Escola Superior d'Enginyeries Industrials, Aeroespacial i Audiovisual de Terrassa
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Master's Degree in Aerospace Engineering

Master's Final Thesis

CLASS SESSIONS

Marc Núñez
Albert Torres
Oriol Trujillo

November 16, 2017

Contents

List of Figures

1 Session 3: The SLERP Method

Albert Torres Rubio
11/10/2017

The SLERP algorithm is used in Topology Optimization when a level-set function describes the topology of the domain. It consist in using the Topological Derivative to minimize a function $(J(\chi))$ in the whole domain providing a topology which has optimal stiffness for a certain value of volume. In the next paragraphs this method is explained.

Firstly, the Topological Derivative has to be defined as the difference between the actual state of the topology and any infinitesimal change in it. Equation (1) represents the definition of this derivative.

$$J\left(\chi+\tilde{\chi}_{\epsilon}\right)-J(\chi)=D_T J(\chi)\,|Be| \tag{1}$$

Where,

- $\tilde{\chi}_{\epsilon}$ is any infinitesimal topology change.
- $J(\chi)$ is the actual topology.
- D_T is the Topological Derivative.
- Be is the ball inserted in the domain

The aim of the whole process is to minimize the topology function, therefore, the optimal condition is expressed as Equation 2 states.

$$J\left(\chi+\tilde{\chi}_{\epsilon}\right)-J(\chi)=D_T J(\chi)\,|Be|\geq 0 \qquad \forall\,\tilde{\chi} \tag{2}$$

Following reference work a g function is defined from the Topological Derivative as:

$$g=\begin{cases} D_T J & \Psi>0 \quad (X\in\Omega^-) \\ -D_T J & \Psi<0 \quad (X\in\Omega^+) \end{cases} \tag{3}$$

Notice that Ψ is the level-set function that defines the topology. As a result, the new definition of the optimal condition becomes:

$$g(\chi)\begin{cases} >0 & \Psi>0 \quad (\chi\in\Omega^-) \\ <0 & \Psi<0 \quad (\chi\in\Omega^+) \end{cases} \tag{4}$$

Alternative as:

$$sign(g(\chi(\Psi)))=sign(\Psi) \tag{5}$$

Thus, in order to find the optimal solution only the signs must coincide between g and Ψ. Moreover, in order to solve the problem, a strong optimality condition is imposed through the following requirements.

1. To satisfy the same sign of the gradient and the level-set, it is imposed to be the same.

2. Transform the optimal condition represented in Equation (5) as the following expression:
$$R(\Psi) = g(\chi(\Psi^*)) - \Psi^* = 0 \tag{6}$$
By this new definition, the problem can be solved by a Fix Point method, performing the iteration: $\Psi_{n+1} = g(\chi(\Psi_n))$.

3. Reduce level-set possible infinite values by making unitary its $L^2(\Omega)$ norm:
$$||\psi||_L^2(\Omega) = 1 \quad \rightarrow \quad (\Psi, \Psi) = \int_\Omega \Psi^2 = 1 \tag{7}$$
Where $(\cdot, \cdot)$ is the scalar product.

Having introduced these changes, we can rewrite Equation (6) in its general case as:

$$R(\Psi) = \frac{g(\chi(\Psi))}{||g(\chi(\Psi))||} - \Psi = 0 \tag{8}$$

It must be commented that when the optimal solution is reached using Equation (8), $||g(\chi(\Psi))||$ becomes equal to 1. Therefore, Equation (6), which is the original optimal condition remains invariant. Trying to solve the problem with this configuration with a Fix-point method would lead to:

$$\Psi_{n+1} = \frac{g(\chi(\Psi_n))}{||g(\chi(\Psi_n))||} \tag{9}$$

The above equation would be too aggressive to solve directly in some cases. Thus, a relaxation the algorithm (9) is proposed through a line-search method. The values α and β are computed by explicitly impose $||\psi|| = 1$, leading to the final expression of the SLERP algorithm (10).

$$\Psi_{n+1} = \beta\Psi_n + \alpha\frac{g(\chi(\Psi_n))}{||g(\chi(\Psi_n))||} \tag{10}$$

Where $\theta_n = \arccos\left(\frac{(\dot{\Psi}_n, g_n)}{||\Psi_n||||g_n||}\right)$ and plays the role of the line search parameter.

2 Session 5: Compliance gradient computation

Albert Torres Rubio
31/10/2017

2.1 Problem Definition

In order to solve a topological optimization problem the gradient of the cost function must be computed, performing this calculation in order to minimize the compliance (as an example), the following steps must be executed to find out which is the gradient of the function. Firstly, defining:

$$l(v) = \int_{\Gamma_N} t \cdot v$$

$$a(\rho, u, v) = \int_{\Omega} J'u : C(\rho) : \nabla^s u$$

In fact, the initial problem we have is the following:

$$\min \; l(u)$$

subject to

$$\left.\begin{array}{c} \int_\Omega \rho \leq V \\ a(\rho, u, v) = l(v) \end{array}\right\} \quad \forall v \in \gamma \quad and \quad \forall \rho \in \chi$$

Where,

$$\gamma = \{\phi \in H^1(\rho) : \phi \mid_{\Gamma_0} = 0\}$$
$$\chi = L^\infty(\Omega, \{0, 1\})$$

2.2 Differential definition

Performing the calculation of the differential of the cost function:

$$l(u(\rho + \tilde{\rho})) - l(u(\rho)) = l[u(\rho + \tilde{\rho}) - u(\rho)] =$$

$$l[D_\rho u(\rho)\tilde{\rho} + O(\rho^2)] = l[D_\rho u(\rho)\tilde{\rho}] + O(\rho^2)$$

Therefore,

$$Dl(\rho)\tilde{\rho} = l[D_\rho u(\rho)\tilde{\rho}] \tag{11}$$

Where D_ρ is the derivative respect to ρ.

2.3 Gradient Computation

Defining,

$$F(\rho) = a(\rho, u(\rho), v) - l(v) = 0 \qquad \forall v, \, \forall \rho$$

Therefore, computing the differential:

$$\underbrace{F(\rho+\tilde{\rho})}_{=0} - \underbrace{F(\rho)}_{=0} = a(\rho+\tilde{\rho}, u(\rho+\tilde{\rho}), v) - a(\rho, u(\rho), v) =$$
$$a(\rho+\tilde{\rho}, u(\rho+\tilde{\rho}), v) - a(\rho+\tilde{\rho}, u(\rho), v) + a(\rho+\tilde{\rho}, u(\rho), v) - a(\rho, u(\rho), v) =$$
$$a(\rho+\tilde{\rho}, Du(\rho)\tilde{\rho}, v) + D_\rho a(\rho, u(\rho), v)\tilde{\rho} + O(\tilde{\rho}^2) =$$
$$a(\rho, Du(\rho)\tilde{\rho}, v) + D_\rho a(\rho, u(\rho), v)\tilde{\rho} + O(\tilde{\rho}^2) = 0$$

Which lasts in

$$a(\rho, Du(\rho)\tilde{\rho}, v) = \boxed{-D_\rho a(\rho, u(\rho), v)\tilde{\rho}} + O(\tilde{\rho}^2) \qquad \forall v \in \gamma, \forall \rho \in \mathbb{R} \tag{12}$$

Continuing with Equation 11:

$$l(u(\rho+\tilde{\rho})) - l(u(\rho)) = l[D_\rho u(\rho)\tilde{\rho}] + O(\tilde{\rho}^2) = a(\rho, u(\rho), D_\rho u(\rho)\tilde{\rho}) + O(\tilde{\rho}^2) =$$
$$a(\rho, D_\rho u(\rho)\tilde{\rho}, u(\rho)) + O(\tilde{\rho}^2) = -D_\rho a(\rho, u(\rho)\tilde{\rho}, u(\rho)) + O(\tilde{\rho}^2) =$$
$$-[a(\rho+\tilde{\rho}, u(\rho), v) - a(\rho, u(\rho), v)] + O(\tilde{\rho}^2) =$$
$$-\int_\Omega \nabla^s u : [C(\rho+\tilde{\rho}) - C(\rho)] : \nabla^s u + O(\tilde{\rho}^2) =$$
$$-\int_\Omega \nabla^s u : C'(\rho)\tilde{\rho} : \nabla^s u + O(\tilde{\rho}^2) =$$
$$= Dl(\rho)\tilde{\rho} + O(\rho^2)$$

Therefore,

$$Dl(\rho)\tilde{\rho} = \int_\Omega \underbrace{(\nabla^s u : C'(\rho) : \nabla^s u)}_{\text{g}} \tilde{\rho} = (g, \tilde{\rho})_{L^2}$$

Finally, the computed gradient is:

$$\boxed{g = \nabla^s u : C'(\rho) : \nabla^s u)} \tag{13}$$

Where,

$$C(\rho) = 2\mu(\rho)I + [\kappa(\rho) - \mu(\rho)]I \otimes I$$