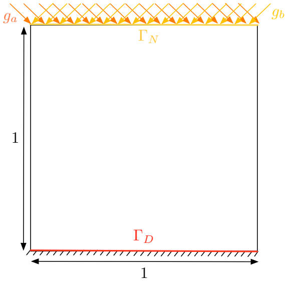
# *Numerical methods for shape optimization problems under uncertainties*


Marc DAMBRINE (Pau)

Charles DAPOGNY (Grenoble) and Helmut HARBRECHT (Basel)


VII Partial differential equations, optimal design and numerics, Benasque 08/23/2017.

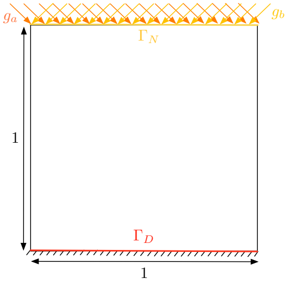
# *The model case of a bridge*




The displacement solves the linear elasticity system

$$\begin{cases} -\mathrm{div}(Ae(u)) = 0 & \text{in } D, \\ u = 0 & \text{on } \Gamma_D, \\ Ae(u)n = g & \text{on } \Gamma_N, \\ Ae(u)n = 0 & \text{on } \Gamma, \end{cases}$$

# *The model case of a bridge*




The displacement solves the linear elasticity system

$$\begin{cases} -\mathrm{div}(Ae(u)) = 0 & \text{in } D, \\ u = 0 & \text{on } \Gamma_D, \\ Ae(u)n = g & \text{on } \Gamma_N, \\ Ae(u)n = 0 & \text{on } \Gamma, \end{cases}$$

The rigidy of the bridge is measured by its compliance aka the work of applied loadings

$$F(D,g) = \int_{\Gamma_N} g u_D = \int_D Ae(u_D) : e(u_D).$$

# *Objectif*

Starting from a parametrized shape problem

$$(D, \omega) \mapsto F(D, \omega), \quad \forall D \in \mathcal{A}, \omega \in \Omega$$

we consider the average objective

$$\mathbb{E}[F](D) = \int_{\Omega} F(D, \omega) dP(\omega).$$

or a weighted combination of moments

$$\mathbb{E}[F](D) + \alpha \mathbb{V}ar[F](D).$$

# *Objectif*

Starting from a parametrized shape problem

$$(D, \omega) \mapsto F(D, \omega), \quad \forall D \in \mathcal{A}, \omega \in \Omega$$

we consider the average objective

$$\mathbb{E}[F](D) = \int_\Omega F(D, \omega) dP(\omega).$$

or a weighted combination of moments

$$\mathbb{E}[F](D) + \alpha \mathbb{V}ar[F](D).$$

*Objective*

Given a partial statistical description of the random loading, design an efficient algorithm to minimize the expectation of the objective

# *The main difficulty : the curse of dimension.*

- The space of events $\Omega$ -in our example, the space of loadings- can be enormous. It is a vector space of infinite dimension !!!
- So the question is how to compute first

$$\int_{L^2(\Gamma_N)} g(\omega)u_D(\omega)dP(\omega)$$

then its gradient withs respect to $D$ ...
- Natural idea : use Galerkin approximation
- But nevertheless integral on high dimensional domains are to be computed there are no deterministic appropriate quadrature methods to perform so and one is forced to use Monte-Carlo method ...

# *Our result*

Consider a special class of problems:

Minimize the expectation of a quadratic shape functional for the state function which is defined by a state equation with a random right-hand side.

# *Our result*

Consider a special class of problems:

Minimize the expectation of a quadratic shape functional for the state function which is defined by a state equation with a random right-hand side.

Then,

- all quantities for performing a gradient-based shape optimization algorithm can be expressed deterministically .
- only the random parameter's first and second moment are needed.

# *Our result*

Consider a special class of problems:

Minimize the expectation of a quadratic shape functional for the state function which is defined by a state equation with a random right-hand side.

Then,

- all quantities for performing a gradient-based shape optimization algorithm can be expressed deterministically .
- only the random parameter's first and second moment are needed.

Consequence:

- a fully deterministic algorithm
- same cost as for classical shape optimization when no uncertainties are taken into account.

# *The idea (1/2)*

Consider a finite dimensional example: $h$ design variable

- the state $u(h,\omega)$ solve the linear system $A(h)u(h,\omega) = f(\omega)$
- the original cost $C$ is quadratic

$$C(h,\omega) = Bu(h,\omega) \cdot u(h,\omega) = B : u(h,\omega) \otimes u(h,\omega).$$

# *The idea (1/2)*

Consider a finite dimensional example: $h$ design variable

- the state $u(h,\omega)$ solve the linear system $A(h)u(h,\omega) = f(\omega)$
- the original cost $C$ is quadratic

$$C(h,\omega) = Bu(h,\omega)\cdot u(h,\omega) = B : u(h,\omega)\otimes u(h,\omega).$$

- the averaged cost is now

$$\mathbb{E}[C](h) = B : \mathbb{E}[u(h,.)\otimes u(h,.)] = B : \mathbb{C}or[u](h)$$

where the correlation matrix

$$\mathbb{C}or[u](h)_{i,j} = \int_\Omega u(h,\omega)_i u(h,\omega)_j dP(\omega)$$

solves the bigger linear system

$$(A(h)\otimes A(h))\mathbb{C}or(u)(h) = \mathbb{C}or(f).$$

# *The idea (2/2)*

Now we derive w.r. the design variable $h$. We get the deterministic expression

$$D_h\mathbb{E}[h].\hat{h} = (D_h A(h).\hat{h} \otimes Id)\mathbb{C}or[u,p](h)$$

where

- the adjoint state $p(h,\omega)$ solves

$$A(h)^T p(h,\omega) = -2B^T u(h,\omega),$$

- the correlation matrix $\mathbb{C}or[u,p](h)$ solves

$$(A(h) \otimes A(h)^T)\mathbb{C}or[u,p](h) = -(A(h) \otimes B)\mathbb{C}or[u](h).$$

# *On the example*

The correlation of the displacement solves the equations:

$$\begin{cases} (\mathrm{div}_x \otimes \mathrm{div}_y)(Ae_x \otimes Ae_y)\,\mathrm{Cor}(u) = 0 & \textit{in } D \times D, \\ \mathrm{Cor}(u) = 0 & \textit{on } \Gamma_D \times \Gamma_D, \\ (\mathrm{div}_x \otimes I_y)(Ae_x \otimes I_y)\,\mathrm{Cor}(u) = 0 & \textit{on } D \times \Gamma_D, \\ (I_x \otimes \mathrm{div}_y)(I_x \otimes Ae_y)\,\mathrm{Cor}(u) = 0 & \textit{on } \Gamma_D \times D, \\ (Ae_x \otimes Ae_y)\,\mathrm{Cor}(u)(n_x \otimes n_y) = \mathrm{Cor}(g) & \textit{on } \Gamma_N \times \Gamma_N, \\ (\mathrm{div}_x \otimes I_y)(Ae_x \otimes Ae_y)\,\mathrm{Cor}(u)(I_x \otimes n_y) = 0 & \textit{on } D \times (\Gamma_N \cup \Gamma), \\ (I_x \otimes \mathrm{div}_y)(Ae_x \otimes Ae_y)\,\mathrm{Cor}(u)(n_x \otimes I_y) = 0 & \textit{on } (\Gamma_N \cup \Gamma_N) \times D, \\ (Ae_x \otimes Ae_y)\,\mathrm{Cor}(u)(n_x \otimes n_y) = 0 & \textit{on } \big((\Gamma_N \cup \Gamma) \times (\Gamma_N \cup \Gamma)\big) \setminus (\Gamma_N \times \Gamma_N), \\ (Ae_x \otimes I_y)\,\mathrm{Cor}(u)(n_x \otimes I_y) = 0 & \textit{on } (\Gamma_N \times \Gamma) \times \Gamma_D, \\ (I_x \otimes Ae_y)\,\mathrm{Cor}(u)(I_x \otimes n_y) = 0 & \textit{on } \Gamma_D \times (\Gamma_N \times \Gamma). \end{cases}$$

Not so easy to solves on the numerical point of view ...

# *Well prepared RHS*

If

$$\mathbb{C}or[f] = \sum_i f_i \otimes f_i$$

Then

$$\mathbb{C}or[u](h) = \sum_i u_i(h) \otimes u_i(h) \text{ and } \mathbb{C}or[u,p](h) = \sum_i u_i(h) \otimes p_i(h)$$

where $u_i(h)$ and $p_i(h)$ solve

$$A(h)u_i(h) = f_i \text{ and } A^T(h)p_i(h) = -Bu_i.$$

# *Well prepared RHS*

If

$$\mathbb{C}or[f] = \sum_i f_i \otimes f_i$$

Then

$$\mathbb{C}or[u](h) = \sum_i u_i(h) \otimes u_i(h) \text{ and } \mathbb{C}or[u,p](h) = \sum_i u_i(h) \otimes p_i(h)$$

where $u_i(h)$ and $p_i(h)$ solve

$$A(h)u_i(h) = f_i \text{ and } A^T(h)p_i(h) = -Bu_i.$$

Idea: approximate $\mathbb{C}or[f]$ by a low rank approximation of the type

$$\mathbb{C}or[f] \approx \sum_{i=N} f_i \otimes f_i$$

(use incomplete Choleski decomposition for example or SVD, PGD...)

# *Numerical example*

Fix two loads:

$$g_a = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \text{ and } g_b = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

The applied loadings are

$$g(x,\omega) = \xi_1(\omega) g_a + \xi_2(\omega) g_b$$

where the random variables $\xi_1, \xi_2$ are centered and normalized and correlated:

$$\alpha = \int_\Omega \xi_1(\omega)\xi_2(\omega) dP(\omega).$$

# *Numerical example*

Fix two loads:

$$g_a = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \text{ and } g_b = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

The applied loadings are

$$g(x,\omega) = \xi_1(\omega) g_a + \xi_2(\omega) g_b$$

where the random variables $\xi_1, \xi_2$ are centered and normalized and correlated:

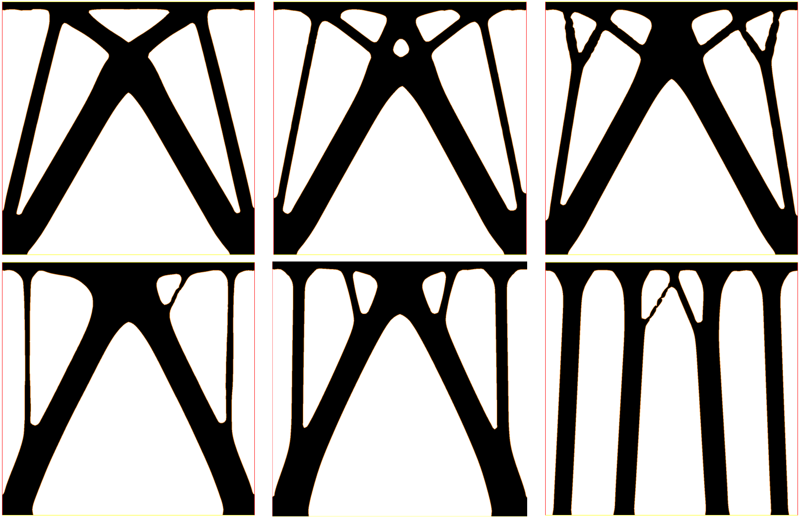
$$\alpha = \int_\Omega \xi_1(\omega)\xi_2(\omega) dP(\omega).$$

Here

$$\mathbb{C}or[g] = g_a \otimes g_a + g_b \otimes g_b + \alpha(g_a \otimes g_b + g_b \otimes g_a)$$

and no approximation is needed

# *Numerical example*



$\alpha = -1, -0.7, 0, 0.5, 0.8, 1$ (from left to right, top to bottom).