\delta\mathbf{w}$ (which is obtained from an adjoint analysis) it reduces to the *pseudo load linear form*

$$\delta_d r(\mathbf{u},\mathbf{d},\delta\mathbf{w};\delta\mathbf{d}) = 0 = \int_\Omega (\nabla\delta\mathbf{w}\cdot\frac{\partial\mathbf{P}}{\partial\mathbf{d}}[\mathbf{d}] - \delta\mathbf{w}\cdot\frac{\partial\mathbf{b}}{\partial\mathbf{d}}\cdot\delta\mathbf{d})\,dv - \int_{A^t}\delta\mathbf{w}\cdot\frac{\partial\mathbf{t}^p}{\partial\mathbf{d}}\cdot\delta\mathbf{d}\,da \tag{158}$$

which is linear wrt. the design variation $\delta\mathbf{d}$; it may be nonlinear wrt $\mathbf{u}$ and $\mathbf{d}$; and hence $\delta_d r(\mathbf{u},\mathbf{d},\delta\mathbf{w};\cdot):\mathcal{D}\to\mathcal{R}$ is a linear form.

14 Uncertainty: Polynomial chaos expansion

Polynomial Chaos Expansion (PCE) is *one* possible way to incorporate uncertainty into the QoIs. PCE works really well on problems where the random variables under consideration are smooth, and the probability spaces under consideration are of dimension 100 or less; above that, and Monte Carlo tends to be a better choice due to its dimension-independent convergence behavior. If the random variables under consideration are nonsmooth in sets of sufficiently small Lebesgue measure, then adaptive h-refinement strategies could be used to mitigate and reduce numerical errors incurred by Gibbs-phenomenon-like behavior around jumps. If h-adaptivity is unavailable, then the best choice of method between PCE and Monte Carlo methods is sort of a tossup, and other sorts of surrogate modeling may be better choices.

In the problem at hand, the QoI θ is a function of the random (possibly vector valued) variable $\boldsymbol{\xi}$; we assume θ is differentiable and has finite variance. In the PCE approach we approximate the random function as

$$\theta(\boldsymbol{\xi}) \approx \sum_{i=0}^{N}\Theta_i\,\psi_i(\boldsymbol{\xi}) \tag{159}$$

where the basis functions ψ_i are mutually orthogonal with respect to the probability density function ρ of the random variable $\boldsymbol{\xi}$ over the probability space Ω_ξ. Hermite and Legendre polynomials satisfy these orthogonality criteria for Gaussian and uniform random variables respectively. [11] The number $N+1$ of unknown PCE coefficients

[11] Other classes of random variables (e.g., beta-distributed random variables) are covered by the Wiener-Askey scheme of polynomial chaos, cf. D. Xiu, G. E. Karniadakis, The Wiener-Askey polynomial chaos for stochastic differential equations, SIAM Journal of Scientific Computing, 24(2): 619-644, 2002

Θ_i in the expansion of (159) is dependent on the order of expansion p, i.e. maximum polynomial order of any ψ_i and the number of random variables s, i.e. the stochastic dimension of the expansion; it is

$$N+1 = \frac{(s+p)!}{s!\,p!} \tag{160}$$

The unknown Θ_j in (159) are obtained by projecting both sides of (159) with ψ_j to obtain

$$< \theta\,\psi_j > = \sum_{i=0}^{N} \Theta_i \ < \psi_i\,\psi_j > = \Theta_j \ < \psi_j^2 > \tag{161}$$

where the second equality follows from the orthogonality of the basis functions and use the expectation notation

$$< a > = E(a) = \int_{\Omega_\xi} a(\boldsymbol{\xi})\,\rho(\boldsymbol{\xi})\,d\boldsymbol{\xi} \tag{162}$$

Rearranging (161) gives

$$\Theta_j = \frac{< \theta\,\psi_j >}{< \psi_j^2 >} \tag{163}$$

The denominator in the above can be evaluated analytically, but the numerator must be evaluated via a quadrature, i.e.

$$< \theta\,\psi_j > = \int_{\Omega_\xi} \theta(\boldsymbol{\xi})\,\psi_j(\boldsymbol{\xi})\,\rho(\boldsymbol{\xi})\,d\boldsymbol{\xi} \approx \sum_{q=1}^{n_q} \theta(\boldsymbol{\xi}^{(q)})\,\psi_j(\boldsymbol{\xi}^{(q)})\,w^{(q)} \tag{164}$$

where $\boldsymbol{\xi}^{(q)}$, $w^{(q)}$ and n_q are the quadrature points, their weights and the number of them. Due to the curse of dimensionality it is recommended to combine a Kronrod–Paterson rule with the Smolyak algorithm to perform the quadrature. The Smolyak algorithm, also referred to as"cubature" or "sparse grid quadrature," constructs multi-dimensional quadrature rules from one-dimensional quadrature rules with fewer quadrature points than full tensor product constructions. This is attractive since it 1) maximizes the degree of polynomial that can be integrated exactly with respect to a given weight function for a given number of points and 2) nests, which makes them a good choice for p-refinement of PCE. And this is critical as each $\theta(\boldsymbol{\xi}^{(q)})$ evaluation is expensive.

Knowing the coefficient Θ_j we can replace computationally expensive evaluations of $\theta(\boldsymbol{\xi})$ with the cheap approximation of (159). In addition we can readily compute the mean and variance

$$\begin{aligned} \mu_\theta = E(\theta) &= \Theta_0 \\ \sigma_\theta^2 = E((\theta-\mu_\theta)^2) &= \sum_{i=1}^{N} \Theta_i^2 \ < \psi_i^2 > \end{aligned} \tag{165}$$

The former follows from the fact that $\psi_0(\boldsymbol{\xi}) = 1$. We may also like to evaluate the probability of failure, i.e. the probability that θ exceeds its limit value θ_{lim}

$$P[\theta \geq \theta_{lim}] = \int_{\Omega_x i} H_\epsilon(\theta(\boldsymbol{\xi}) - \theta_{lim})\, \rho(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{166}$$

where H_ϵ is a suitably smooth approximation of the heaviside, i.e. unit step, function. We can evaluate this integral via quadrature or Monte Carlo (MC). The latter gives

$$P[\theta \geq \theta_{lim}] = \frac{n_{\theta(\boldsymbol{\xi}^{(i)})-\theta_{lim}}}{n_{samples}} \approx \frac{1}{n_{samples}} \sum_{i=1}^{n_{samples}} H_\epsilon(\theta(\boldsymbol{\xi}^{(i)}) - \theta_{lim}) \tag{167}$$

where $n_{samples}$ is the number of samples and $n_{\theta(\boldsymbol{\xi}^{(i)})-\theta_{lim}}$ is the number of samples that satisfies $\theta(\boldsymbol{\xi}^{(i)}) - \theta_{lim} > 0$. *We emphasize that the MC is fast as the $\theta(\boldsymbol{\xi}^{(i)})$ are evaluated using the PCE of* (159).

We also use the PCE to represent the sensitivity $\delta\theta$ wrt. the design parameters, i.e.

$$\delta\theta(\boldsymbol{\xi}) = \sum_{i=0}^{N} \delta\Theta_i\, \psi_i(\boldsymbol{\xi}) \tag{168}$$

where

$$\delta\Theta_j = \frac{< \delta\theta\, \psi_j >}{< \psi_j^2 >} \approx \frac{\sum_{q=1}^{n_q} \delta\theta(\boldsymbol{\xi}^{(q)})\, \psi_j(\boldsymbol{\xi}^{(q)})\, w^{(q)}}{< \psi_j^2 >} \tag{169}$$

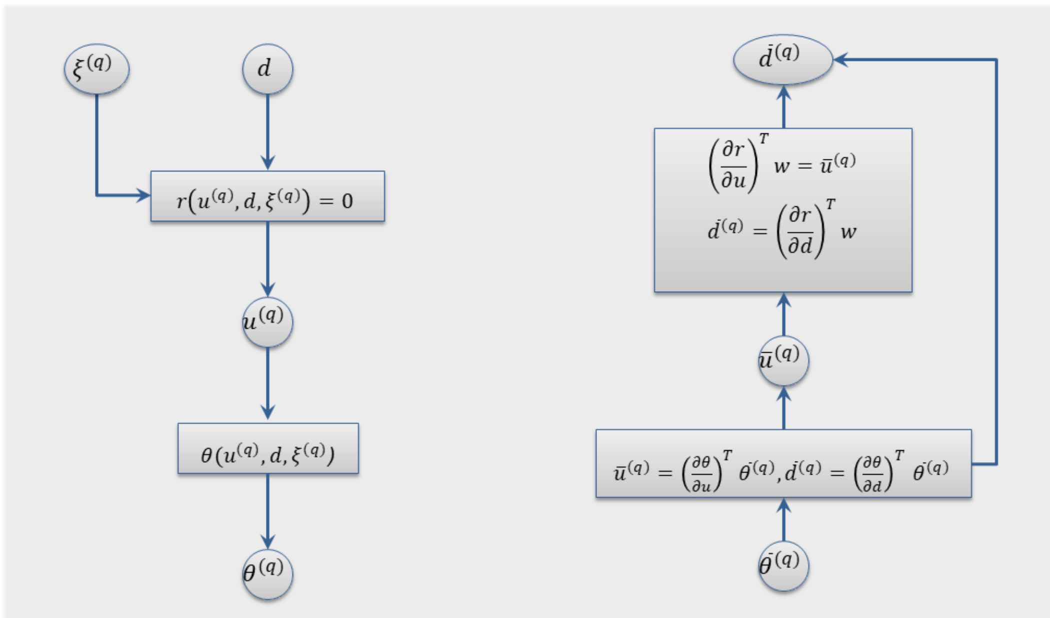
The $\delta\theta(\boldsymbol{\xi}^{(q)})$ in the above are evaluated via an adjoint sensitivity analysis. Having the expansion (168) we can now obtain

$$\begin{aligned}
\delta\mu_\theta &= \delta\Theta_0 \\
\delta\sigma_\theta^2 &= 2 \sum_{i=1}^{N} \delta\Theta_i\, \Theta_i\, < \psi_i^2 > \\
\delta P[\theta \geq \theta_{lim}] &= \frac{1}{n_{samples}} \sum_{i=1}^{n_{samples}} H'_\epsilon(\theta(\boldsymbol{\xi}^{(i)}) - \theta_{lim})\, \delta\theta(\boldsymbol{\xi}^{(i)})
\end{aligned} \tag{170}$$

We again emphasize that the MC is fast as the $\theta(\boldsymbol{\xi}^{(i)})$ and $\delta\theta(\boldsymbol{\xi}^{(i)})$ are evaluated using the PCEs of (159) *and* (168).

A possible algorithm for evaluating the PCEs appears in Figure 17. Once the PCEs are obtained, the mean, variance and probability of failure and their sensitivities can be readily computed.

$\Theta_j = \delta\Theta_j = 0, j = 0, 1, 2, \ldots, N$
For $q = 1, n_q$



For $j = 0, N$

$$\Theta_j = \Theta_j + \theta^{(q)}\, \psi_j\big(\xi^{(q)}\big)\, w^{(q)}$$

$$\delta\Theta_j = \delta\Theta_j + \bar{\theta}^{(q)} \psi_j\big(\xi^{(q)}\big)\, w^{(q)}$$

End

End
For $j = 0, N$

$$\Theta_j = \frac{\Theta_j}{< \psi_j^2 >}$$

$$\delta\Theta_j = \frac{\delta\Theta_j}{< \psi_j^2 >}$$

End

Figure 17: PCE evaluation for θ and $\delta\theta$.

15 Finite difference sensitivity computations

For our optimization we need the gradients of the cost and constraint functions, e.g. f. The sensitivity of the function f at the design $\mathbf{d}$, if it exists, is defined as

$$\frac{\partial f(\mathbf{d})}{\partial d_i} = \lim_{\epsilon_i \to 0} \frac{1}{\epsilon_i} \left(f(\mathbf{d} + \boldsymbol{\epsilon}_i) - f(\mathbf{d}) \right) \tag{171}$$

where $\mathbf{d} = \{d_1, d_2, \cdots, d_{i-1}, d_i, d_{i+1}, \cdots, d_n\}$ and $\mathbf{d} + \boldsymbol{\epsilon}_i = \{d_1, d_2, \cdots, d_{i-1}, d_i + \epsilon_i, d_{i+1}, \cdots, d_n\}$. Second order derivatives, if they exist, are similarly defined by

$$\frac{\partial^2 f(\mathbf{d})}{\partial d_i \partial d_j} = \lim_{\epsilon_i \to 0} \frac{1}{\epsilon_i} \left(\frac{\partial f(\mathbf{d} + \boldsymbol{\epsilon}_i)}{\partial d_i} - \frac{\partial f(\mathbf{d})}{\partial d_i} \right) \tag{172}$$

and so on to define still higher-order derivatives.

Knowing the derivative we can use a first-order Taylor series expansion to obtain

$$f(\mathbf{d} + \boldsymbol{\epsilon}_i) = f(\mathbf{d}) + \frac{\partial f(\mathbf{d})}{\partial d_i} \epsilon_i + \frac{1}{2} \frac{\partial^2 f(\mathbf{d} + \alpha\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} \epsilon_i^2 \tag{173}$$

where $\alpha \in [0, 1]$. Rearranging the above gives

$$\frac{\partial f(\mathbf{d})}{\partial d_i} = \frac{1}{\epsilon_i} \left(f(\mathbf{d} + \boldsymbol{\epsilon}_i) - f(\mathbf{d}) \right) - \frac{1}{2} \frac{\partial^2 f(\mathbf{d} + \alpha\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} \epsilon_i \tag{174}$$

Dropping the ϵ_i term we obtain the *forward finite difference estimate*

$$\frac{\partial f(\mathbf{d})}{\partial d_i} \approx \frac{1}{\epsilon_i} \left(f(\mathbf{d} + \boldsymbol{\epsilon}_i) - f(\mathbf{d}) \right) \tag{175}$$

which is accurate to order ϵ_i.

Repeating the above derivation with a second-order Taylor series expansion we obtain

$$f(\mathbf{d} + \boldsymbol{\epsilon}_i) = f(\mathbf{d}) + \frac{\partial f(\mathbf{d})}{\partial d_i} \epsilon_i + \frac{1}{2} \frac{\partial^2 f(\mathbf{d})}{\partial d_i^2} \epsilon_i^2 + \frac{1}{6} \frac{\partial^2 f(\mathbf{d} + \alpha\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} \epsilon_i^3 \tag{176}$$

where $\alpha \in [0, 1]$. We similarly obtain

$$f(\mathbf{d} - \boldsymbol{\epsilon}_i) = f(\mathbf{d}) - \frac{\partial f(\mathbf{d})}{\partial d_i} \epsilon_i + \frac{1}{2} \frac{\partial^2 f(\mathbf{d})}{\partial d_i^2} \epsilon_i^2 - \frac{1}{6} \frac{\partial^2 f(\mathbf{d} - \alpha'\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} \epsilon_i^3 \tag{177}$$

where again $\alpha' \in [0, 1]$. Subtracting (177) from (176) and rearranging we obtain

$$\frac{\partial f(\mathbf{d})}{\partial d_i} = \frac{f(\mathbf{d} + \boldsymbol{\epsilon}_i) - f(\mathbf{d} - \boldsymbol{\epsilon}_i)}{2\, \epsilon_i} - \left(\frac{1}{12} \frac{\partial^2 f(\mathbf{d} + \alpha\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} + \frac{1}{12} \frac{\partial^2 f(\mathbf{d} - \alpha'\, \boldsymbol{\epsilon}_i)}{\partial d_i^2} \right) \epsilon_i^2 \tag{178}$$

and thus we have the *central finite difference approximation*

$$\frac{\partial f(\mathbf{d})}{\partial d_i} \approx \frac{1}{2\, \epsilon_i} \left(f(\mathbf{d} + \boldsymbol{\epsilon}_i) - f(\mathbf{d} - \boldsymbol{\epsilon}_i) \right) \tag{179}$$

which is accurate to order ϵ_i^2.

$f(\mathbf{d}) =$				
ϵ_i	$f(\mathbf{d}+\boldsymbol{\epsilon}_i)$	$f(\mathbf{d}-\boldsymbol{\epsilon}_i)$	$\frac{1}{\epsilon_i}\left(f(\mathbf{d}+\boldsymbol{\epsilon}_i)-f(\mathbf{d})\right)$	$\frac{1}{2\,\epsilon_i}\left(f(\mathbf{d}+\boldsymbol{\epsilon}_i)-f(\mathbf{d}-\boldsymbol{\epsilon}_i)\right)$
10^{-1}				
10^{-2}				
10^{-3}				
$\vdots$				

Table 5: Finite difference convergence study.

We can test the finite difference gradients by computing them by both the forward and central finite difference methods and by using multiple perturbation sizes, i.e. for each parameter $d_1, d_2, \cdots$ fill out Table 5. When ϵ_i is too small round-off error due to the machine precision of the computations will erode the results and when ϵ_i is too large truncation error due the order ϵ_i and ϵ_i^2 finite difference approximations will erode the results. Hopefully there will be a range of ϵ_i over which consistent finite difference computations are obtained.

We can perform the Taylor remainder convergence test to further verify the accuracy of the derivative approximation. To do this we rearrange (175) to obtain

$$f(\mathbf{d}+\boldsymbol{\epsilon}_i) - f(\mathbf{d}) - \frac{\partial f(\mathbf{d})}{\partial d_i}\,\epsilon_i = \frac{1}{2}\,\frac{\partial^2 f(\mathbf{d}+\alpha\,\boldsymbol{\epsilon}_i)}{\partial d_i^2}\,\epsilon_i^2 \tag{180}$$

Next suppose we have decided that $\partial f/\partial \epsilon_i{}^*$ is a good approximation of the derivative and hence

$$f(\mathbf{d}+\boldsymbol{\epsilon}_i) - f(\mathbf{d}) - \frac{\partial f}{\partial d_i}^*\,\epsilon_i \approx \frac{1}{2}\,\frac{\partial^2 f(\mathbf{d}+\alpha\,\boldsymbol{\epsilon}_i)}{\partial d_i^2}\,\epsilon_i^2 \tag{181}$$

To check this assumption fill out Table 6. Based on (181), as we move down the column halving the value of ϵ_i we should expect to see the value of $f(\mathbf{d}+\boldsymbol{\epsilon}_i) - f(\mathbf{d}) - \frac{\partial f}{\partial d_i}^*\,\epsilon_i$ decrease by a factor of 4.

ϵ_i	$f(\mathbf{d}+\boldsymbol{\epsilon}_i) - f(\mathbf{d}) - \frac{\partial f}{\partial d_i}^*\,\epsilon_i$
10^{-1}	
$10^{-1}/2$	
$10^{-1}/4$	
$10^{-1}/8$	
$\vdots$	

Table 6: Finite difference convergence verification.

Accurate simulations are necessary to compute accurate finite difference sensitivities. As such the solver tolerances in the simulations should be as "tight" as possible. Indeed, as seen in Figure 18, large tolerances in the $f(d)$ and $f(d+\epsilon)$ computations, result in large discrepancies in the possible $\partial f(d)/\partial d \approx \frac{1}{\epsilon}(f(d+\epsilon) - f(d))$

computations, shown in blue, verses the actual sensitivity shown in red. Countering our intuition, we see that larger perturbations ϵ reduce the discrepancies in the possible $\partial f(d)/\partial d \approx \frac{1}{\epsilon}(f(d+\epsilon) - f(d))$ computations.

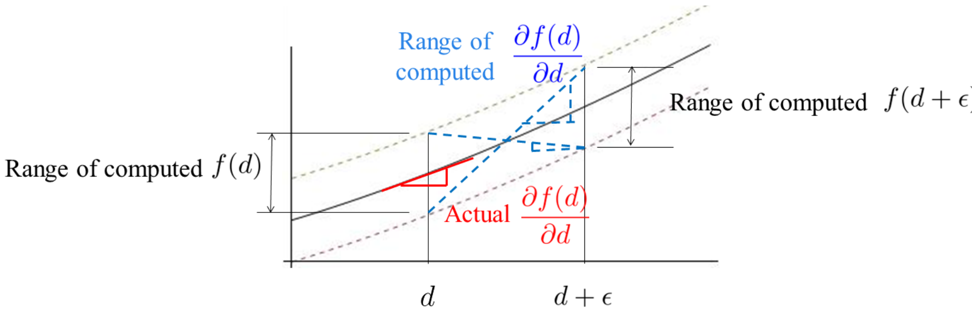


Figure 18: Effects of tolerance and perturbation on finite difference sensitivity computation.

16 Riesz maps

For this material the reader is referred to [1, 2].

There is much going on in Figure 19. So let's discuss it. We see three spaces: the design parameter space D, the design field space Y and the real number line $\mathcal{R}$. The spaces are equipped with the inner products $(\cdot,\cdot)_D$, $(\cdot,\cdot)_Y$ and $(\cdot,\cdot)_{\mathcal{R}}$, making them normed spaces. The mapping $F : D \to Y$ relates D to Y, this is our "parameterization" that maps the design parameters $\mathbf{d} \in D$ into MFEM GridFunctions $\mathbf{y} = F(\mathbf{d}) \in Y$.

Two QoI *functionals*, i.e. functions that eat a function and spit out a real number, are defined. They are $\beta : D \to \mathcal{R}$ and $\theta : Y \to \mathcal{R}$. The functionals are related by composition such that

$$\beta(\mathbf{d}) = \theta(F(\mathbf{d})) = \theta \circ F(\mathbf{d}) \tag{182}$$

i.e. $\beta = \theta \circ F : D \to \mathcal{R}$.

We assume all functions are differentiable. The derivative of β at $\mathbf{d}$ is the *linear* functional $\beta'(\mathbf{d}) \in \mathcal{L}(D, \mathcal{R})$ defined such that

$$\beta'(\mathbf{d})[\delta\mathbf{d}] = \beta(\mathbf{d} + \delta\mathbf{d}) - \beta(\mathbf{d}) + o(\|\delta\mathbf{d}\|_D) \tag{183}$$

for all $\delta\mathbf{d} \in D$. In the above $o(\|\delta\mathbf{d}\|_D)$ is the remainder that tends to zero faster than $\|\delta\mathbf{d}\|_D$. The derivative of θ at $\mathbf{y}$ is defined in a similar fashion, notably $\theta'(\mathbf{y}) \in \mathcal{L}(Y, \mathcal{R})$. The derivative of F at $\mathbf{d}$ is also similarly defined; it is the linear function $DF(\mathbf{d}) \in \mathcal{L}(D, Y)$ defined such that

$$DF(\mathbf{d})[\delta\mathbf{d}] = F(\mathbf{d} + \delta\mathbf{d}) - F(\mathbf{d}) + o(\|\delta\mathbf{d}\|_D) \tag{184}$$

for all $\delta\mathbf{d} \in D$. We use the square brackets $[\cdot]$ rather than the usual parentheses $(\cdot)$ surrounding the arguments of these functions, e.g. when the linear function $DF(\mathbf{d}) : D \to Y$ operates on $\delta\mathbf{d} \in D$ we write $DF(\mathbf{d})[\delta\mathbf{d}]$, to emphasize the linearity.

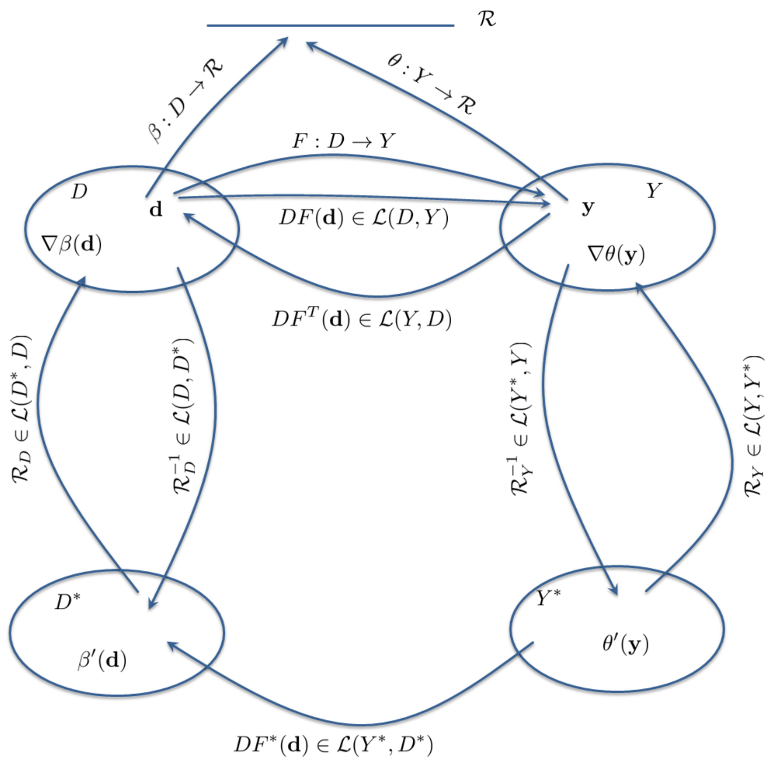


Figure 19: Riesz map, transpose, adjoint, derivative and gradient.

The *dual space* D^* of D is the space of all linear functionals that eat elements of D and spit out real numbers, i.e. $D^* = \mathcal{L}(D, \mathcal{R})$. The *Riesz representation theorem* tells us that for each $\mathbf{d}^* \in D^*$ there is a unique element $\mathbf{d} \in D$ defined such that $(\mathbf{d}, \mathbf{e})_D = \mathbf{d}^*[\mathbf{e}]$ for all $\mathbf{e} \in D$. It continues to say the converse, i.e. for each $\mathbf{d} \in D$ there is a unique linear functional $\mathbf{d}^* \in D^*$ such that $\mathbf{d}^*[\mathbf{e}] = (\mathbf{d}, \mathbf{e})_D$ for all $\mathbf{e} \in D$. The *Riesz map*, which is a linear map, defines this correspondence such that $\mathbf{d} = \mathcal{R}_D[\mathbf{d}^*]$ for any $\mathbf{d}^* \in D^*$ and $\mathbf{d}^* = \mathcal{R}_D^{-1}[\mathbf{d}]$ for and $\mathbf{d} \in D$. In particular, since the derivative $\beta'(\mathbf{d}) \in D^*$ is a linear functional we can define its *gradient* $\nabla\beta(\mathbf{d}) = \mathcal{R}_D[\beta'(\mathbf{d})] \in D$ such that

$$(\nabla\beta(\mathbf{d}), \delta\mathbf{d})_D = \beta'(\mathbf{d})[\delta\mathbf{d}] \tag{185}$$

for all $\delta\mathbf{d} \in D$. The dual space Y^* and gradient $\nabla\theta(\mathbf{y})$ are analogously defined.

The *transpose* $DF^T(\mathbf{d}) : Y \to D \in \mathcal{L}(Y, D)$ of the linear operator $DF(\mathbf{d}) \in \mathcal{L}(D, Y)$ is the unique linear operator that satisfies

$$(DF(\mathbf{d})[\mathbf{d}], \mathbf{y})_Y = (\mathbf{d}, DF^T(\mathbf{d})[\mathbf{y}])_D \tag{186}$$

for all $\mathbf{d} \in D$ and all $\mathbf{y} \in Y$. Using the transpose allows is to evaluate the gradient of $\theta \circ F$, cf. (182). Indeed, we apply the *chain-rule* to (182) to obtain

$$\beta'(\mathbf{d})[\delta\mathbf{d}] = \theta'(\overbrace{F(\mathbf{d})}^{\mathbf{y}})[\overbrace{DF(\mathbf{d})[\delta\mathbf{d}]}^{\delta\mathbf{y}}] \tag{187}$$

Next we use the definitions of the gradients to write the above as

$$(\nabla\beta(\mathbf{d}), \delta\mathbf{d})_D = (\nabla\theta(\overbrace{F(\mathbf{d})}^{\mathbf{y}}), \overbrace{DF(\mathbf{d})[\delta\mathbf{d}]}^{\delta\mathbf{y}})_Y \tag{188}$$

The definition of the transpose gives

$$(\nabla\beta(\mathbf{d}), \delta\mathbf{d})_D = (DF^T(\mathbf{d})[\nabla\theta(F(\mathbf{d}))], \delta\mathbf{d}])_D \tag{189}$$

And finally the arbitrariness of $\delta\mathbf{d}$ gives

$$\nabla\beta(\mathbf{d}) = DF^T(\mathbf{d})[\nabla\theta(F(\mathbf{d}))] \tag{190}$$

The *adjoint* $DF^*(\mathbf{d}) \in \mathcal{L}(Y^*, D^*)$ of linear operator $DF(\mathbf{d}) \in \mathcal{L}(D, Y)$ is the unique linear map that eats linear functionals $\mathbf{y}^* \in Y^*$ and spits out linear functionals $\mathbf{d}^* \in D^*$ such that

$$[DF^*(\mathbf{d})[\mathbf{y}^*] = \mathbf{y}^* \circ DF(\mathbf{d}) \tag{191}$$

i.e. such that

$$(DF^*(\mathbf{d})[\mathbf{y}^*])\,[\delta\mathbf{d}] = \mathbf{y}^*[DF(\mathbf{d})[\delta\mathbf{d}]] \tag{192}$$

for all $\delta\mathbf{d} \in \mathcal{D}$. We now prove that

$$DF^*(\mathbf{d}) \circ \mathcal{R}_Y^{-1} = \mathcal{R}_D^{-1} \circ DF^T(\mathbf{d}) \tag{193}$$

First we look at the left side of (193) and note that $DF^*(\mathbf{d}) \circ \mathcal{R}_Y^{-1} \in \mathcal{L}(Y, D^*)$ and use the definition (192) so that

$$DF^*(\mathbf{d}) \circ \overbrace{\mathcal{R}_Y^{-1}[\mathbf{y}]}^{\mathbf{y}^*} = \overbrace{\mathcal{R}_Y^{-1}[\mathbf{y}]}^{\mathbf{y}^*} \circ DF(\mathbf{d}) \tag{194}$$

Next, from the above (194) we have for any $\delta\mathbf{d} \in D$

$$\overbrace{\mathcal{R}_Y^{-1}[\mathbf{y}]}^{\mathbf{y}^*} \circ DF(\mathbf{d})[\delta\mathbf{d}] = \overbrace{\mathcal{R}_Y^{-1}[\mathbf{y}]}^{\mathbf{y}^*}[\overbrace{DF(\mathbf{d})[\delta\mathbf{d}]}^{\delta\mathbf{y}}] = (\mathbf{y}, \overbrace{DF(\mathbf{d})[\delta\mathbf{d}]}^{\delta\mathbf{y}})_Y = (\overbrace{DF^T(\mathbf{d})[\mathbf{y}]}^{\mathbf{e}\in D}, \delta\mathbf{d})_D = \overbrace{\mathcal{R}_D^{-1}[\overbrace{DF^T(\mathbf{d})[\mathbf{y}]}^{\mathbf{e}\in D}]}^{\mathbf{e}^*\in D^*}[\delta\mathbf{d}] \tag{195}$$

where we used the Riesz representation theorem and definition of the transpose. And finally, upon using (194) and the arbitrariness of $\delta\mathbf{d}$ in (195) and then the arbitrariness of $\mathbf{y}$ we obtain (193).

Rearranging (193) gives us the "formula"

$$DF^*(\mathbf{d}) = \mathcal{R}_D^{-1} \circ DF^T(\mathbf{d}) \circ \mathcal{R}_Y \in \mathcal{L}(Y^*, D^*) \tag{196}$$

So revisiting the chain-rule (187) we see that we can relate the derivatives $\beta'(\mathbf{d}) \in \mathcal{L}(D, D^*$ to $\theta'(F(\mathbf{d})) \in \mathcal{L}(Y, Y^*)$ via

$$\beta'(\mathbf{d}) = DF^*(\mathbf{d})[\theta'(\overbrace{F(\mathbf{d})}^{\mathbf{y}})] = \mathcal{R}_D^{-1} \circ \underbrace{DF^T(\mathbf{d}) \circ \overbrace{\mathcal{R}_Y[\theta'(F(\mathbf{d}))]}^{\nabla\theta(F(\mathbf{d}))}}_{\nabla\beta(\mathbf{d})} \tag{197}$$

As seen in (190) and (197), the transpose and adjoint allow us to "pull-back" the gradient and derivative from the domains Y and Y^* to the domains D and D^*, respectively. Note that *both* Riesz maps appear in the above. So it seems that the Riesz maps must be considered to generate mesh independent designs.

17 Periodic loading

Suppose we have the linear transient system

$$\mathbf{C}(\mathbf{d})\,\dot{\mathbf{T}}(t) + \mathbf{K}(\mathbf{d})\,\mathbf{T}(t) = \mathbf{F}(t) \tag{198}$$

where $\mathbf{C}$, $\mathbf{K}$, $\mathbf{d}$, $\mathbf{T}$, $\mathbf{F}$ and t denote the capacitance matrix, conductivity matrix, design parameter vector, temperature vector, load vector and time. We are interested in a periodic loading over the time interval $[0, P]$ where P is the period. As such we can express

$$\mathbf{F}(t) = \mathbf{A}_0 + \sum_{j=1}^{N} (\mathbf{A}_j \cos(j\,\omega\,t) + \mathbf{B}_j \sin(j\,\omega\,t)) \tag{199}$$

where $\omega = 2\,\pi/P$ and

$$\begin{aligned} \mathbf{A}_0 &= \frac{1}{P}\int_0^P \mathbf{F}(t)\,dt \\ \mathbf{A}_j &= \frac{2}{P}\int_0^P \mathbf{F}(t)\,\cos(j\,\omega\,t)\,dt \\ \mathbf{B}_j &= \frac{2}{P}\int_0^P \mathbf{F}(t)\,\sin(j\,\omega\,t)\,dt \end{aligned} \tag{200}$$

We express the temperature similar to the load, i.e.

$$\mathbf{T}(t) = \mathbf{R}_0 + \sum_{j=1}^{N} (\mathbf{R}_j \cos(j\,\omega\,t) + \mathbf{S}_j \sin(j\,\omega\,t)) \tag{201}$$

so that

$$\dot{\mathbf{T}}(t) = \sum_{j=1}^{N} j\,\omega\,\left(-\mathbf{R}_j\,\sin(j\,\omega\,t) + \mathbf{S}_j\,\cos(j\,\omega\,t)\right) \tag{202}$$

Substituting (199), (201) and (202) into (198) and factoring out the $\cos(j\,\omega\,t)$ and $\sin(j\,\omega\,t)$ leads to a series of problems that we can solve for the $\mathbf{R}_j$ and $\mathbf{S}_j$, i.e.

$$\begin{aligned}
\mathbf{K}(\mathbf{d})\,\mathbf{R}_0 &= \mathbf{A}_0\\
j\,\omega\,\mathbf{C}(\mathbf{d})\,\mathbf{S}_j + \mathbf{K}(\mathbf{d})\,\mathbf{R}_j &= \mathbf{A}_j\\
-j\,\omega\,\mathbf{C}(\mathbf{d})\,\mathbf{R}_j + \mathbf{K}(\mathbf{d})\,\mathbf{S}_j &= \mathbf{B}_j
\end{aligned} \tag{203}$$

Knowing the $\mathbf{R}_j$ and $\mathbf{S}_j$ we can compute the temperature via (201) and subsequently any QoI of the form

$$\Theta = \int_0^P \Pi(\mathbf{T},\mathbf{d})\,dt \tag{204}$$

We now want to evaluate the sensitivity of the above QoI, i.e.

$$\begin{aligned}
\delta\Theta &= \int_0^P \left(\frac{\partial\Pi}{\partial\mathbf{T}}\cdot\delta\mathbf{T} + \frac{\partial\Pi}{\partial\mathbf{d}}\cdot\delta\mathbf{d}\right)\,dt\\
&= \int_0^P \left(\frac{\partial\Pi}{\partial\mathbf{T}}\cdot\left(\delta\mathbf{R}_0 + \sum_{j=1}^{N}\left(\delta\mathbf{R}_j\,\cos(j\,\omega\,t) + \delta\mathbf{S}_j\,\sin(j\,\omega\,t)\right)\right) + \frac{\partial\Pi}{\partial\mathbf{d}}\cdot\delta\mathbf{d}\right)\,dt
\end{aligned} \tag{205}$$

As per usual, we annihilate the implicitly defined variations $\delta\mathbf{R}_j$ and $\delta\mathbf{S}_j$ by using the adjoint method, whereby we subtract the variations of (203) from (205) to obtain

$$\begin{aligned}
\delta\Theta = &\int_0^P \left(\frac{\partial\Pi}{\partial\mathbf{T}}\cdot\left(\delta\mathbf{R}_0 + \sum_{j=1}^{N}\left(\delta\mathbf{R}_j\,\cos(j\,\omega\,t) + \delta\mathbf{S}_j\,\sin(j\,\omega\,t)\right)\right) + \frac{\partial\Pi}{\partial\mathbf{d}}\cdot\delta\mathbf{d}\right)\,dt -\\
&\left(\mathbf{X}_0^T(\delta\mathbf{K}\,\mathbf{R}_0 + \mathbf{K}\,\delta\mathbf{R}_0) + \sum_{j=1}^{N}\left(\mathbf{X}_j^T(j\,\omega\,\delta\mathbf{C}\,\mathbf{S}_j + j\,\omega\,\mathbf{C}\,\delta\mathbf{S}_j + \delta\mathbf{K}\,\mathbf{R}_j + \mathbf{K}\,\delta\mathbf{R}_j) + \right.\right.\\
&\left.\left.\mathbf{Y}_j^T(-j\,\omega\,\delta\mathbf{C}\,\mathbf{R}_j - j\,\omega\,\mathbf{C}\,\delta\mathbf{R}_j + \delta\mathbf{K}\,\mathbf{S}_j + \mathbf{K}\,\delta\mathbf{S}_j)\right)\right)
\end{aligned} \tag{206}$$

where $\mathbf{X}_j$ and $\mathbf{Y}_j$ are the adjoint variables. We annihilate the implicitly defined variations $\delta\mathbf{R}_j$ and $\delta\mathbf{S}_j$ by solving a series of adjoint problems:

$$\begin{aligned}
\mathbf{K}\,\mathbf{X}_0 &= \int_0^P \frac{\partial \Pi}{\partial \mathbf{T}}\,dt \\
-j\,\omega\,\mathbf{C}\,\mathbf{Y}_j + \mathbf{K}\,\mathbf{X}_j &= \int_0^P \frac{\partial \Pi}{\partial \mathbf{T}}\,\cos(j\,\omega\,t)\,dt \\
\omega\,\mathbf{C}\,\mathbf{X}_j + \mathbf{K}\,\mathbf{Y}_j &= \int_0^P \frac{\partial \Pi}{\partial \mathbf{T}}\,\sin(j\,\omega\,t)\,dt
\end{aligned} \tag{207}$$

The above adjoint problems closely resemble the primal problems of (203) when we view, e.g. $\int_0^P \frac{\partial \Pi}{\partial \mathbf{T}}\cos(j\,\omega\,t)\,dt$ as a Fourier coefficient of the periodic adjoint load $\partial\Pi/\partial\mathbf{T}$. Upon solving the adjoint problems, the sensitivity reduces to

$$\delta\Theta = \int_0^P \frac{\partial \Pi}{\partial \mathbf{d}}\cdot\delta\mathbf{d}\,dt - \mathbf{X}_0^T\,\delta\mathbf{K}\,\mathbf{R}_0 + \sum_{j=1}^{N}\left(\mathbf{X}_j^T(j\,\omega\,\delta\mathbf{C}\,\mathbf{S}_j + \delta\mathbf{K}\,\mathbf{R}_j) + \mathbf{Y}_j^T(-j\,\omega\,\delta\mathbf{C}\,\mathbf{R}_j + \mathbf{K}\,\delta\mathbf{S}_j)\right) \tag{208}$$

18 Transient Sensitivity Analysis

Here we consider the sensitivity analysis of a general Quantity of Interest (QoI) which is defined in terms of a design field d and response field u. The design field is obtained from some sort of parameterization with respect to the parameters p such that $d(x) = \hat{d}(x,p)$ where d is the field and $\hat{d}$ is the function that produces the field. The response u for a fixed design d is computed by solving the possibly nonlinear primal problem, i.e IVPDE: find a kinematically admissible u such that

$$\begin{aligned}
m(\dot{u}, w, d) &= r(u, w, d, t) \\
u(0) &= u_0(d)
\end{aligned} \tag{209}$$

for all kinematically admissible fields w. Note that the residual linear form r is linear wrt w but possibly nonlinear wrt u and d while the bilinear form m is linear wrt $\dot{u}$ and w but possibly nonlinear wrt u and d. [12]

We solve the above IVPDE (209) over the time interval $\mathcal{I} = [0, T]$ for the fixed design d. Having the trio $(\dot{u}, u, d)$ we define the QoI

$$\begin{aligned}
\tilde{\theta}_i(d) &= \int_{\mathcal{I}} \theta_i(u, d, t)\,dt \\
&= \int_{\mathcal{I}}\int_{\Omega} \pi(u, d, t)\,dv\,dt \\
&= \int_{\mathcal{I}}\int_{\Omega} \pi(u, d, t)\,dv\,dt - \int_{\mathcal{I}} (m(\dot{u}, w, d) - r(u, w, d, t))\,dt
\end{aligned} \tag{210}$$

[12] Without loss of generality we assume m is a bilinear form, i.e. linear in $\dot{u}$.

where the last equality follows since $m(\dot{u}, w, d) - r(u, w, d, t) = 0$. The variation of the above yields, after some rearranging.

$$
\begin{aligned}
\delta\tilde{\theta}_i(d;\delta d) &= \int_{\mathcal{I}} \overbrace{\int_{\Omega} \frac{\partial \pi}{\partial d} \cdot \delta d\, dv}^{\delta\theta_i(u,d,t;\delta d)}\, dt - \int_{\mathcal{I}} (\delta m(\dot{u}, w, d; \delta d) - \delta r(u, w, d, t; \delta d))\, dt + \\
&\quad \int_{\mathcal{I}} \overbrace{\int_{\Omega} \frac{\partial \pi}{\partial u} \cdot \delta u\, dv}^{\delta\theta_i(u,d,t;\delta u)}\, dt - \int_{\mathcal{I}} (m(\delta\dot{u}, w, d) - a(\delta u, w, d, t))\, dt \\
&= \int_{\mathcal{I}} \delta\theta_i(u, d, t; \delta d)\, dt - \int_{\mathcal{I}} (\delta m(\dot{u}, w, d; \delta d) - \delta r(u, w, d, t; \delta d))\, dt - \\
&\quad m(\delta u, w, d)\bigg|_0^T - \int_{\mathcal{I}} (-m(\delta u, \dot{w}, d) - a(\delta u, w, d, t) - \delta\theta_i(u, d, t; \delta u))\, dt \\
&= \int_{\mathcal{I}} \delta\theta_i(u, d, t; \delta d)\, dt - \int_{\mathcal{I}} (\delta m(\dot{u}, w, d; \delta d) - \delta r(u, w, d, t; \delta d))\, dt + m(\frac{\partial u_0}{\partial d}(d)\delta d, w, d) - \\
&\quad m(\delta u, w, d)\bigg|_T - \int_{\mathcal{I}} (-m(\delta u, \dot{w}, d) - a(\delta u, w, d, t) - \delta\theta_i(u, d, t; \delta u))\, dt
\end{aligned} \tag{211}
$$

where we use the definition of the tangent, i.e. bilinear form wrt δu and w, $a(\delta u, w, d, t) = \delta r(u, w, d, t; \delta u)$, integration by parts and the bilinearity of m.

The second line in the final equality in (211) contains the implicitly defined response variation δu. To annihilate it we require the heretofore arbitrary w to solve the adjoint problem: Find the kinematically admissible w such that

$$
\begin{aligned}
-m(\delta u, \dot{w}, d) &= a(\delta u, w, d, t) + \delta\theta_i(u, d, t; \delta u) \\
w(T) &= 0
\end{aligned} \tag{212}
$$

for all kinematically admissible δu. Note that the adjoint problem is a *terminal value* BVP, not an IVBVP! Because of this it is customary to introduce the change of variables and define the adjoint field $\hat{w}$ such that

$$
\hat{w}(t) = w(T - t) \tag{213}
$$

In this way, we can express (212) as: find the kinematically admissible $\hat{w}$ such that

$$
\begin{aligned}
m(\delta u, \dot{\hat{w}}, d) &= a(\delta u, \hat{w}, d, T - t) + \delta\theta_i(u, d, T - t; \delta u) \\
\hat{w}(0) &= 0
\end{aligned} \tag{214}
$$

for all kinematically admissible δu. In the above, $\hat{w}$ is evaluated at time t while the tangent $a(\delta u, \hat{w}, d, T - t)$ and the adjoint load $\delta\theta_i(u, d, T - t; \delta u)$ are evaluated at time $T - t$. This poses a challenge for the computation as we must complete the primal analysis to evaluate u over the interval $\mathcal{I}$ before beginning the adjoint analysis for $\hat{w}$. That said, once $\hat{w}$ is known, the sensitivity (211) reduces to

$$\delta\tilde{\theta}_i(d;\delta d) = \int_{\mathcal{I}} \delta\theta_i(u,d,t;\delta d)\,dt - \int_{\mathcal{I}} (\delta m(\dot{u},\hat{w}\Big|_{T-t}, d;\delta d) - \delta r(u,\hat{w}\Big|_{T-t}, d,t;\delta d))\,dt + m(\frac{\partial u_0}{\partial d}(d)\delta d, w, d) \quad (215)$$

where all quantities are evaluate at time t, unless specified otherwise.

19 Discrete Transient Sensitivity Analysis

The response u for a fixed design d is computed by solving the possibly nonlinear primal problem, i.e IVPDE: find u such that

$$\begin{aligned} m(d)\,\dot{u} &= r(u,d,t) \\ u(0) &= u_0(d) \end{aligned} \quad (216)$$

We solve the above IVPDE (216) over the time interval $\mathcal{I} = [0,T]$ for the fixed design d. Having the trio $(\dot{u},u,d)$ we define the QoI

$$\begin{aligned} \tilde{\theta}_i(d) &= \int_{\mathcal{I}} \pi(u,d,t)\,dt \\ &= \int_{\mathcal{I}} \pi(u,d,t)\,dt - \int_{\mathcal{I}} w^T\left(m(d)\,\dot{u} - r(u,d,t)\right)dt \end{aligned} \quad (217)$$

where the last equality follows from (216) since $w^T(m(d)\,\dot{u} - r(u,d,t)) = 0$ with w being any vector. The variation of the above yields, after some rearranging.

$$\begin{aligned} \delta\tilde{\theta}_i(d;\delta d) &= \int_{\mathcal{I}} \frac{\partial \pi}{\partial d}\,\delta d\,dt - \int_{\mathcal{I}} w^T\left(\frac{\partial m}{\partial d}\,\delta d\,\dot{u} - \frac{\partial r}{\partial d}\,\delta d\right)dt + \\ &\quad \int_{\mathcal{I}} \frac{\partial \pi}{\partial u}\,\delta u\,dt - \int_{\mathcal{I}} w^T\left(m\,\delta\dot{u} - \frac{\partial r}{\partial u}\,\delta u\right)dt \\ &= \int_{\mathcal{I}} \frac{\partial \pi}{\partial d}\cdot\delta d\,dt - \int_{\mathcal{I}} w^T\left(\frac{\partial m}{\partial d}\,\dot{u} - \frac{\partial r}{\partial d}\right)\delta d\,dt + w^T(0)\,m\,\frac{\partial u_0}{\partial d}\,\delta d - \\ &\quad w^T\,m(d)\,\delta u\Big|_T - \int_{\mathcal{I}} \delta u^T\left(-m^T\,\dot{w} - \left(\frac{\partial r}{\partial u}\right)^T w - \left(\frac{\partial \pi}{\partial u}\right)^T\right)dt \end{aligned} \quad (218)$$

where we use integration by parts.

The second line in the final equality in (218) contains the implicitly defined response variation δu. To annihilate it we require the heretofore arbitrary w to solve the adjoint problem: Find w such that

$$\begin{aligned} -m^T\,\dot{w} &= \left(\frac{\partial r}{\partial u}\right)^T w + \left(\frac{\partial \pi}{\partial u}\right)^T \\ w(T) &= 0 \end{aligned} \quad (219)$$

Note that the adjoint problem is a *terminal value* problem, not an IVP! Because of this it is customary to introduce the change of variables and define the adjoint vector $\hat{w}$ such that

$$\hat{w}(t) = w(T-t) \tag{220}$$

In this way, we can express (219) as $\hat{w}$ such that

$$\begin{aligned} m^T \,\dot{\hat{w}} &= \left(\frac{\partial r}{\partial u}\right)^T \Bigg|_{T-t} \hat{w} + \left(\frac{\partial \pi}{\partial u}\right)^T \Bigg|_{T-t} \\ \hat{w}(0) &= 0 \end{aligned} \tag{221}$$

In the above, $\hat{w}$ is evaluated at time t while the tangent $\partial r/\partial u$ and adjoint load $\partial \pi/\partial u$ are evaluated at time $T-t$. This poses a challenge for the computation as we must complete the primal analysis to evaluate u over the interval $\mathcal{I}$ before beginning the adjoint analysis for $\hat{w}$. That said, once $\hat{w}$ is known, the sensitivity (211) reduces to

$$\delta\tilde{\theta}_i(d;\delta d) = \int_{\mathcal{I}} \frac{\partial \pi}{\partial d}\cdot \delta d\, dt - \int_{\mathcal{I}} \hat{w}^T \Bigg|_{T-t} \left(\frac{\partial m}{\partial d}\,\dot{u} - \frac{\partial r}{\partial d}\right)\delta d\, dt + \hat{w}^T(T)\, m\, \frac{\partial u_0}{\partial d}\,\delta d \tag{222}$$

where all quantities are evaluate at time t, unless specified otherwise.

20 History dependent problems

- Residual: Solve macro equilibrium equation for $\mathbf{u}^n$

$$r^n(\mathbf{u}^n) = \int_{\Omega} \nabla \mathbf{w}^n \cdot \mathbf{P}(\nabla \mathbf{u}^n, \mathbf{q}^n, d)\, dv - \ell^n(\mathbf{w}^n) = 0 \tag{223}$$

  For future reference we drop the arguments and write $\mathbf{P}^n = \mathbf{P}(\nabla \mathbf{u}^n, \mathbf{q}^n, d)$.

- State variables: Solve evolution equation for $\mathbf{q}^n$.

$$\mathbf{g}^n(\nabla \mathbf{u}^n, \mathbf{q}^n, \mathbf{q}^{n-1}) = 0 \tag{224}$$

- Consistent tangent

  Let's discuss the application of Newton's method to solve (232). Neglecting the design d for a moment, and discretizing (232) yields the residual equation

$$\mathbf{r}^n(\mathbf{U}^u, \mathbf{q}^n(\mathbf{U}^u, \mathbf{q}^{n-1})) = 0 \tag{225}$$

  In the above, $\mathbf{q}^n(\mathbf{U}^u, \mathbf{q}^{n-1})$ is an implicit relation obtained from (233) and $\mathbf{q}^{n-1}$ is fixed. To solve (225) we use Newton's method. First we initialize the variables by equating them to their previous values, i.e. $\mathbf{U}^n = \mathbf{U}^{n-1}$ and $\mathbf{q}^n = \mathbf{q}^{n-1}$. Now we start the Newton loop. While $\mathbf{r}^n(\mathbf{U}^u, \mathbf{q}^n(\mathbf{U}^u, \mathbf{q}^{n-1})) \neq 0$ we find the update $\Delta\mathbf{U}$ such that

1. $$\mathbf{r}^n(\mathbf{U}^u + \Delta\mathbf{U}, \mathbf{q}^n(\mathbf{U}^u + \Delta\mathbf{U}, \mathbf{q}^{n-1})) = 0$$

2. We estimate this equality via the expansion

$$\mathbf{r}^n(\mathbf{U}^u + \Delta\mathbf{U}, \mathbf{q}^n(\mathbf{U}^u + \Delta\mathbf{U}, \mathbf{q}^{n-1})) = 0 \approx \mathbf{r}^n + (\partial\mathbf{r}^n/\partial\mathbf{U}^n + \partial\mathbf{r}^n/\partial\mathbf{q}^n\,\partial\mathbf{q}^n/\partial\mathbf{U}^n)\,\Delta\mathbf{U} \tag{226}$$

   where the quantities on the righthand side are evaluated at $(\mathbf{U}^n, \mathbf{q}^n(\mathbf{U}^n, \mathbf{q}^{n-1}))$. The proper evaluation of $\partial\mathbf{q}^n/\partial\mathbf{U}^n$ is the magic behind the consistent tangent operator described below.

3. We solve $(\partial\mathbf{r}^n/\partial\mathbf{U}^n + \partial\mathbf{r}^n/\partial\mathbf{q}^n\,\partial\mathbf{q}^n/\partial\mathbf{U}^n)\,\Delta\mathbf{U} = -\mathbf{r}^n$ for $\Delta\mathbf{U}$

4. We update

$$\mathbf{U}^n \leftarrow \mathbf{U}^n + \Delta\mathbf{U}^n$$

   and

$$\mathbf{q}^n(\mathbf{U}^n, \mathbf{q}^{n-1})) \leftarrow \mathbf{q}^n(\mathbf{U}^n + \Delta\mathbf{U}^n, \mathbf{q}^{n-1}))$$

   Notable, if we do not update $\mathbf{q}^n(\mathbf{U}^n, \mathbf{q}^{n-1}))$ then the expansion of (226) is about the wrong value of $\mathbf{q}^n$, i.e. it will be about its initial value $\mathbf{q}^n = \mathbf{q}^{n-1}$ rather then the value dictated by the current iterate $\mathbf{U}^n$, i.e. $\mathbf{q}^n(\mathbf{U}^n, \mathbf{q}^{n-1}))$.

- Newton iteration
- QoI

$$\begin{aligned}
\theta &= \sum_{n=1}^{N} \int_\Omega \pi^n(\nabla\mathbf{u}^n, \mathbf{q}^n, d)\,dv \qquad (227)\\
&= \sum_{n=1}^{N} \left(\int_\Omega \pi^n(\nabla\mathbf{u}^n, \mathbf{q}^n, d)\,dv + \overbrace{\int_\Omega \nabla\mathbf{w}^n \cdot \mathbf{P}^n(\nabla\mathbf{u}^n, \mathbf{q}^n, d)\,dv - \ell^n(\mathbf{w}^n) + \int_\Omega \boldsymbol{\lambda}^n \cdot \mathbf{g}^n(\nabla\mathbf{u}^n, \mathbf{q}^n, \mathbf{q}^{n-1}, d)\,dv}^{=0} \right)
\end{aligned}$$

Sensitivity analysis, without loss of generality neglect $\partial\ell^n(\mathbf{w}^n)/\partial d$ and shape changes and assume the initial

Algorithm 1 Primal and adjoint material model

if Primal **then**
  Input: $\mathbf{H}^n$, $\mathbf{q}^n$, $\mathbf{q}^{n-1}$, d
  repeat
    Evaluate $\mathbf{g}^n$ and $\partial \mathbf{g}^n / \partial \mathbf{q}^n$
    Solve $\partial \mathbf{g}^n / \partial \mathbf{q}^n \, \Delta \mathbf{q}^n = -\mathbf{g}^n$ for $\Delta \mathbf{q}^n$
    Update $\mathbf{q}^n = \mathbf{q}^n + \Delta \mathbf{q}^n$
  until $\mathbf{g}^n \approx 0$
  Evaluate stress $\mathbf{P}^n = \mathbf{P}^n(\mathbf{H}^n, \mathbf{q}^n)$
  Evaluate $\partial \mathbf{g}^n / \partial \mathbf{H}^n$
  Solve $\partial \mathbf{g}^n / \partial \mathbf{q}^n \, D\mathbf{q}^n / D\mathbf{H}^n = -\partial \mathbf{g}^n / \partial \mathbf{H}^n$ for $D\mathbf{q}^n / D\mathbf{H}^n$
  Evaluate tangent $D\mathbf{P}^n / D\mathbf{H}^n = \partial \mathbf{P}^n / \partial \mathbf{H}^n + \partial \mathbf{P}^n / \partial \mathbf{q}^n \, D\mathbf{q}^n / D\mathbf{H}^n$
  Output: $\mathbf{P}^n$, $D\mathbf{P}^n / D\mathbf{H}^n$
else if Adjoint **then**
  Input: $\mathbf{H}^{n+1}$, $\mathbf{H}^n$, $\mathbf{q}^{n+1}$, $\mathbf{q}^n$, $\mathbf{q}^{n-1}$, d, $\partial \pi^n / \partial \mathbf{H}^n$, $\partial \pi^n / \partial \mathbf{q}^n$, $\boldsymbol{\lambda}^{n+1}$
  Evaluate $\partial \mathbf{g}^{n+1} / \partial \mathbf{q}^n$
  Evaluate $\partial \mathbf{g}^n / \partial \mathbf{H}^n$
  Solve $\partial \mathbf{g}^n / \partial \mathbf{q}^n \, D\mathbf{q}^n / D\mathbf{H}^n = -\partial \mathbf{g}^n / \partial \mathbf{H}^n$ for $D\mathbf{q}^n / D\mathbf{H}^n$
  Evaluate tangent $D\mathbf{P}^n / D\mathbf{H}^n = \partial \mathbf{P}^n / \partial \mathbf{H}^n + \partial \mathbf{P}^n / \partial \mathbf{q}^n \, D\mathbf{q}^n / D\mathbf{H}^n$
  Evaluate $\mathbf{P}_w^{init^n} = \left(\partial \pi^n / \partial \mathbf{H}^n + \left(D\mathbf{q}^n / D\mathbf{H}^n \right)^T \left(\partial \pi^n / \partial \mathbf{q}^n + \left(\partial \mathbf{g}^{n+1} / \partial \mathbf{q}^n \right)^T \boldsymbol{\lambda}^{n+1} \right) \right)$
  Output: $\left(D\mathbf{P}^n / D\mathbf{H}^n \right)^T$, $\mathbf{P}_w^{init^n}$
end if

Algorithm 2 Calculate displacement $\mathbf{U}^n$ and state $\mathbf{q}^n$

: Input $\mathbf{U}^{n-1}$, $\mathbf{q}^{n-1}$
: Initialize $\mathbf{U}^n = \mathbf{U}^{n-1}$ and $\mathbf{q} = \mathbf{q}^{n-1}$
repeat
  Evaluate residual vector $R_i^n = \int_\Omega \nabla \boldsymbol{\Psi}_i \cdot \mathbf{P}^n \, dv - F_i^n$, cf. Algorithm 1
  Evaluate tangent stiffness matrix $K_{ij} = \int_\Omega \nabla \boldsymbol{\Psi}_i \cdot \frac{D\mathbf{P}^n}{D\mathbf{H}^n} \nabla \boldsymbol{\Psi}_j \, dv$, cf. Algorithm 1
  Evaluate $\Delta \mathbf{U}^n$ from $\mathbf{K} \, \Delta \mathbf{U}^n = -\mathbf{R}^n$
  Update $\mathbf{U}^n = \mathbf{U}^n + \Delta \mathbf{U}^n$
until $\mathbf{R}^n \approx 0$

condition $D\mathbf{q}^0/Dd = 0$,

$$
\begin{aligned}
\frac{D\theta}{Dd} \quad = \quad & \sum_{n=1}^{N} \int_\Omega \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} \cdot \frac{D\mathbf{H}^n}{Dd} + \frac{\partial \pi^n}{\partial \mathbf{q}^n} \cdot \frac{D\mathbf{q}^n}{Dd} + \frac{\partial \pi^n}{\partial d} \right) dv \, + \\
& \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^n} \frac{D\mathbf{H}^n}{Dd} + \frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^n} \frac{D\mathbf{q}^n}{Dd} + \frac{\partial \mathbf{P}^n}{\partial d} \right) dv \, + \\
& \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \frac{D\mathbf{H}^n}{Dd} + \frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n} \frac{D\mathbf{q}^n}{Dd} + \frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^{n-1}} \frac{D\mathbf{q}^{n-1}}{Dd} + \frac{\partial \mathbf{g}^n}{\partial d} \right) dv \\
= \quad & \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d} \, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d} \, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d} \, dv \, + \\
& \sum_{n=1}^{N} \int_\Omega \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} \cdot \frac{D\mathbf{H}^n}{Dd} + \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^n} \frac{D\mathbf{H}^n}{Dd} + \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \frac{D\mathbf{H}^n}{Dd} \right) dv \, + \\
& \sum_{n=1}^{N} \int_\Omega \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} \cdot \frac{D\mathbf{q}^n}{Dd} + \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^n} \frac{D\mathbf{q}^n}{Dd} + \boldsymbol{\lambda}^n \cdot \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n} \frac{D\mathbf{q}^n}{Dd} + \frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^{n-1}} \frac{D\mathbf{q}^{n-1}}{Dd} \right) \right) dv \\
= \quad & \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d} \, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d} \, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d} \, dv \, + \\
& \sum_{n=1}^{N-1} \int_\Omega \frac{D\mathbf{H}^n}{Dd} \cdot \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^n} \right)^T \nabla \mathbf{w}^n + \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \right)^T \boldsymbol{\lambda}^n \right) dv \, + \\
& \sum_{n=1}^{N-1} \int_\Omega \frac{D\mathbf{q}^n}{Dd} \cdot \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^n} \right)^T \nabla \mathbf{w}^n + \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n} \right)^T \boldsymbol{\lambda}^n + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n} \right)^T \boldsymbol{\lambda}^{n+1} \right) dv \, + \\
& \int_\Omega \frac{D\mathbf{H}^N}{Dd} \cdot \left(\frac{\partial \pi^N}{\partial \mathbf{H}^N} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^N} \right)^T \nabla \mathbf{w}^N + \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{H}^N} \right)^T \boldsymbol{\lambda}^N \right) dv \, + \\
& \int_\Omega \frac{D\mathbf{q}^N}{Dd} \cdot \left(\frac{\partial \pi^N}{\partial \mathbf{q}^N} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^N} \right)^T \nabla \mathbf{w}^N + \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{q}^N} \right)^T \boldsymbol{\lambda}^N \right) dv
\end{aligned}
\tag{228}
$$

Annihilate the $D\mathbf{q}^n/Dd$ terms by requiring

$$
\boldsymbol{\lambda}^n = \begin{cases} -\left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{q}^N}\right)^{-T} \left(\frac{\partial \pi^N}{\partial \mathbf{q}^N} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^N} \right)^T \nabla \mathbf{w}^N \right) & \text{if } n = N \\ -\left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n}\right)^{-T} \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^n} \right)^T \nabla \mathbf{w}^n + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n} \right)^T \boldsymbol{\lambda}^{n+1} \right) & \text{if } n < N \end{cases}
\tag{229}
$$

Insert these $\boldsymbol{\lambda}^n$ into sensitivity expression

$$\begin{aligned}
\frac{D\theta}{Dd} &= \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d}\, dv + \\
&\sum_{n=1}^{N-1} \int_\Omega \frac{D\mathbf{H}^n}{Dd} \cdot \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^n} \right)^T \nabla \mathbf{w}^n + \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \right)^T \boldsymbol{\lambda}^n \right) dv + + \\
&\int_\Omega \frac{D\mathbf{H}^N}{Dd} \cdot \left(\frac{\partial \pi^N}{\partial \mathbf{H}^N} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^N} \right)^T \nabla \mathbf{w}^N + \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{H}^N} \right)^T \boldsymbol{\lambda}^N \right) dv \\
&= \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d}\, dv + \\
&\sum_{n=1}^{N-1} \int_\Omega \frac{D\mathbf{H}^n}{Dd} \cdot \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^n} - \frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^n} \left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n} \right)^{-1} \frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \right)^T [\nabla \mathbf{w}^n] - \right. \\
&\qquad \left. \left(\left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{q}^n} \right)^{-1} \frac{\partial \mathbf{g}^n}{\partial \mathbf{H}^n} \right)^T \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n} \right)^T [\boldsymbol{\lambda}^{n+1}] \right) \right) dv + \\
&\int_\Omega \frac{D\mathbf{H}^N}{Dd} \cdot \left(\frac{\partial \pi^N}{\partial \mathbf{H}^N} + \left(\frac{\partial \mathbf{P}^n}{\partial \mathbf{H}^N} - \frac{\partial \mathbf{P}^n}{\partial \mathbf{q}^N} \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{q}^N} \right)^{-1} \frac{\partial \mathbf{g}^N}{\partial \mathbf{H}^N} \right)^T [\nabla \mathbf{w}^N] - \left(\left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{q}^N} \right)^{-1} \frac{\partial \mathbf{g}^N}{\partial \mathbf{H}^N} \right)^T \frac{\partial \pi^N}{\partial \mathbf{q}^N} \right) dv \\
&= \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d}\, dv + \\
&\sum_{n=1}^{N-1} \int_\Omega \frac{D\mathbf{H}^n}{Dd} \cdot \left(\frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{D\mathbf{P}^n}{D\mathbf{H}^n} \right)^T [\nabla \mathbf{w}^n] + \left(\frac{D\mathbf{q}^n}{D\mathbf{H}^n} \right)^T \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n} \right)^T [\boldsymbol{\lambda}^{n+1}] \right) \right) dv + \\
&\int_\Omega \frac{D\mathbf{H}^N}{Dd} \cdot \left(\frac{\partial \pi^N}{\partial \mathbf{H}^N} + \left(\frac{D\mathbf{P}^n}{D\mathbf{H}^N} \right)^T [\nabla \mathbf{w}^N] + \left(\frac{D\mathbf{q}^N}{D\mathbf{H}^N} \right)^T \frac{\partial \pi^N}{\partial \mathbf{q}^N} \right) dv
\end{aligned}$$

In the above, $D\mathbf{P}^n/D\mathbf{H}^n$ and $D\mathbf{q}^n/D\mathbf{H}^n$ are obtained from Algorithm 1; i.e. by viewing $\mathbf{P}^n$ only as a function of $\mathbf{H}^n$ and $\mathbf{q}^n$ in which $\mathbf{q}^n$ is treated as an implicit function of only $\mathbf{H}^n$ via (233), thus making $\mathbf{P}^n$ and $\mathbf{q}^n$ functions of $\mathbf{H}^n$. Reemphasizing, in these $D\mathbf{P}^n/D\mathbf{H}^n$ and $D\mathbf{q}^n/D\mathbf{H}^n$ computations are for fixed $\mathbf{q}^{n-1}$ and d. To annihilate the implicit response variations $D\mathbf{H}^n/Dd$ for $n < N$ we require $\mathbf{w}^n$ to satisfy

$$\int_\Omega \nabla \frac{D\mathbf{u}}{Dd} \cdot \left(\left(\frac{D\mathbf{P}^n}{D\mathbf{H}^n} \right)^T [\nabla \mathbf{w}^n] + \frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{D\mathbf{q}^n}{D\mathbf{H}^n} \right)^T \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n} \right)^T [\boldsymbol{\lambda}^{n+1}] \right) \right) dv = 0 \tag{230}$$

for all admissible $\nabla D\mathbf{u}/Dd$. The strong form of this problem requires we find $\mathbf{w}$ such that

$$\begin{aligned}
\operatorname{div}\mathbf{P}_w^n &= \mathbf{0} && \text{in } \Omega \\
\mathbf{P}_w^n &= \left(\tfrac{D\mathbf{P}^n}{D\mathbf{H}^n}\right)^T [\nabla \mathbf{w}^n] + \overbrace{\frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{D\mathbf{q}^n}{D\mathbf{H}^n}\right)^T \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n}\right)^T [\boldsymbol{\lambda}^{n+1}]\right)}^{\mathbf{P}_w^{init^n}} && \text{in } \Omega \\
\mathbf{P}_w^n\,\mathbf{n} &= \mathbf{0} && \text{on } \partial\Omega_{Neumann} \\
\mathbf{w} &= \mathbf{0} && \text{on } \partial\Omega_{Dirichlet}
\end{aligned}$$

Note that the adjoint stress $\mathbf{P}_w^n$ is linear. It contains a "normal" $\mathbb{C}[\nabla \mathbf{u}]$ looking part $\left(\frac{D\mathbf{P}^n}{D\mathbf{H}^n}\right)^T [\nabla \mathbf{w}^n]$ and an "initial stress" part $\mathbf{P}_w^{init^n} = \frac{\partial \pi^n}{\partial \mathbf{H}^n} + \left(\frac{D\mathbf{q}^n}{D\mathbf{H}^n}\right)^T \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{q}^n}\right)^T [\boldsymbol{\lambda}^{n+1}]\right)$, cf. Algorithm 1. If $n = N$ we solve the above for $\mathbf{w}^N$, but we omit the terms containing $\boldsymbol{\lambda}^{N+1}$. Because of this, we first solve for $(\mathbf{w}^N, \boldsymbol{\lambda}^N)$ and then $(\mathbf{w}^{N-1}, \boldsymbol{\lambda}^{N-1})$ and so on until finally we solve for $(\mathbf{w}^1, \boldsymbol{\lambda}^1)$.

Upon solving for the adjoint response $(\mathbf{w}^n, \boldsymbol{\lambda}^n)$, the sensitivity reduces to

$$\frac{D\theta}{Dd} = \sum_{n=1}^{N} \int_\Omega \frac{\partial \pi^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \nabla \mathbf{w}^n \cdot \frac{\partial \mathbf{P}^n}{\partial d}\, dv + \sum_{n=1}^{N} \int_\Omega \boldsymbol{\lambda}^n \cdot \frac{\partial \mathbf{g}^n}{\partial d}\, dv \tag{231}$$

20.1 Brandon's way: take 2

As per Brandon's suggestion, we treat the state variables as functions of the deformation gradient and design, i.e. the $\mathbf{q}^n$ are no longer independent variables. Notably we evaluate derivatives of the state variables wrt. the design *for fixed deformation gradient* as in the direct differentiation (forward) method and we annihilate derivatives of the state variables wrt. the deformation gradient via the adjoint (backward) method. This works because the number of design parameters affecting a state variable is small, e.g. an element volume fraction or the element nodal coordinates. The advantage of the method is that we needn't modify the material constitutive routines for the adjoint problem as we do above.

- Residual: Solve macro equilibrium equation for $\mathbf{u}^n$

$$r^n(\mathbf{u}^n) = \int_\Omega \nabla \mathbf{w}^n \cdot \mathbf{P}(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)\, dv - \ell^n(\mathbf{w}^n) = 0 \tag{232}$$

  Again, for future reference we drop the arguments and write $\mathbf{P}^n = \mathbf{P}(\nabla \mathbf{u}^{n-1}, \mathbf{q}^n, d)$.

- State variables: Solve evolution equation $\mathbf{g}^n = \mathbf{0}$ for $\mathbf{q}^n$ for the given $(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)$.

$$\mathbf{g}^n(\mathbf{q}^n, (\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)) = \mathbf{0} \tag{233}$$

Algorithm 3 Material model

Input: $\mathbf{H}^n$, $\mathbf{q}^{n-1}$, d

Form vector of unknowns $\mathbf{z} = \left\{ \begin{array}{c} \mathbf{q}^n \\ \mathbf{P}^n \end{array} \right\}$

Define state variable and stress expressions $\mathbf{h} = \left\{ \begin{array}{c} \mathbf{g}^n \\ \mathbf{P}^n - \widehat{\mathbf{P}}^n(\mathbf{q}^n) \end{array} \right\}$

repeat

  Evaluate $\mathbf{h}$ and $\partial \mathbf{h}/\partial \mathbf{z}$

  Solve $\partial \mathbf{h}/\partial \mathbf{z}\, \Delta \mathbf{z} = -\mathbf{h}$ for $\Delta \mathbf{z}$

  Update $\mathbf{z} = \mathbf{z} + \Delta \mathbf{z}$

until $\mathbf{h} \approx 0$

Evaluate $\partial \mathbf{h}/\partial(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)$

Solve $\partial \mathbf{h}/\partial \mathbf{z}\, \partial \mathbf{z}/\partial(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d) = -\partial \mathbf{h}/\partial(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)$ for $\partial \mathbf{z}/\partial(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)$

Output: $\mathbf{z}$, $\partial \mathbf{z}/\partial(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)$

- Consistent tangent
- Newton iteration, cf. Algorithm 2, but replace all references to Algorithm 1 with Algorithm 3
- QoI

$$
\begin{aligned}
\theta &= \sum_{n=1}^{N} \int_\Omega \pi^n(\mathbf{u}^n, \nabla \mathbf{u}^n, \mathbf{q}^n, d)\, dv \qquad (234) \\
&= \sum_{n=1}^{N} \left(\int_\Omega \pi^n(\mathbf{u}^n, \nabla \mathbf{u}^n, \mathbf{q}^n, d)\, dv + \overbrace{\int_\Omega \nabla \mathbf{w}^n \cdot \mathbf{P}^n(\nabla \mathbf{u}^n, \mathbf{q}^{n-1}, d)\, dv - \ell^n(\mathbf{w}^n)}^{=0} \right)
\end{aligned}
$$

- Sensitivity analysis, without loss of generality neglect $\partial \ell^n(\mathbf{w}^n)/\partial d$ and shape changes and assume $D\mathbf{q}^0/Dd = 0$,

$$
\begin{aligned}
\frac{D\theta}{Dd} &= \sum_{n=1}^{N}\int_{\Omega}\left(\frac{\partial\pi^n}{\partial\mathbf{u}^n}\cdot\frac{D\mathbf{u}^n}{Dd}+\frac{\partial\pi^n}{\partial\mathbf{H}^n}\cdot\frac{D\mathbf{H}^n}{Dd}+\frac{\partial\pi^n}{\partial\mathbf{q}^n}\cdot\left(\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\frac{D\mathbf{H}^n}{Dd}+\frac{\partial\mathbf{q}^n}{\partial\mathbf{q}^{n-1}}\left(\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{H}^{n-1}}\frac{D\mathbf{H}^{n-1}}{Dd}+\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{q}^{n-2}}\left(\cdots\right)+\frac{\partial\mathbf{q}^{n-1}}{\partial d}\right)+\frac{\partial\mathbf{q}^n}{\partial d}\right)+\frac{\partial\pi^n}{\partial d}\right)dv \\
&\quad \sum_{n=1}^{N}\int_{\Omega}\nabla\mathbf{w}^n\cdot\left(\frac{\partial\mathbf{P}^n}{\partial\mathbf{H}^n}[\frac{D\mathbf{H}^n}{Dd}]+\frac{\partial\mathbf{P}^n}{\partial d}+\right. \\
&\quad \left.\frac{\partial\mathbf{P}^n}{\partial\mathbf{q}^{n-1}}\left(\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{H}^{n-1}}\frac{D\mathbf{H}^{n-1}}{Dd}+\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{q}^{n-2}}\left(\frac{\partial\mathbf{q}^{n-2}}{\partial\mathbf{H}^{n-2}}\frac{D\mathbf{H}^{n-2}}{Dd}+\frac{\partial\mathbf{q}^{n-2}}{\partial\mathbf{q}^{n-3}}\left(\cdots\right)+\frac{\partial\mathbf{q}^{n-2}}{\partial d}\right)+\frac{\partial\mathbf{q}^{n-1}}{\partial d}\right)\right)dv \\
&= \sum_{n=1}^{N}\int_{\Omega}\left(\frac{\partial\pi^n}{\partial d}+\nabla\mathbf{w}^n\cdot\left(\frac{\partial\mathbf{P}^n}{\partial d}+\frac{\partial\mathbf{P}^n}{\partial\mathbf{q}^{n-1}}\left(\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{q}^{n-2}}\left(\frac{\partial\mathbf{q}^{n-2}}{\partial\mathbf{q}^{n-3}}\left(\cdots\right)+\frac{\partial\mathbf{q}^{n-2}}{\partial d}\right)+\frac{\partial\mathbf{q}^{n-1}}{\partial d}\right)\right)\right. \\
&\quad \left.\frac{\partial\pi^n}{\partial\mathbf{q}^n}\cdot\left(\frac{\partial\mathbf{q}^n}{\partial d}+\frac{\partial\mathbf{q}^n}{\partial\mathbf{q}^{n-1}}\left(\frac{\partial\mathbf{q}^{n-1}}{\partial d}+\frac{\partial\mathbf{q}^{n-1}}{\partial\mathbf{q}^{n-2}}\left(\frac{\partial\mathbf{q}^{n-2}}{\partial d}+\frac{\partial\mathbf{q}^{n-2}}{\partial\mathbf{q}^{n-3}}\left(\cdots\right)\right)\right)\right)\right)dv+ \\
&\quad \sum_{n=1}^{N}\int_{\Omega}\left[\frac{D\mathbf{u}^n}{Dd}\cdot\frac{\partial\pi^n}{\partial\mathbf{u}^n}+\frac{D\mathbf{H}^n}{Dd}\cdot\left(\frac{\partial\pi^n}{\partial\mathbf{H}^n}+\left(\frac{\partial\mathbf{P}^n}{\partial\mathbf{H}^n}\right)^T[\nabla\mathbf{w}^n]+\right.\right. \\
&\quad \left(\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T\left[\frac{\partial\pi^n}{\partial\mathbf{q}^n}+\left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\left[\frac{\partial\pi^{n+1}}{\partial\mathbf{q}^{n+1}}+\left(\frac{\partial\mathbf{q}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T\left[\frac{\partial\pi^{n+2}}{\partial\mathbf{q}^{n+2}}+\left(\frac{\partial\mathbf{q}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T\left[\frac{\partial\pi^{n+3}}{\partial\mathbf{q}^{n+3}}+\left(\frac{\partial\mathbf{q}^{n+4}}{\partial\mathbf{q}^{n+3}}\right)^T\left[\cdots\right]\right]\right]\right]\right]+ \\
&\quad \left(\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T\left[\left(\frac{\partial\mathbf{P}^{n+1}}{\partial\mathbf{q}^n}\right)^T[\nabla\mathbf{w}_{n+1}]+\left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\left[\left(\frac{\partial\mathbf{P}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T[\nabla\mathbf{w}_{n+2}]+\right.\right. \\
&\quad \left.\left.\left.\left.\left(\frac{\partial\mathbf{q}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T\left[\left(\frac{\partial\mathbf{P}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T[\nabla\mathbf{w}_{n+3}]+\left(\frac{\partial\mathbf{q}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T\left[\cdots\right]\right]\right]\right]\right)\right]dv \\
&= \sum_{n=N}^{1}\int_{\Omega}\left(\frac{\partial\pi^n}{\partial d}+\nabla\mathbf{w}^n\cdot\left(\frac{\partial\mathbf{P}^n}{\partial d}+\mathbf{G}_w^n\frac{\partial\mathbf{q}^n}{\partial d}\right)+\left(\frac{\partial\pi^n}{\partial\mathbf{q}^n}+\mathbf{J}_w^n\right)\cdot\frac{\partial\mathbf{q}^n}{\partial d}\right)dv+ \\
&\quad \sum_{n=N}^{1}\int_{\Omega}\left[\frac{D\mathbf{u}^n}{Dd}\cdot\frac{\partial\pi^n}{\partial\mathbf{u}^n}+\frac{D\mathbf{H}^n}{Dd}\cdot\left(\frac{\partial\pi^n}{\partial\mathbf{H}^n}+\left(\frac{\partial\mathbf{P}^n}{\partial\mathbf{H}^n}+\frac{\partial\mathbf{P}^n}{\partial\mathbf{q}^n}\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T[\nabla\mathbf{w}^n]+\left(\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T\left(\mathbf{G}_w^n+\frac{\partial\pi^n}{\partial\mathbf{q}^n}+\mathbf{J}_w^n\right)\right)\right]dv
\end{aligned}
$$

For convenience we define

$$
\begin{aligned}
\mathbf{G}_w^n &= \left(\frac{\partial\mathbf{P}^{n+1}}{\partial\mathbf{q}^n}\right)^T[\nabla\mathbf{w}_{n+1}]+\left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\left[\left(\frac{\partial\mathbf{P}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T[\nabla\mathbf{w}_{n+2}]+\left(\frac{\partial\mathbf{q}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T\left[\left(\frac{\partial\mathbf{P}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T[\nabla\mathbf{w}_{n+3}]+\left(\frac{\partial\mathbf{q}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T\left[\cdots\right]\right]\right] \\
&= \left(\frac{\partial\mathbf{P}^{n+1}}{\partial\mathbf{q}^n}\right)^T[\nabla\mathbf{w}_{n+1}]+\left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\mathbf{G}_w^{n+1}
\end{aligned}
\tag{236}
$$

where $\mathbf{G}_w^N=0$. For convenience we also define

$$
\begin{aligned}
\mathbf{J}_w^n &= \left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\left[\frac{\partial\pi^{n+1}}{\partial\mathbf{q}^{n+1}}+\left(\frac{\partial\mathbf{q}^{n+2}}{\partial\mathbf{q}^{n+1}}\right)^T\left[\frac{\partial\pi^{n+2}}{\partial\mathbf{q}^{n+2}}+\left(\frac{\partial\mathbf{q}^{n+3}}{\partial\mathbf{q}^{n+2}}\right)^T\left[\cdots\right]\right]\right] \\
&= \left(\frac{\partial\mathbf{q}^{n+1}}{\partial\mathbf{q}^n}\right)^T\left(\frac{\partial\pi^{n+1}}{\partial\mathbf{q}^{n+1}}+\mathbf{J}_n^{n+1}\right)
\end{aligned}
\tag{237}
$$

where $\mathbf{J}_w^N=0$. Note the recursive nature of $\mathbf{G}_w^n$, $\mathbf{J}_w^n$ and $D\mathbf{q}^n/D\mathbf{d}$!

To annihilate the implicit response derivatives $D\mathbf{H}^n/Dd$, we solve n adjoint problems, i.e. we evaluate the $\mathbf{w}_n$ such that

$$
\int_{\Omega}\left[\frac{D\mathbf{u}^n}{Dd}\cdot\frac{\partial\pi^n}{\partial\mathbf{u}^n}+\frac{D\mathbf{H}^n}{Dd}\cdot\left(\frac{\partial\pi^n}{\partial\mathbf{H}^n}+\left(\frac{\partial\mathbf{P}^n}{\partial\mathbf{H}^n}+\frac{\partial\mathbf{P}^n}{\partial\mathbf{q}^n}\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T[\nabla\mathbf{w}^n]+\left(\frac{\partial\mathbf{q}^n}{\partial\mathbf{H}^n}\right)^T\left(\mathbf{G}_w^n+\frac{\partial\pi^n}{\partial\mathbf{q}^n}+\mathbf{J}_w^n\right)\right)\right]dv=0 \tag{238}
$$

for all $D\mathbf{H}^n/Dd$. In this way $D\theta/Dd$ reduces to

$$\frac{D\theta}{Dd} = \sum_{n=N}^{1} \int_{\Omega} \left(\frac{\partial \pi^n}{\partial d} + \nabla \mathbf{w}^n \cdot \left(\frac{\partial \mathbf{P}^n}{\partial d} + \mathbf{G}_w^n \frac{\partial \mathbf{q}^n}{\partial d} \right) + \left(\frac{\partial \pi^n}{\partial \mathbf{q}^n} + \mathbf{J}_w^n \right) \cdot \frac{\partial \mathbf{q}^n}{\partial d} \right) dv \tag{239}$$

Due to the history dependence of $\mathbf{q}$ and the recursive nature of $\mathbf{G}_w^n$ and $\mathbf{J}_w^n$ we must follow a strict ordering of operations.

1. First we solve forward in time, i.e. $n = 1, 2, \cdots N$ for $\mathbf{u}_n$ and $\mathbf{q}_n$. These quantities must be stored for later use. During this sequence we can also evaluate θ as per (234).
2. Second, we solve backward in time, i.e. $n = N, N-1, \cdots 1$ for the adjoint response $\mathbf{w}_n$. And after each $\mathbf{w}^n$ is evaluated we update the sensitivity $D\theta/Dd$ as per (239).

20.1.1 1-D plasticity

We follow we follow [3] Section 1.4.2, with a twist.

The equilibrium equation is solved over the domain $\Omega = [0, l]$ with $l = 1$

$$\begin{aligned} \sigma'(x) + b(x) &= 0 \\ u(0) &= 0 \\ \sigma(l) &= 0 \end{aligned} \tag{240}$$

The state variables $\mathbf{q} = (\epsilon^p, \alpha)$ are the plastic strain ϵ^p and accumulated plastic strain α which are related as

$$\alpha = \int_0^t |\dot{\epsilon}^p(s)| \, ds \tag{241}$$

The material model is

$$\sigma = E(u' - \epsilon^p) \tag{242}$$

and the yield function is

$$f_n = |\sigma_n| - \bar{\sigma}(\alpha_n) \tag{243}$$

Evolution equations

- if $f_n \leq 0$, i.e. elastic response

$$\begin{aligned} \epsilon^p_n &= \epsilon^p_{n-1} \\ \alpha_n &= \alpha_{n-1} \end{aligned} \tag{244}$$

- elseif $f_n > 0$, i.e. plastic response, solve for $\mathbf{z} = (\mathbf{q}_n, \sigma_n) = ((\epsilon_n^p, \alpha_n), \sigma_n)$ such that

$$\mathbf{h}(\mathbf{z}) = \left\{ \begin{array}{c} \mathbf{g}(\mathbf{q}_n) \\ \sigma_n - E\,(u_n' - \epsilon_n^p) \end{array} \right\} = \left\{ \begin{array}{c} |\sigma^T| - E\,(\alpha_n - \alpha_{n-1}) - \bar{\sigma}(\alpha_n) \\ \epsilon_n^p - (\epsilon_{n-1}^p + (\alpha_n - \alpha_{n-1})\,\mathrm{sign}(\sigma^T)) \\ \sigma_n - E\,(u_n' - \epsilon_n^p) \end{array} \right\} = \mathbf{0} \tag{245}$$

where

$$\sigma^T = E(u_n' - \epsilon_{n-1}^p) \tag{246}$$

is the trial stress and the function $\bar{\sigma}$ quantifies the hardening. We solve the above equation using Newton's method.

Note that $h_1 = 0$ can be solved for α_n independently of the $h_2 = 0$ and $h_3 = 0$ equations. Upon solving for α_n, the explicit equations $h_2 = 0$ and $h_3 = 0$ can be solved for ϵ_n^p and σ_n.

For definitiveness we use a Voce type law

$$\bar{\sigma} = \sigma_Y - (\sigma_s - \sigma_Y)\,(1 - \exp(-k\,\alpha)) \tag{247}$$

where σ_Y is the initial yield stress, σ_s is the saturation stress and k is a nonlinearity parameter, cf. Figure 20.

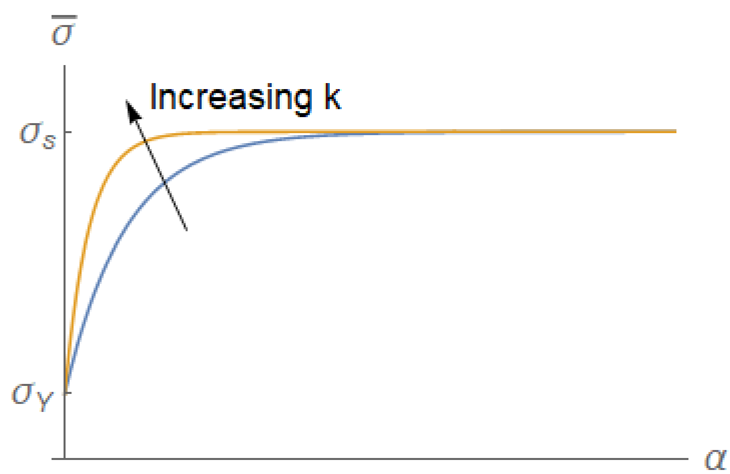


Figure 20: Hardening curve.

To see where the second of (245) comes from we discretize the associative flow rule

$$\dot{\epsilon}^p \;=\; \lambda \frac{\partial f}{\partial \sigma} = \lambda \frac{\sigma}{|\sigma|} = |\dot{\epsilon}^p|\,\mathrm{sign}\sigma = \dot{\alpha}\,\mathrm{sign}\sigma \tag{248}$$

to obtain

$$\epsilon_n^p - \epsilon_{n-1}^p \;=\; (\alpha_n - \alpha_{n-1})\,\mathrm{sign}\sigma_n \tag{249}$$

To finish this we must now see where the first of (245) comes from. Note that

$$\begin{aligned}
\sigma_n &= \sigma^T - E\,(\epsilon^p_n - \epsilon^p_{n-1}) \\
&= \sigma^T - E\,(\alpha_n - \alpha_{n-1})\,\mathrm{sign}\sigma_n \\
|\sigma_n|\,\mathrm{sign}\sigma_n &= |\sigma^T|\,\mathrm{sign}\sigma^T - E\,(\alpha_n - \alpha_{n-1})\,\mathrm{sign}\sigma_n
\end{aligned} \tag{250}$$

which implies

$$(|\sigma_n| + E\,(\alpha_n - \alpha_{n-1}))\,\mathrm{sign}\sigma_n = |\sigma^T|\,\mathrm{sign}\sigma^T \tag{251}$$

And since $E > 0$ and $(\alpha_n - \alpha_{n-1}) > 0$ we see that

$$\mathrm{sign}\sigma_n = \mathrm{sign}\sigma^T \tag{252}$$

Consequently

$$\begin{aligned}
f &= |\sigma_n| - \bar{\sigma}(\alpha_n) \\
&= |\sigma^T| - E\,(\alpha_n - \alpha_{n-1}) - \bar{\sigma}(\alpha_n)
\end{aligned} \tag{253}$$

Returning to the second of (245), we use (249) and (252) to obtain

$$\epsilon^p_n = \epsilon^p_{n-1} + (\alpha_n - \alpha_{n-1})\,\mathrm{sign}\sigma^T \tag{254}$$

Upon solving for $\mathbf{z}$ we solve $\partial\mathbf{h}/\partial\mathbf{z}\,\partial\mathbf{z}/\partial u'_n = -\partial\mathbf{h}/\partial u'_n$ for $\partial\mathbf{z}/\partial u'_n$, $\partial\mathbf{h}/\partial\mathbf{z}\,\partial\mathbf{z}/\partial\mathbf{q}_{n-1} = -\partial\mathbf{h}/\partial\mathbf{q}_{n-1}$ for $\partial\mathbf{z}/\partial\mathbf{q}_{n-1}$ and $\partial\mathbf{h}/\partial\mathbf{z}\,\partial\mathbf{z}/\partial\mathbf{d} = -\partial\mathbf{h}/\partial\mathbf{d}$ for $\partial\mathbf{z}/\partial\mathbf{d}$. In this way we can readily extract $\frac{D\sigma_n}{Du'_n}$, $\frac{D\sigma_n}{D\mathbf{q}_{n-1}}$ and $\frac{\partial\sigma_n}{\partial\mathbf{d}}$ as well as $\frac{D\mathbf{q}_n}{Du'_n}$, $\frac{D\mathbf{q}_n}{D\mathbf{q}_{n-1}}$ and $\frac{\partial\mathbf{q}_n}{\partial\mathbf{d}}$.

Again we note that the $h_1 = 0$ equation only involves α_n and hence we can readily evaluate, e.g. $\partial\alpha_n/\partial u'_n = -(\partial h_1/\partial\alpha_n)^{-1}\,\partial h_1/\partial u'_n$ Upon computing $\partial\alpha_n/\partial u'_n$ we can use the the explicit relations in h_2 and h_3 to evaluate $\partial\epsilon^p_n/\partial u'_n$ and $\partial\sigma_n/\partial u'_n$

For definitiveness we note that

$$
\begin{aligned}
\frac{\partial \mathbf{h}}{\partial \mathbf{z}} &= \left[\begin{array}{ccc} 0 & -E-\frac{\partial\bar\sigma}{\partial\alpha_n} & 0 \\ 1 & -\mathrm{sign}\sigma^T & 0 \\ E & 0 & 1 \end{array}\right] \\
\frac{\partial \mathbf{h}}{\partial u'_n} &= \left\{\begin{array}{c} E\,\mathrm{sign}\sigma^T \\ 0 \\ -E \end{array}\right\} \\
\frac{\partial \mathbf{h}}{\partial \mathbf{q}_{n-1}} &= \left[\begin{array}{cc} -E\,\mathrm{sign}\sigma^T & E \\ -1 & \mathrm{sign}\sigma^T \\ 0 & 0 \end{array}\right] \\
\frac{\partial \mathbf{h}}{\partial \mathbf{d}} &= \left[\begin{array}{cccc} (u'_n-\epsilon^p_{n-1})\,\mathrm{sign}\sigma^T-(\alpha_n-\alpha_{n-1}) & -\frac{\partial\bar\sigma}{\partial\sigma_Y} & -\frac{\partial\bar\sigma}{\partial\sigma_s} & -\frac{\partial\bar\sigma}{\partial k} \\ 0 & 0 & 0 & 0 \\ -(u_n-\epsilon^p_n) & 0 & 0 & 0 \end{array}\right]
\end{aligned}
\tag{255}
$$

where $\mathbf{d}=(E,\sigma_Y,\sigma_s,k)$ are the material parameters.

For exemplary purposes we wrote a Matlab code with the simple QoI choices

$$
\theta=\sum_{n=1}^{N}\int_\Omega \pi^n\,dx \tag{256}
$$

where $\pi^n=u$, $\pi^n=u'$, $\pi^n=\alpha^n$ or $\pi^n=E$. First, the code solves for u and evaluates θ, next the adjoint response w is evaluated and finally the sensitivities of θ wrt. the element parameters (E,σ_Y,σ_s,k) are computed.

20.2 Small deformation J_2 plasticity

For small deformation J_2 plasticity we follow [3].

The equilibrium equation is solved over the domain Ω

$$
\begin{aligned}
\mathrm{div}\boldsymbol{\sigma}+\mathbf{b} &= \mathbf{0} \quad \text{in } \Omega \\
\mathbf{u} &= \mathbf{0} \quad \text{on } A_u \\
\boldsymbol{\sigma}\,\mathbf{n} &= \mathbf{0} \quad \text{on } A_t
\end{aligned}
$$

The state variables $\mathbf{q}=(\boldsymbol{\epsilon}^p,\alpha)$ are the plastic strain $\boldsymbol{\epsilon}^p$ and accumulated plastic strain α which are related as

$$
\alpha=\int_0^t\sqrt{\frac{2}{3}}\,|\dot{\boldsymbol{\epsilon}}^p(s)|\,ds \tag{257}
$$

In this way, since $\dot{\boldsymbol{\epsilon}}^p$ is deviatoric (as seen below) for a uniaxial test we have $\dot{\boldsymbol{\epsilon}}^p=\mathrm{diag}\,(1,-1/2,-1/2)\;\dot\epsilon^p$ and $|\dot{\boldsymbol{\epsilon}}^p|=\sqrt{3/2}\,|\dot\epsilon^p|$, and hence (257) gives the 1-D result $\dot\alpha=|\dot\epsilon^p|$.

The isotropic material model is

$$\boldsymbol{\sigma} = \mathbb{C}[\nabla \mathbf{u} - \boldsymbol{\epsilon}^p] \tag{258}$$

where

$$\mathbb{C} = 2\,\mu\,\mathbb{I}_{dev} + \kappa\,\mathbb{I}_{sph} \tag{259}$$

μ and κ are the shear and bulk modulii and $\mathbb{I}_{dev} = \mathbb{I} - \frac{1}{3}\,\mathbf{I}\otimes\mathbf{I}$ and $\mathbb{I}_{sph} = \frac{1}{3}\,\mathbf{I}\otimes\mathbf{I}$ are the deviatoric and spherical 4-tensor projectors which are defined such that $\mathbb{I}_{dev}[\mathbf{A}] = \mathbf{A} - \frac{1}{3}\,\mathrm{tr}\mathbf{A}\,\mathbf{I}$ and $\mathbb{I}_{sph}[\mathbf{A}] = \frac{1}{3}\,\mathrm{tr}\mathbf{A}\,\mathbf{I}$ for all 2-tensors $\mathbf{A}$.

The yield surface

$$f = |\boldsymbol{\sigma}_{dev}| - \sqrt{\frac{2}{3}}\,\bar{\sigma}(\alpha) \tag{260}$$

As for α, for the uniaxial test we have $\boldsymbol{\sigma}_{dev} = \mathrm{diag}\,(2/3, -1/3, -1/3)\ \sigma$ and $|\boldsymbol{\sigma}_{dev}| = \sqrt{2/3}\,|\sigma|$, and hence (260) is consistent with the 1-D case $f = |\sigma| - \bar{\sigma}(\alpha)$.

Evolution equations

- if $f_n \leq 0$, i.e. elastic response

$$\begin{aligned} \boldsymbol{\epsilon}^p_n &= \boldsymbol{\epsilon}^p_{n-1} \\ \alpha_n &= \alpha_{n-1} \end{aligned} \tag{261}$$

- elseif $f_n > 0$, i.e. plastic response, solve for $\mathbf{q}_n = (\boldsymbol{\epsilon}^p_n, \alpha_n)$ such that

$$\mathbf{g} = \left\{ \begin{array}{c} |\boldsymbol{\sigma}^T_{dev}| - 2\sqrt{\frac{3}{2}}\ \mu\,(\alpha_n - \alpha_{n-1})\,\mathbf{N}^T - \sqrt{\frac{2}{3}}\,\bar{\sigma}(\alpha_n) \\ \boldsymbol{\epsilon}^p_n - (\boldsymbol{\epsilon}^p_{n-1} + \sqrt{\frac{3}{2}}\,(\alpha_n - \alpha_{n-1})\,\mathbf{N}^T) \end{array} \right\} = \mathbf{0} \tag{262}$$

where

$$\boldsymbol{\sigma}^T = \mathbb{C}(\nabla \mathbf{u}_n - \boldsymbol{\epsilon}^p_{n-1}) \tag{263}$$

is the trial stress and

$$\mathbf{N}^T = \frac{\boldsymbol{\sigma}^T_{dev}}{|\boldsymbol{\sigma}^T_{dev}|} \tag{264}$$

Note that $g_1 = 0$ can be solved for α_n independently of $g_2 = 0$. Upon solving for α_n, the linear equation $g_2 = 0$ can be solved for $\boldsymbol{\epsilon}^p_n$.

To see where the second of (262) comes from we discretize the associative flow rule

$$\dot{\boldsymbol{\epsilon}}^p = \lambda \frac{\partial f}{\partial \sigma} = \lambda \frac{\boldsymbol{\sigma}_{dev}}{|\boldsymbol{\sigma}_{dev}|} = |\dot{\boldsymbol{\epsilon}}^p|\,\mathbf{N} = \sqrt{\frac{3}{2}}\,\dot{\alpha}\,\mathbf{N} \tag{265}$$

to obtain

$$\boldsymbol{\epsilon}^p_n - \boldsymbol{\epsilon}^p_{n-1} = \sqrt{\frac{3}{2}}\,(\alpha_n - \alpha_{n-1})\,\mathbf{N} \tag{266}$$

To finish this we must now see where the first of (262) comes from. Note that

$$\begin{aligned}
\boldsymbol{\sigma}_{dev_n} &= \boldsymbol{\sigma}^T_{dev} - \left(\mathbb{C}[\boldsymbol{\epsilon}^p_n - \boldsymbol{\epsilon}^p_{n-1}]\right)_{dev} \\
&= \boldsymbol{\sigma}^T_{dev} - \mathbb{C}[\sqrt{\frac{3}{2}}\,(\alpha_n - \alpha_{n-1})\,\mathbf{N}] \\
|\boldsymbol{\sigma}_{dev_n}|\,\mathbf{N} &= |\boldsymbol{\sigma}^T_{dev}|\,\mathbf{N}^T - 2\sqrt{\frac{3}{2}}\,\mu(\alpha_n - \alpha_{n-1})\,\mathbf{N}
\end{aligned} \tag{267}$$

which implies

$$(|\boldsymbol{\sigma}_{dev_n}| + 2\sqrt{\frac{3}{2}}\,\mu\,(\alpha_n - \alpha_{n-1}))\,\mathbf{N} = |\boldsymbol{\sigma}^T_{dev}|\,\mathbf{N}^T \tag{268}$$

And since $\mu > 0$ and $(\alpha_n - \alpha_{n-1}) > 0$ we see that

$$\mathbf{N} = \mathbf{N}^T \tag{269}$$

Consequently

$$\begin{aligned}
f &= |\boldsymbol{\sigma}_{dev_n}| - \sqrt{\frac{2}{3}}\,\bar{\sigma}(\alpha_n) \\
&= |\boldsymbol{\sigma}^T_{dev}| - 2\sqrt{\frac{3}{2}}\,\mu\,(\alpha_n - \alpha_{n-1}) - \sqrt{\frac{2}{3}}\,\bar{\sigma}(\alpha_n)
\end{aligned} \tag{270}$$

Returning to the second of (262), we use (266) and (269) to obtain

$$\boldsymbol{\epsilon}^p_n = \boldsymbol{\epsilon}^p_{n-1} + \sqrt{\frac{3}{2}}\,(\alpha_n - \alpha_{n-1})\,\mathbf{N}^T \tag{271}$$

The J_2 version of 1 is presented in Algorithm 4.

Algorithm 4 J_2 Primal material model

Input: $\mathbf{H}^n$, $\mathbf{q}^n$, $\mathbf{q}^{n-1}$, d
Evaluate $\boldsymbol{\sigma}^T = \mathbb{C}[\nabla \mathbf{u}_n - \boldsymbol{\epsilon}^p_{n-1}]$
Evaluate $\mathbf{N}^T = \boldsymbol{\sigma}^T_{dev}$
repeat
  Evaluate g^n_1 and $\partial g^n_1 / \partial \alpha_n$
  Solve $\partial g^n_1 / \partial \alpha_n \, \Delta\alpha_n = -g^n_1$ for $\Delta\alpha_n$
  Update $\alpha_n = \alpha_n + \Delta\alpha_n$
until $g^n_1 \approx 0$
Evaluate $\boldsymbol{\epsilon}^p_n = \boldsymbol{\epsilon}^p_{n-1} + \sqrt{\frac{3}{2}}\,(\alpha_n - \alpha_{n-1})\ \mathbf{N}^T$
Evaluate stress $\boldsymbol{\sigma} = \mathbb{C}[\nabla \mathbf{u}_n - \boldsymbol{\epsilon}^p_n]$
Evaluate $\partial g^n_1 / \partial \mathbf{H}^n$
Solve $\partial g^n_1 / \partial \alpha_n \, D\alpha_n / D\mathbf{H}^n = -\partial g^n_1 / \partial \mathbf{H}^n$ for $D\alpha_n / D\mathbf{H}^n$
Evaluate $D\boldsymbol{\epsilon}^p_n / D\mathbf{H}_n = \partial \boldsymbol{\epsilon}^p_n / \partial \mathbf{H}_n + \partial \boldsymbol{\epsilon}^p_n / \partial \alpha_n \, D\alpha_n / D\mathbf{H}_n = \sqrt{\frac{3}{2}}\, D\alpha_n / D\mathbf{H}^n$
Evaluate tangent $D\boldsymbol{\sigma} / D\mathbf{H}^n = \partial \boldsymbol{\sigma} / \partial \mathbf{H}^n + \partial \boldsymbol{\sigma} / \partial \boldsymbol{\epsilon}^p_n \, D\boldsymbol{\epsilon}^p_n / D\mathbf{H}^n = \mathbb{C}[\mathbf{I} - D\boldsymbol{\epsilon}^p_n / D\mathbf{H}^n]$
Output: $\boldsymbol{\sigma}$, $D\boldsymbol{\sigma} / D\mathbf{H}^n$

21 9, Mandel, Voigt and other notations

21.1 The 9 notation

I like using the 9x9 notation in which

$$\begin{aligned}
\boldsymbol{\epsilon}_9 &= \left\{ \begin{array}{c} \epsilon_{11} \\ \epsilon_{21} \\ \epsilon_{31} \\ \epsilon_{12} \\ \epsilon_{22} \\ \epsilon_{32} \\ \epsilon_{13} \\ \epsilon_{23} \\ \epsilon_{33} \end{array} \right\} \\
\boldsymbol{\sigma}_9 &= \left\{ \begin{array}{c} \sigma_{11} \\ \sigma_{21} \\ \sigma_{31} \\ \sigma_{12} \\ \sigma_{22} \\ \sigma_{32} \\ \sigma_{13} \\ \sigma_{23} \\ \sigma_{33} \end{array} \right\}
\end{aligned} \tag{272}$$

$$
\mathbb{C}_9 \quad = \quad \begin{bmatrix}
C_{1111} & C_{1121} & C_{1131} & C_{1121} & C_{1122} & C_{1132} & C_{1131} & C_{1132} & C_{1133} \\
C_{1121} & C_{2121} & C_{2131} & C_{2121} & C_{2122} & C_{2132} & C_{2131} & C_{2132} & C_{2133} \\
C_{1131} & C_{2131} & C_{3131} & C_{2131} & C_{3122} & C_{3132} & C_{3131} & C_{3132} & C_{3133} \\
C_{1121} & C_{2121} & C_{2131} & C_{2121} & C_{2122} & C_{2132} & C_{2131} & C_{2132} & C_{2133} \\
C_{1122} & C_{2122} & C_{3122} & C_{2122} & C_{2222} & C_{2232} & C_{3122} & C_{2232} & C_{2233} \\
C_{1132} & C_{2132} & C_{3132} & C_{2132} & C_{2232} & C_{3232} & C_{3132} & C_{3232} & C_{3233} \\
C_{1131} & C_{2131} & C_{3131} & C_{2131} & C_{3122} & C_{3132} & C_{3131} & C_{3132} & C_{3133} \\
C_{1132} & C_{2132} & C_{3132} & C_{2132} & C_{2232} & C_{3232} & C_{3132} & C_{3232} & C_{3233} \\
C_{1133} & C_{2133} & C_{3133} & C_{2133} & C_{2233} & C_{3233} & C_{3133} & C_{3233} & C_{3333}
\end{bmatrix} \tag{273}
$$

1. This notation preserves inner products, i.e. $\boldsymbol{\epsilon} \cdot \boldsymbol{\sigma} = \boldsymbol{\epsilon}_9^T \, \boldsymbol{\sigma}_9$

2. This notation makes rotations easy to implement. Indeed, if I rotate my material particle by $\mathbf{R}$, then the rotated elasticity tensor $\mathbb{C}_9'$ is given by

$$
\mathbb{C}_9' = (\mathbf{R} \odot \mathbf{R}) \, \mathbb{C}_9 \, (\mathbf{R} \odot \mathbf{R})^T = \mathbb{R}_9 \, \mathbf{C}_9 \, \mathbb{R}_9^T
$$

where

$$
\mathbb{R}_9 = (\mathbf{R} \odot \mathbf{R})
$$

is orthogonal and $\odot$ is the Kronecker product.

3. For

$$
\mathbf{R} = \begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)(-\cos(\theta)) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}
$$

we have

$$
\mathbb{R}_9 = \begin{pmatrix}
C_\theta^2 & -C_\alpha S_\theta C_\theta & S_\alpha S_\theta C_\theta & -C_\alpha S_\theta C_\theta & C_\alpha^2 \sin^2(\theta) & S_\alpha(-C_\alpha)\sin^2(\theta) & S_\alpha S_\theta C_\theta & S_\alpha(-C_\alpha)\sin^2(\theta) & S_\alpha^2 \sin^2(\theta) \\
S_\theta C_\theta & C_\alpha C_\theta^2 & S_\alpha\left(-C_\theta^2\right) & -C_\alpha \sin^2(\theta) & -C_\alpha^2 S_\theta C_\theta & S_\alpha C_\alpha S_\theta C_\theta & S_\alpha \sin^2(\theta) & S_\alpha C_\alpha S_\theta C_\theta & S_\alpha^2 S_\theta(-C_\theta) \\
0 & S_\alpha C_\theta & C_\alpha C_\theta & 0 & S_\alpha(-C_\alpha)S_\theta & -C_\alpha^2 S_\theta & 0 & S_\alpha^2 S_\theta & S_\alpha C_\alpha S_\theta \\
S_\theta C_\theta & -C_\alpha \sin^2(\theta) & S_\alpha \sin^2(\theta) & C_\alpha C_\theta^2 & -C_\alpha^2 S_\theta C_\theta & S_\alpha C_\alpha S_\theta C_\theta & S_\alpha\left(-C_\theta^2\right) & S_\alpha C_\alpha S_\theta C_\theta & S_\alpha^2 S_\theta(-C_\theta) \\
\sin^2(\theta) & C_\alpha S_\theta C_\theta & S_\alpha S_\theta(-C_\theta) & C_\alpha S_\theta C_\theta & C_\alpha^2 C_\theta^2 & S_\alpha(-C_\alpha)C_\theta^2 & S_\alpha S_\theta(-C_\theta) & S_\alpha(-C_\alpha)C_\theta^2 & S_\alpha^2 C_\theta^2 \\
0 & S_\alpha S_\theta & C_\alpha S_\theta & 0 & S_\alpha C_\alpha C_\theta & C_\alpha^2 C_\theta & 0 & S_\alpha^2(-C_\theta) & S_\alpha(-C_\alpha)C_\theta \\
0 & 0 & 0 & S_\alpha C_\theta & S_\alpha(-C_\alpha)S_\theta & S_\alpha^2 S_\theta & C_\alpha C_\theta & -C_\alpha^2 S_\theta & S_\alpha C_\alpha S_\theta \\
0 & 0 & 0 & S_\alpha S_\theta & S_\alpha C_\alpha C_\theta & S_\alpha^2(-C_\theta) & C_\alpha S_\theta & C_\alpha^2 C_\theta & S_\alpha(-C_\alpha)C_\theta \\
0 & 0 & 0 & 0 & S_\alpha^2 & S_\alpha C_\alpha & 0 & S_\alpha C_\alpha & C_\alpha^2
\end{pmatrix} \tag{274}
$$

21.2 Mandel notation

My next favorite is the Mandel notation in which

$$
\begin{aligned}
\boldsymbol{\epsilon}_m &= \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ \sqrt{2}\,\epsilon_{23} \\ \sqrt{2}\,\epsilon_{13} \\ \sqrt{2}\,\epsilon_{12} \end{Bmatrix} \\
\boldsymbol{\sigma}_m &= \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sqrt{2}\,\sigma_{23} \\ \sqrt{2}\,\sigma_{13} \\ \sqrt{2}\,\sigma_{12} \end{Bmatrix} \\
\mathbb{C}_m &= \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & \sqrt{2}C_{1132} & \sqrt{2}C_{1131} & \sqrt{2}C_{1121} \\ C_{1122} & C_{2222} & C_{2233} & \sqrt{2}C_{2232} & \sqrt{2}C_{3122} & \sqrt{2}C_{2122} \\ C_{1133} & C_{2233} & C_{3333} & \sqrt{2}C_{3233} & \sqrt{2}C_{3133} & \sqrt{2}C_{2133} \\ \sqrt{2}C_{1132} & \sqrt{2}C_{2232} & \sqrt{2}C_{3233} & 2C_{3232} & 2C_{3132} & 2C_{2132} \\ \sqrt{2}C_{1131} & \sqrt{2}C_{3122} & \sqrt{2}C_{3133} & 2C_{3132} & 2C_{3131} & 2C_{2131} \\ \sqrt{2}C_{1121} & \sqrt{2}C_{2122} & \sqrt{2}C_{2133} & 2C_{2132} & 2C_{2131} & 2C_{2121} \end{bmatrix}
\end{aligned} \tag{275}
$$

To see where $\mathbb{C}_m$ comes from we define the transformation matrix

$$
\mathbb{T}_{9m} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{276}
$$

such that

$$
\begin{aligned}
\boldsymbol{\epsilon}_m &= \mathbb{T}_{9m}\,\boldsymbol{\epsilon}_9 \\
\boldsymbol{\epsilon}_9 &= \mathbb{T}_{9m}^T\,\boldsymbol{\epsilon}_m \\
\boldsymbol{\sigma}_m &= \mathbb{T}_{9m}\,\boldsymbol{\sigma}_9 \\
\boldsymbol{\sigma}_9 &= \mathbb{T}_{9m}^T\,\boldsymbol{\sigma}_m \\
\mathbb{T}_{9m}\,\mathbb{T}_{9m}^T &= \mathbf{I}_6 \\
\mathbb{T}_{9m}^T\,\mathbb{T}_{9m} &= \mathbb{S}_9 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{277}
$$

where $\mathbb{S}_9$ is the 9 notation symmetrizer, i.e. the identity on the space of symmetric 2-tensors such that e.g. $\mathbb{S}_9\,\boldsymbol{\epsilon}_9 = \boldsymbol{\epsilon}_9$. So $\mathbb{T}_{9m}$ is "orthogonal like." Because of this we have

$$
\boldsymbol{\sigma}_m = \mathbb{T}_{9m}\,\boldsymbol{\sigma}_9 = \mathbb{T}_{9m}\,\mathbb{C}_9\,\boldsymbol{\epsilon}_9 = \mathbb{T}_{9m}\,\mathbb{C}_9\,\mathbb{T}_{9m}^T\boldsymbol{\epsilon}_m \tag{278}
$$

and hence

$$
\begin{aligned}
\mathbb{C}_m &= \mathbb{T}_{9m}\,\mathbb{C}_9\,\mathbb{T}_{9m}^T \\
\mathbb{C}_9 &= \mathbb{T}_{9m}^T\,\mathbb{C}_m\,\mathbb{T}_{9m}
\end{aligned}
\tag{279}
$$

Now we need the rotated $\mathbb{C}'_m$. To obtain this we again use the transformation $\mathbb{T}_{9m}$, i.e.

$$
\mathbb{C}'_m = \mathbb{T}_{9m}\,\mathbb{C}'_9\,\mathbb{T}_{9m}^T = \mathbb{T}_{9m}\,\mathbb{R}_9\,\mathbb{C}_9\,\mathbb{R}_9^T\,\mathbb{T}_{9m}^T = \mathbb{T}_{9m}\,\mathbb{R}_9\,\mathbb{T}_{9m}^T\,\mathbb{C}_m\,\mathbb{T}_{9m}\,\mathbb{R}_9^T\,\mathbb{T}_{9m}^T = \mathbb{R}_m\,\mathbb{C}_m\,\mathbb{R}_m^T \tag{280}
$$

where

$$
\mathbb{R}_m = \mathbb{T}_{9m}\,\mathbb{R}_9\,\mathbb{T}_{9m}^T
$$

is orthogonal.

1. This notation preserves inner products, i.e. $\boldsymbol{\epsilon}\cdot\boldsymbol{\sigma} = \boldsymbol{\epsilon}_m^T\,\boldsymbol{\sigma}_m$
2. This notation makes rotations "easy" to obtain as seen above and preserves the orthogonality of the rotation $\mathbb{R}_m$.

3. For

$$\mathbf{R} = \begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)(-\cos(\theta)) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

we have

$$\mathbb{R}_m = \begin{pmatrix} \cos^2(\theta) & \cos^2(\alpha)\sin^2(\theta) & \sin^2(\alpha)\sin^2(\theta) & -\frac{\sin(2\alpha)\sin^2(\theta)}{\sqrt{2}} & \frac{\sin(\alpha)\sin(2\theta)}{\sqrt{2}} & -\frac{\cos(\alpha)\sin(2\theta)}{\sqrt{2}} \\ \sin^2(\theta) & \cos^2(\alpha)\cos^2(\theta) & \sin^2(\alpha)\cos^2(\theta) & -\frac{\sin(2\alpha)\cos^2(\theta)}{\sqrt{2}} & -\frac{\sin(\alpha)\sin(2\theta)}{\sqrt{2}} & \frac{\cos(\alpha)\sin(2\theta)}{\sqrt{2}} \\ 0 & \sin^2(\alpha) & \cos^2(\alpha) & \frac{\sin(2\alpha)}{\sqrt{2}} & 0 & 0 \\ 0 & \frac{\sin(2\alpha)\cos(\theta)}{\sqrt{2}} & -\frac{\sin(2\alpha)\cos(\theta)}{\sqrt{2}} & \cos(2\alpha)\cos(\theta) & \cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ 0 & -\frac{\sin(2\alpha)\sin(\theta)}{\sqrt{2}} & \frac{\sin(2\alpha)\sin(\theta)}{\sqrt{2}} & -\cos(2\alpha)\sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)\cos(\theta) \\ \frac{\sin(2\theta)}{\sqrt{2}} & -\frac{\cos^2(\alpha)\sin(2\theta)}{\sqrt{2}} & -\frac{\sin^2(\alpha)\sin(2\theta)}{\sqrt{2}} & \frac{\sin(2\alpha)\sin(2\theta)}{2} & \sin(\alpha)(-\cos(2\theta)) & \cos(\alpha)\cos(2\theta) \end{pmatrix} \tag{281}$$

21.3 Voigt notation

I'm not a fan of Voigt notation, but it is very popular.

$$\begin{aligned} \boldsymbol{\epsilon}_v &= \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\,\epsilon_{23} \\ 2\,\epsilon_{13} \\ 2\,\epsilon_{12} \end{Bmatrix} \\ \boldsymbol{\sigma}_v &= \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{Bmatrix} \\ \mathbb{C}_v &= \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1132} & C_{1131} & C_{1121} \\ C_{1122} & C_{2222} & C_{2233} & C_{2232} & C_{3122} & C_{2122} \\ C_{1133} & C_{2233} & C_{3333} & C_{3233} & C_{3133} & C_{2133} \\ C_{1132} & C_{2232} & C_{3233} & C_{3232} & C_{3132} & C_{2132} \\ C_{1131} & C_{3122} & C_{3133} & C_{3132} & C_{3131} & C_{2131} \\ C_{1121} & C_{2122} & C_{2133} & C_{2132} & C_{2131} & C_{2121} \end{bmatrix} \end{aligned} \tag{282}$$

To see where $\mathbb{C}_m$ comes from we define the transformation matrix

$$\mathbb{T}_v = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \tag{283}$$

such that

$$\begin{aligned} \boldsymbol{\epsilon}_v &= \mathbb{T}_v^{-1}\,\boldsymbol{\epsilon}_m \\ \boldsymbol{\epsilon}_m &= \mathbb{T}_v\,\boldsymbol{\epsilon}_v \\ \boldsymbol{\sigma}_v &= \mathbb{T}_v\,\boldsymbol{\sigma}_m \\ \boldsymbol{\sigma}_m &= \mathbb{T}_v^{-1}\,\boldsymbol{\sigma}_v \end{aligned} \tag{284}$$

Note the lack of "symmetry" here. Further note that $\mathbb{T}_v$ is *not* orthogonal. Nonetheless, the transformation allows us to obtain

$$\boldsymbol{\sigma}_v = \mathbb{T}_v\,\boldsymbol{\sigma}_m = \mathbb{T}_v\,\mathbb{C}_m\,\boldsymbol{\epsilon}_m = \mathbb{T}_v\,\mathbb{C}_m\,\mathbb{T}_v\,\boldsymbol{\epsilon}_v \tag{285}$$

and hence

$$\begin{aligned} \mathbb{C}_v &= \mathbb{T}_v\,\mathbb{C}_m\,\mathbb{T}_v \\ \mathbb{C}_m &= \mathbb{T}_v^{-1}\,\mathbb{C}_v\,\mathbb{T}_v^{-1} \end{aligned} \tag{286}$$

Now we need the rotated $\mathbb{C}'_v$. To obtain this we again use the transformation $\mathbb{T}_v$, i.e.

$$\mathbb{C}'_v = \mathbb{T}_v\,\mathbb{C}'_m\,\mathbb{T}_v = \mathbb{T}_v\,\mathbb{R}_m\,\mathbb{C}_m\,\mathbb{R}_m^T\,\mathbb{T}_v = \mathbb{T}_v\,\mathbb{R}_m\,\mathbb{T}_v^{-1}\,\mathbb{C}_v\,\mathbb{T}_v^{-1}\,\mathbb{R}_m^T\,\mathbb{T}_v = \mathbb{R}_v\,\mathbb{C}_v\,\mathbb{R}_v^T \tag{287}$$

where

$$\mathbb{R}_v = \mathbb{T}_v\,\mathbb{R}_m\,\mathbb{T}_v^{-1}$$

To obtain the above we use the facts that $\mathbb{T}_v$ and $\mathbb{T}_v^{-1}$ are diagonal. Note that $\mathbb{R}_v$ is *not* orthogonal.

1. This notation preserves inner products, i.e. $\boldsymbol{\epsilon} \cdot \boldsymbol{\sigma} = \boldsymbol{\epsilon}_v^T\,\boldsymbol{\sigma}_v$
2. This notation makes rotations "easy" to implement as seen above, but $\mathbb{R}_v$ is not orthogonal .
3. For

$$\mathbf{R} = \begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)(-\cos(\theta)) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

we have

$$\mathbb{R}_v = \begin{bmatrix} \cos^2(\theta) & \cos^2(\alpha)\sin^2(\theta) & \sin^2(\alpha)\sin^2(\theta) & -\sin(2\alpha)\sin^2(\theta) & \sin(\alpha)\sin(2\theta) & -\cos(\alpha)\sin(2\theta) \\ \sin^2(\theta) & \cos^2(\alpha)\cos^2(\theta) & \sin^2(\alpha)\cos^2(\theta) & \sin(2\alpha)\left(-\cos^2(\theta)\right) & -\sin(\alpha)\sin(2\theta) & \cos(\alpha)\sin(2\theta) \\ 0 & \sin^2(\alpha) & \cos^2(\alpha) & \sin(2\alpha) & 0 & 0 \\ 0 & \frac{1}{2}\sin(2\alpha)\cos(\theta) & -\frac{1}{2}\sin(2\alpha)\cos(\theta) & \cos(2\alpha)\cos(\theta) & \cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ 0 & -\frac{1}{2}\sin(2\alpha)\sin(\theta) & \frac{1}{2}\sin(2\alpha)\sin(\theta) & -\cos(2\alpha)\sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)\cos(\theta) \\ \frac{1}{2}\sin(2\theta) & -\frac{1}{2}\cos^2(\alpha)\sin(2\theta) & -\frac{1}{2}\sin^2(\alpha)\sin(2\theta) & \frac{\sin(2\alpha)\sin(2\theta)}{2} & \sin(\alpha)(-\cos(2\theta)) & \cos(\alpha)\cos(2\theta) \end{bmatrix} \quad (288)$$

21.4 Bastard Voigt notation

I'm not really not a fan of adhoc Voigt notations, but they appear so I will include them.

$$\begin{aligned} \boldsymbol{\epsilon}_b &= \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ \epsilon_{12} \\ \epsilon_{23} \\ \epsilon_{13} \end{Bmatrix} \\ \boldsymbol{\sigma}_b &= \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{13} \end{Bmatrix} \\ \mathbb{C}_b &= \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & 2C_{1121} & 2C_{1132} & 2C_{1131} \\ C_{1122} & C_{2222} & C_{2233} & 2C_{2122} & 2C_{2232} & 2C_{3122} \\ C_{1133} & C_{2233} & C_{3333} & 2C_{2133} & 2C_{3233} & 2C_{3133} \\ C_{1121} & C_{2122} & C_{2133} & 2C_{2121} & 2C_{2132} & 2C_{2131} \\ C_{1132} & C_{2232} & C_{3233} & 2C_{2132} & 2C_{3232} & 2C_{3132} \\ C_{1131} & C_{3122} & C_{3133} & 2C_{2131} & 2C_{3132} & 2C_{3131} \end{bmatrix} \end{aligned} \quad (289)$$

Note that $\mathbb{C}_b$ is not symmetric!

To see where $\mathbb{C}_m$ comes from we define the transformation matrix

$$\mathbb{T}_b = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \quad (290)$$

such that

$$
\begin{aligned}
\boldsymbol{\epsilon}_b &= \mathbb{T}_b\,\boldsymbol{\epsilon}_m \\
\boldsymbol{\epsilon}_m &= \mathbb{T}_b^{-1}\,\boldsymbol{\epsilon}_b \\
\boldsymbol{\sigma}_b &= \mathbb{T}_b\,\boldsymbol{\sigma}_m \\
\boldsymbol{\sigma}_m &= \mathbb{T}_b^{-1}\,\boldsymbol{\sigma}_b
\end{aligned} \tag{291}
$$

Note that $\mathbb{T}_b$ is *not* orthogonal. Nonetheless, the transformation allows us to obtain

$$
\boldsymbol{\sigma}_b = \mathbb{T}_b\,\boldsymbol{\sigma}_m = \mathbb{T}_b\,\mathbb{C}_m\,\boldsymbol{\epsilon}_m = \mathbb{T}_b\,\mathbb{C}_m\,\mathbb{T}_b^{-1}\,\boldsymbol{\epsilon}_b \tag{292}
$$

and hence

$$
\begin{aligned}
\mathbb{C}_b &= \mathbb{T}_b\,\mathbb{C}_m\,\mathbb{T}_b^{-1} \\
\mathbb{C}_m &= \mathbb{T}_b^{-1}\,\mathbb{C}_b\,\mathbb{T}_b
\end{aligned} \tag{293}
$$

Now we need the rotated $\mathbb{C}'_b$. To obtain this we again use the transformation $\mathbb{T}_b$, i.e.

$$
\mathbb{C}'_b = \mathbb{T}_b\,\mathbb{C}'_m\,\mathbb{T}_b^{-1} = \mathbb{T}_b\,\mathbb{R}_m\,\mathbb{C}_m\,\mathbb{R}_m^T\,\mathbb{T}_b^{-1} = \mathbb{T}_b\,\mathbb{R}_m\,\mathbb{T}_b^{-1}\,\mathbb{C}_b\,\mathbb{T}_b\,\mathbb{R}_m^T\,\mathbb{T}_b^{-1} = \mathbb{R}_b\,\mathbb{C}_b\,\mathbb{R}_b^{-1} \tag{294}
$$

where

$$
\mathbb{R}_b = \mathbb{T}_b\,\mathbb{R}_m\,\mathbb{T}_b^{-1} \tag{295}
$$

is *not* orthogonal. However, since $\mathbb{T}_b$ is diagonal and $\mathbb{R}_m$ is orthogonal, $\mathbb{R}_b^{-1} = \mathbb{T}_b\,\mathbb{R}_m^T\,\mathbb{T}_b^{-1}$ is easily computed.

1. This notation *does not* inner products, i.e. $\boldsymbol{\epsilon}\cdot\boldsymbol{\sigma} \neq \boldsymbol{\epsilon}_b^T\,\boldsymbol{\sigma}_b$. ***This must have ramifications in FEA as the virtual work*** $\delta\boldsymbol{\epsilon}\cdot\boldsymbol{\sigma} \neq \delta\boldsymbol{\epsilon}_b^T\,\boldsymbol{\sigma}_b$***.***

2. This notation makes rotations "easy" to implement as seen above, but it is very clunky.

3. For
$$
\mathbf{R} = \begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & \sin(\alpha)(-\cos(\theta)) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}
$$
we have
$$
\mathbb{R}_b = \begin{bmatrix} \cos^2(\theta) & \cos^2(\alpha)\sin^2(\theta) & \sin^2(\alpha)\sin^2(\theta) & -\cos(\alpha)\sin(2\theta) & -\sin(2\alpha)\sin^2(\theta) & \sin(\alpha)\sin(2\theta) \\ \sin^2(\theta) & \cos^2(\alpha)\cos^2(\theta) & \sin^2(\alpha)\cos^2(\theta) & \cos(\alpha)\sin(2\theta) & \sin(2\alpha)\left(-\cos^2(\theta)\right) & -\sin(\alpha)\sin(2\theta) \\ 0 & \sin^2(\alpha) & \cos^2(\alpha) & 0 & \sin(2\alpha) & 0 \\ \frac{1}{2}\sin(2\theta) & -\frac{1}{2}\cos^2(\alpha)\sin(2\theta) & -\frac{1}{2}\sin^2(\alpha)\sin(2\theta) & \cos(\alpha)\cos(2\theta) & \frac{\sin(2\alpha)\sin(2\theta)}{2} & \sin(\alpha)(-\cos(2\theta)) \\ 0 & \frac{1}{2}\sin(2\alpha)\cos(\theta) & -\frac{1}{2}\sin(2\alpha)\cos(\theta) & \sin(\alpha)\sin(\theta) & \cos(2\alpha)\cos(\theta) & \cos(\alpha)\sin(\theta) \\ 0 & -\frac{1}{2}\sin(2\alpha)\sin(\theta) & \frac{1}{2}\sin(2\alpha)\sin(\theta) & \sin(\alpha)\cos(\theta) & -\cos(2\alpha)\sin(\theta) & \cos(\alpha)\cos(\theta) \end{bmatrix} \tag{296}
$$

References

[1] Schwedes T, Ham DA, Funke SW, Piggott MD (2017) Mesh dependence in PDE-constrained optimisation: an application in tidal turbine array layouts. Springer Publishing Company, Incorporated

[2] Antil H., Leykekhman D. (2018) A Brief Introduction to PDE-Constrained Optimization. In: Antil H., Kouri D.P., Lacasse MD., Ridzal D. (eds) Frontiers in PDE-Constrained Optimization. The IMA Volumes in Mathematics and its Applications, vol 163. Springer, New York, NY

[3] J.C. SIMO and R.L. TAYLOR, CONSISTENT TANGENT OPERATORS FOR RATE-INDEPENDENT ELASTOPLASTICITY, COMPUTER METHODS IN APPLIED MECHANICS AND ENGINEERING 48 (1985) 101-118