Metamodels and Sensitivity Analysis for Uncertainty Estimation and Robust Design in Aerodynamics

R. Duvigneau

INRIA Sophia-Antipolis Méditerranée
OPALE Project-Team

Seminar Stochastic and Noisy Simulators, May 17th 2011

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Classical point of view for CFD-based design

CFD-based drag estimation for
a deterministic system:

- Imposed geometry
- Imposed boundary condition
- Imposed fluid modeling

Statistical point of view for CFD-based design

CFD-based drag estimation for a deterministic system:

- Imposed geometry
- Imposed boundary condition
- Imposed fluid modeling

CFD-based drag estimation for a *stochastic* system:

- Geometrical uncertainty:
  - Manufacturing tolerance
  - Structural deformation
- Boundary condition uncertainty:
  - Velocity fluctuations
  - Incidence fluctuation
- Modeling uncertainty:
  - Weather fluctuation
  - Turbulence modeling

Methodology for statistical estimation

Hypothesis

- Uncertain parameters can be characterized by probability density functions (PDFs)

Method

- Uncertainty propagation through the CFD code
- Obtain the PDF of a functional / field
- Compute the statistics for the functional / field (mean, variance)

Approaches for functionals

- Use of functional derivatives with respect to uncertain parameters
- Use of surrogate models to represent the functional fluctuation

Approaches for fields

- Use of field derivatives with respect to uncertain parameter

Methodology for statistical estimation based on derivatives

Use of derivatives (Method of Moments)

- Compute the derivatives of the functional / field values w.r.t. uncertain parameters for the nominal conditions
- Describe the functional / field value with a Taylor series expansion around the nominal conditions

$$j(\alpha) = j(\mu_\alpha) + \sum_i G_i \delta\alpha_i + \frac{1}{2} \sum_{i,k} H_{i,k} \delta\alpha_i \delta\alpha_k + O(\|\delta\alpha\|^3)$$

- Analytic integration to compute statistics

$$\mu_j \simeq j(\mu_\alpha) + \frac{1}{2} \sum_{i,k} C_{i,k} H_{i,k}$$

$$\sigma_j^2 \simeq \sum_{i,k} C_{i,k} G_i G_k + \frac{1}{4} \sum_{i,k,l,m} (C_{i,l} C_{k,m} + C_{i,m} C_{k,l}) H_{i,k} H_{l,m}$$

Uncertainty propagation using derivatives

Problem description

- Drag is a functional: $$j : \gamma \mapsto j(\gamma) = J(\gamma, W)$$
- Constrained by a state equation: $$\Psi(\gamma, W) = 0.$$

Derivatives

- Derivative of the functional: $$\frac{dj}{d\gamma_i} = \frac{dj}{d\gamma} e_i = \frac{\partial J}{\partial \gamma_i} + \frac{\partial J}{\partial W}\frac{dW}{d\gamma_i}.$$
- Derivative of the state equation: $$\frac{\partial \Psi}{\partial \gamma_i} + \frac{\partial \Psi}{\partial W}\frac{dW}{d\gamma_i} = 0.$$

Uncertainty propagation using derivatives

Direct approach

- Sensitivity equation: $$\frac{\partial \Psi}{\partial W}\theta_i = -\frac{\partial \Psi}{\partial \gamma_i}.$$

Adjoint approach

- Combine previous equations:

$$\left(\frac{dj}{d\gamma}\right)^\top = \left(\frac{\partial J}{\partial \gamma}\right)^\top - \left(\frac{\partial \Psi}{\partial \gamma}\right)^\top \left(\frac{\partial \Psi}{\partial W}\right)^{-\top} \left(\frac{\partial J}{\partial W}\right)^\top$$

- Adjoint equation:

$$\left(\frac{\partial \Psi}{\partial W}\right)^\top \Pi = \left(\frac{\partial J}{\partial W}\right)^\top \qquad \left(\frac{dj}{d\gamma}\right)^\top = \left(\frac{\partial J}{\partial \gamma}\right)^\top - \left(\frac{\partial \Psi}{\partial \gamma}\right)^\top \Pi.$$

Uncertainty propagation using derivatives

Second-order derivative

- Functional:

$$\frac{d^2 j}{d\gamma_i d\gamma_k} = D^2_{i,k} J + \frac{\partial J}{\partial W}\frac{d^2 W}{d\gamma_i d\gamma_k}$$

$$D^2_{i,k} J = \frac{\partial}{\partial\gamma}\left(\frac{\partial J}{\partial\gamma} e_i\right) e_k + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma} e_i\right)\frac{dW}{d\gamma_k} + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma} e_k\right)\frac{dW}{d\gamma_i} + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial W}\frac{dW}{d\gamma_i}\right)\frac{dW}{d\gamma_k}$$

- State equation:

$$D^2_{i,k}\Psi + \frac{\partial\Psi}{\partial W}\frac{d^2 W}{d\gamma_i d\gamma_k} = 0$$

$$D^2_{i,k}\Psi = \frac{\partial}{\partial\gamma}\left(\frac{\partial\Psi}{\partial\gamma} e_i\right) e_k + \frac{\partial}{\partial W}\left(\frac{\partial\Psi}{\partial\gamma} e_i\right)\frac{dW}{d\gamma_k} + \frac{\partial}{\partial W}\left(\frac{\partial\Psi}{\partial\gamma} e_k\right)\frac{dW}{d\gamma_i} + \frac{\partial}{\partial W}\left(\frac{\partial\Psi}{\partial W}\frac{dW}{d\gamma_i}\right)\frac{dW}{d\gamma_k}$$

Uncertainty propagation using derivatives

Direct-on-direct approach:

- Combine previous equations:

$$
\begin{aligned}
\frac{d^2 j}{d\gamma_i d\gamma_k} &= D^2_{i,k} J - \frac{\partial J}{\partial W} \left(\frac{\partial \Psi}{\partial W} \right)^{-1} D^2_{i,k} \Psi \\
&= D^2_{i,k} J - \Pi^\top D^2_{i,k} \Psi,
\end{aligned}
$$

Uncertainty propagation using derivatives

Algorithm:

1. Solve for the adjoint variables Π in:
$$\left(\frac{\partial\Psi}{\partial W}\right)^{\top}\Pi = \left(\frac{\partial J}{\partial W}\right)^{\top}$$

2. Compute the gradient of j:
$$\left(\frac{dj}{d\gamma}\right)^{\top} = \left(\frac{\partial J}{\partial\gamma}\right)^{\top} - \left(\frac{\partial\Psi}{\partial\gamma}\right)^{\top}\Pi$$

3. For each $i \in 1 \ldots n$:
Solve for the flow sensitivities θ_i in:
$$\left(\frac{\partial\Psi}{\partial W}\right)\theta_i = -\left(\frac{\partial\Psi}{\partial\gamma_i}\right)$$

4. For each $i \in 1 \ldots n$ and $k \in 1 \ldots i$, compute:
$$\frac{d^2 j}{d\gamma_i d\gamma_k} = D^2_{i,k}J - \Pi^{\top}(D^2_{i,k}\Psi)$$

Use of Automatic Differentiation (AD)

Principles

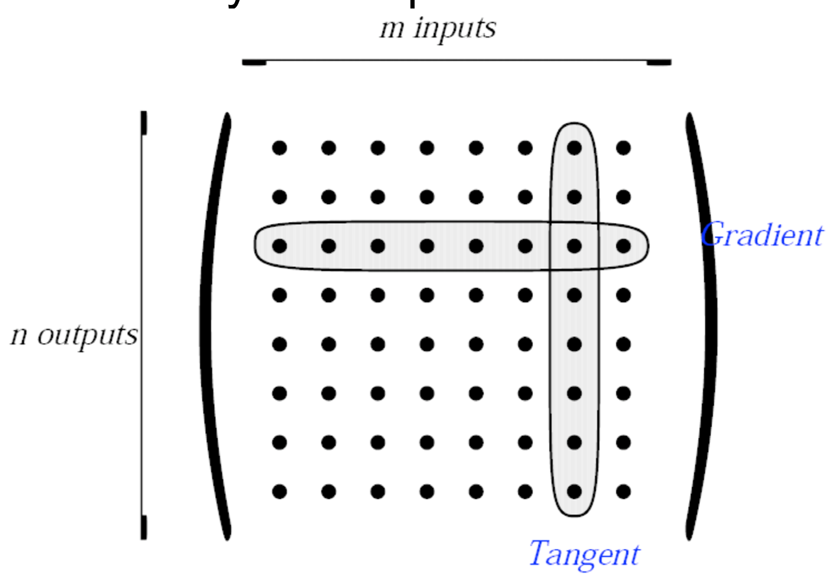
- Consider a program as a sequence of elementary instructions
- Apply the chain rule to differentiate these elementary instructions
- Write automatically a new program that compute the derivative

Results

- If a program is considered as a function $v = \phi(u)$, its derivative is a Jacobian matrix
- AD tool TAPENADE allows to compute the matrix by vector products:

Tangent mode: $u, \dot{u} \mapsto \left.\frac{\partial \phi}{\partial u}\right|_u \dot{u}$

Reverse mode: $u, \bar{\phi} \mapsto \left(\left.\frac{\partial \phi}{\partial u}\right|_u \right)^\top \bar{\phi}$



Use of Automatic Differentiation (AD)

Example : Tangent mode

Original F77 code

```
SUBROUTINE FOO(v1,    v2,    v4,    p1)

 REAL v1,v2,v3,v4,p1


 v3 = 2.0*v1 + 5.0


 v4 = v3 + p1*v2/v3
END
```

Derivative of v4 w.r.t. v1 and v2

```
SUBROUTINE FOO(v1,v1d,v2,v2d,v4,v4d,p1)
 REAL v1d,v2d,v3d,v4d
 REAL v1,v2,v3,v4,p1
 v3d = 2.0*v1d
 v3 = 2.0*v1 + 5.0
 v4d = v3d + p1*(v2d*v3-v2*v3d)/(v3*v3)
 v4 = v3 + p1*v2/v3
END
```

Use of Automatic Differentiation (AD)

Application to aerodynamic test-case : adjoint equation

$$\texttt{state}(\underset{\downarrow \atop \Psi}{\texttt{psi}}, \overset{\gamma \atop \downarrow}{\texttt{gamma}}, \overset{W \atop \downarrow}{\texttt{w}})$$

$$\texttt{func}(\underset{\downarrow \atop J}{\texttt{j}}, \overset{\gamma \atop \downarrow}{\texttt{gamma}}, \overset{W \atop \downarrow}{\texttt{w}})$$

Application of Reverse Mode :

$$\texttt{state\_b}(\underset{\downarrow \atop \Psi}{\texttt{psi}}, \overset{\bar{\Psi} \atop \downarrow}{\texttt{psib}}, \overset{\gamma \atop \downarrow}{\texttt{gamma}}, \underset{\downarrow \atop \bar{\gamma}}{\texttt{gammab}}, \overset{W \atop \downarrow}{\texttt{w}}, \underset{\downarrow \atop \bar{W}}{\texttt{wb}})$$

$$\bar{W} = \left(\frac{\partial \Psi}{\partial W}\right)^{\top} \bar{\Psi}$$

Application of Reverse Mode :

$$\texttt{func\_b}(\underset{\downarrow \atop J}{\texttt{j}}, \overset{\bar{J} \atop \downarrow}{\texttt{jb}}, \overset{\gamma \atop \downarrow}{\texttt{gamma}}, \underset{\downarrow \atop \bar{\gamma}}{\texttt{gammab}}, \overset{W \atop \downarrow}{\texttt{w}}, \underset{\downarrow \atop \bar{W}}{\texttt{wb}})$$

$$\bar{\gamma} = \left(\frac{\partial J}{\partial \gamma}\right)^{\top} \bar{J} \qquad \bar{W} = \left(\frac{\partial J}{\partial W}\right)^{\top} \bar{J}$$

$$\left(\frac{\partial \Psi}{\partial W}\right)^{\top} \Pi = \left(\frac{\partial J}{\partial W}\right)^{\top}$$

Use of Automatic Differentiation (AD)

Application to aerodynamic test-case : second-order derivative

`func(j, gamma, w)`

$\gamma \rightarrow$ gamma, $W \rightarrow$ w ; j $\rightarrow J$

Application of Direct Mode:

`func_d(j, jd, gamma, gammad, w, wd)`

$\gamma \rightarrow$ gamma, $\dot\gamma \rightarrow$ gammad, $W \rightarrow$ w, $\dot W \rightarrow$ wd ; j $\rightarrow J$, jd $\rightarrow \dot j$

$$\ddot{j} = \frac{\partial}{\partial\gamma}\left(\frac{\partial J}{\partial\gamma}\dot\gamma\right)\dot\gamma_0 + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma}\dot\gamma\right)\dot W_0 + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma}\dot\gamma_0\right)\dot W + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial W}\dot W\right)\dot W_0$$

Application of Direct Mode:

`func_dd(j, jd, jdd, gamma, gammad0, gammad, w, wd0, wd)`

$\gamma \rightarrow$ gamma, $\dot\gamma_0 \rightarrow$ gammad0, $\dot\gamma \rightarrow$ gammad, $W \rightarrow$ w, $\dot W_0 \rightarrow$ wd0, $\dot W \rightarrow$ wd ; j $\rightarrow J$, jd $\rightarrow \dot j$, jdd $\rightarrow \ddot j$

$$D^2_{i,k}J = \frac{\partial}{\partial\gamma}\left(\frac{\partial J}{\partial\gamma}e_i\right)e_k + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma}e_i\right)\frac{dW}{d\gamma_k} + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial\gamma}e_k\right)\frac{dW}{d\gamma_i} + \frac{\partial}{\partial W}\left(\frac{\partial J}{\partial W}\frac{dW}{d\gamma_i}\right)\frac{dW}{d\gamma_k}$$

$$\frac{d^2 j}{d\gamma_i d\gamma_k} = D^2_{i,k}J - \Pi^\top (D^2_{i,k}\Psi)$$

Estimation of uncertainty for fields

Principle

- Compute sensitivity fields by a direct approach

$$\mathbf{s_u} = \frac{\partial \mathbf{u}}{\partial a} \quad s_p = \frac{\partial p}{\partial a}$$

- Differentiate state equations to obtain the PDEs for sensitivity fields (continuous sensitivity equation method) :

$$\begin{aligned}
\rho \mathbf{u} \cdot \nabla \mathbf{u} &= -\nabla p + \nabla \cdot \tau(\mathbf{u}) + \mathbf{f}, \\
\nabla \cdot \mathbf{u} &= 0, \\
\rho' \mathbf{u} \cdot \nabla \mathbf{u} + \rho\left(\mathbf{s_u} \cdot \nabla \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{s_u}\right) &= -\nabla s_p + \nabla \cdot \tau(\mathbf{s_u}) + \nabla \cdot S_\tau(\mathbf{u}) + \mathbf{f}', \\
\nabla \cdot \mathbf{s_u} &= 0,
\end{aligned}$$

- Solve sensitivity equations using similar numerical methods

Methodology for statistical estimation based on surrogate models

Use of meta-models

- Compute the functional for a few number of uncertain parameter values
- Construct a meta-model
- Numerical integration or Monte-Carlo to compute statistics

Use of meta-models

Principle

- Build a database (several drag computations for different condition values) : Latin Hypercube Sampling (LHS)

- Construct an approximate model for the drag:
  - Least-squares fitting
  - Artifical Neural Networks (ANNs)
  - Radial Basis Functions (RBFs)
  - Gaussian processes (Kriging)

Use of meta-models

Radial Basis Functions (RBFs)

- Seek an approximation of the form:

$$\hat{j}(\gamma) = \sum_{i=1}^{N} \omega_i \, \phi_i(\gamma)$$

$$\phi_i(\gamma) = \Phi(\|\gamma - \gamma^i\|)$$

$$\Phi(r) = e^{-\frac{r^2}{s^2}}$$

- Interpolation condition yields:

$$\begin{pmatrix} \phi_1(\gamma^1) & \dots & \phi_N(\gamma^1) \\ \phi_1(\gamma^2) & \dots & \phi_N(\gamma^2) \\ \dots & \dots & \dots \\ \phi_1(\gamma^N) & \dots & \phi_N(\gamma^N) \end{pmatrix} \begin{Bmatrix} \omega_1 \\ \omega_2 \\ \dots \\ \omega_N \end{Bmatrix} = \begin{Bmatrix} j(\gamma^1) \\ j(\gamma^2) \\ \dots \\ j(\gamma^N) \end{Bmatrix}$$

- Optimization of attenuation factor by leave-one-out + PSO method

Use of meta-models

Gaussian Processes (Kriging)

- Hypothesis : the sample results from a Gaussian process:

$$p(J^N|\Gamma^N) = \frac{\exp\left(-\frac{1}{2}{J^N}^\top C_N^{-1} J^N\right)}{\sqrt{(2\pi)^N \det(C_N)}} \quad c(\gamma^i,\gamma^k) = \theta_1 \exp\left[-\frac{1}{2}\sum_{l=1}^{n}\frac{(\gamma_l^i-\gamma_l^k)^2}{r_l^2}\right] + \theta_2$$

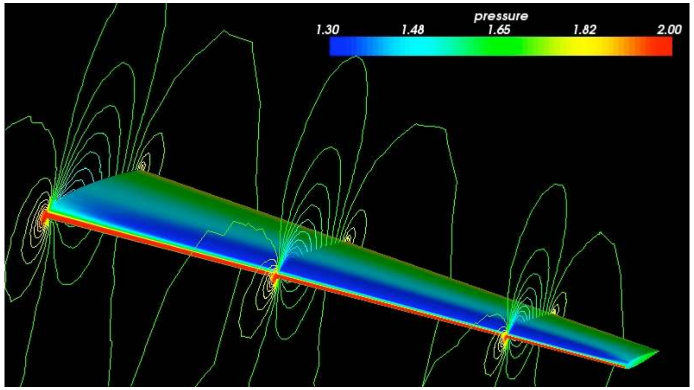
- Maximize the likelihood of the sample (log-likelihood) by PSO method:

$$\mathfrak{L} = {J^N}^\top C_N^{-1} J^N + \log\det(C_N)$$

- Derive estimation of a new point (mean and variance)

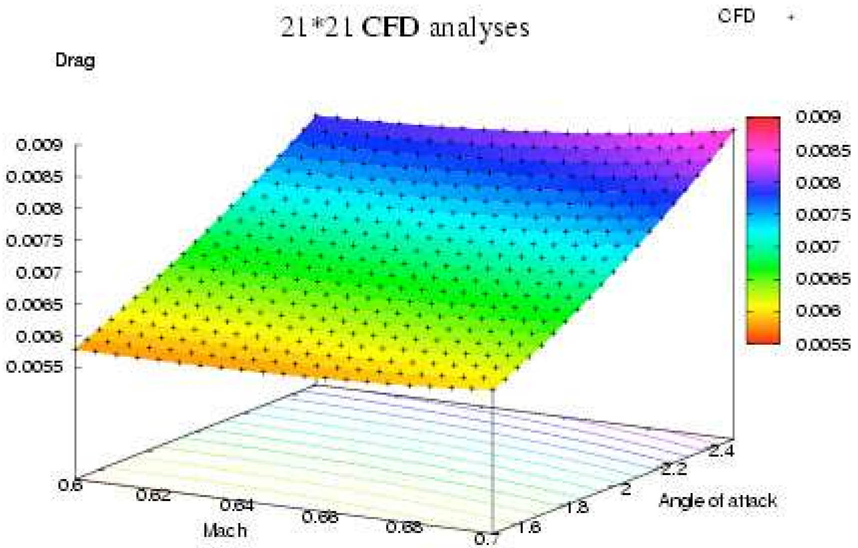
$$\hat{j}^{N+1} = k^\top C_N^{-1} J^N, \qquad \sigma^2_{j^{N+1}} = \kappa - k^\top C_N^{-1} k$$

Application : subsonic flow around a wing



Test-case:

- 3D Eulerian flow
- Nominal conditions: M=0.65 α = 2°
- Gaussian random fluctuations



Reference results:

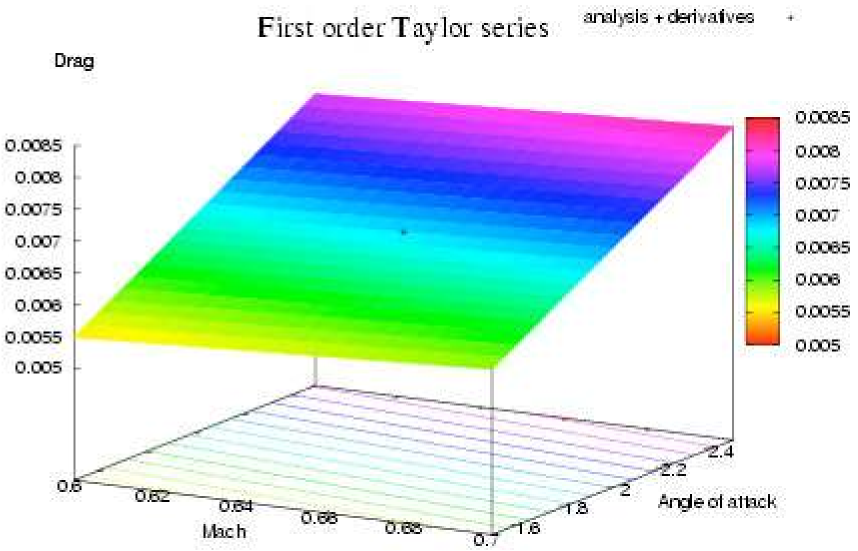
- 21 * 21 CFD analyses

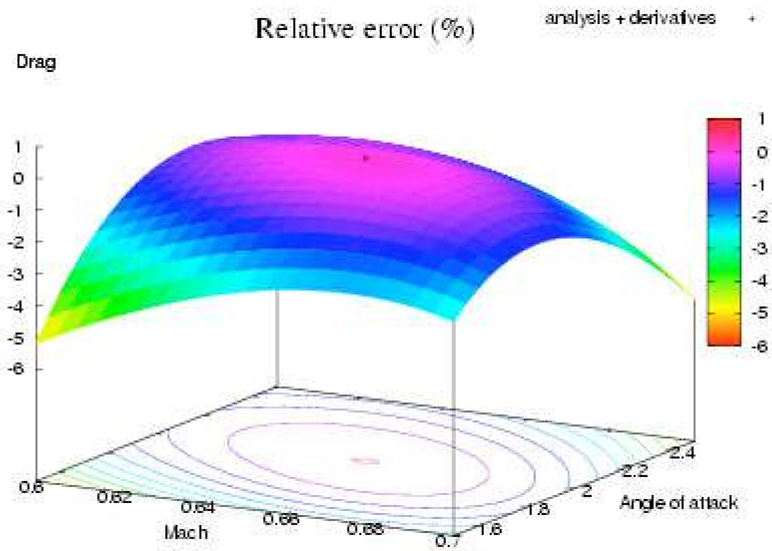
Martinelli & Duvigneau, AIAA-2071, 2008

Martinelli & Duvigneau, Computers & Fluids, 2010

Application : subsonic flow around a wing

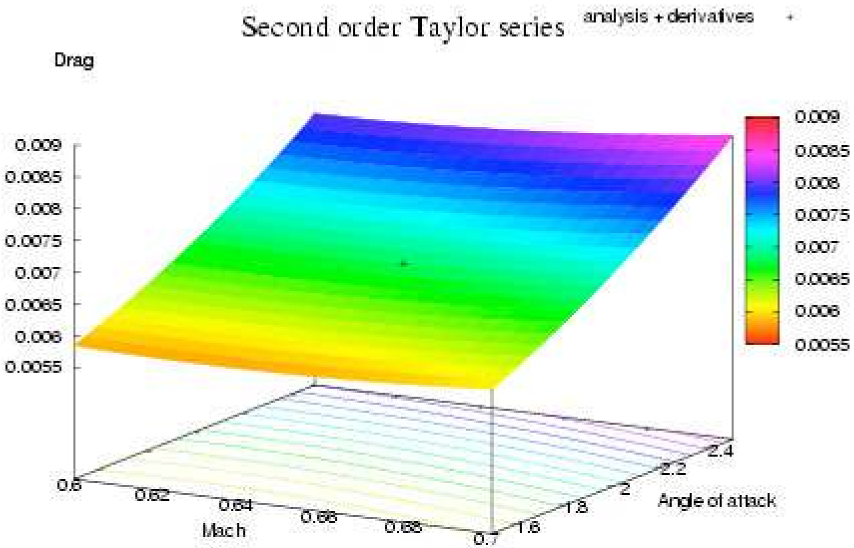
First-order Taylor series :

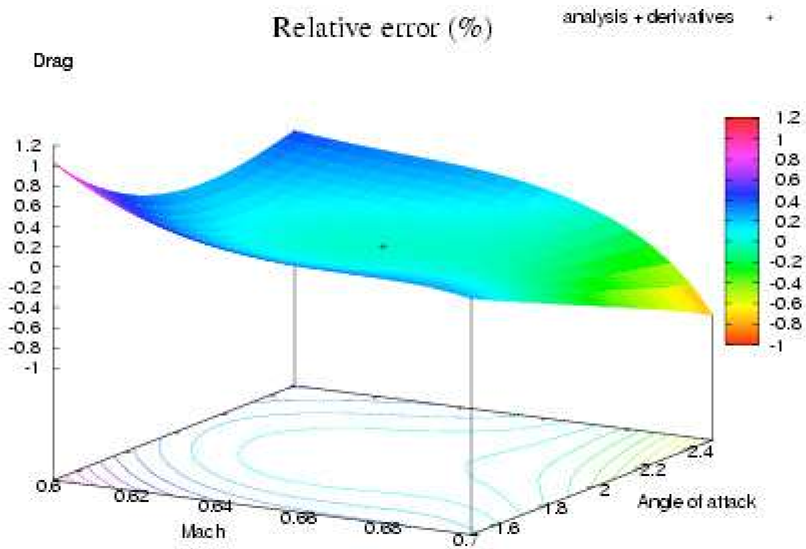




Application : subsonic flow around a wing

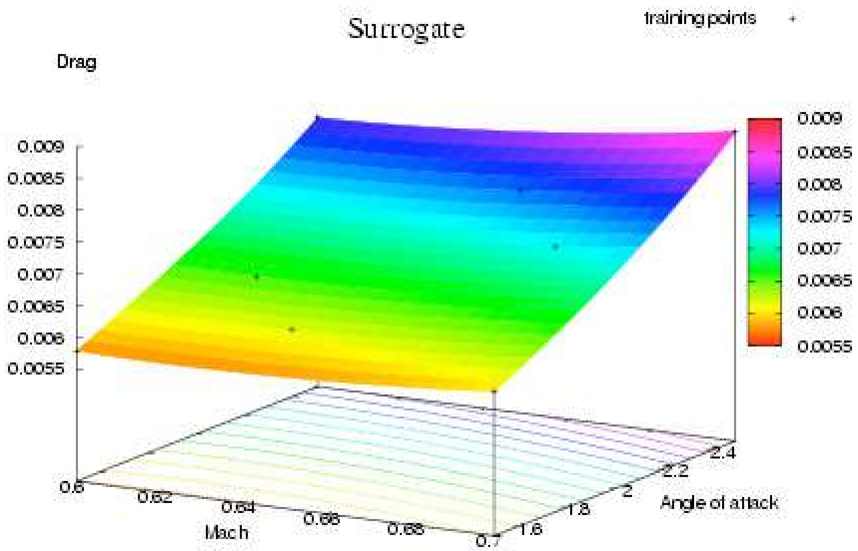
Second-order Taylor series :

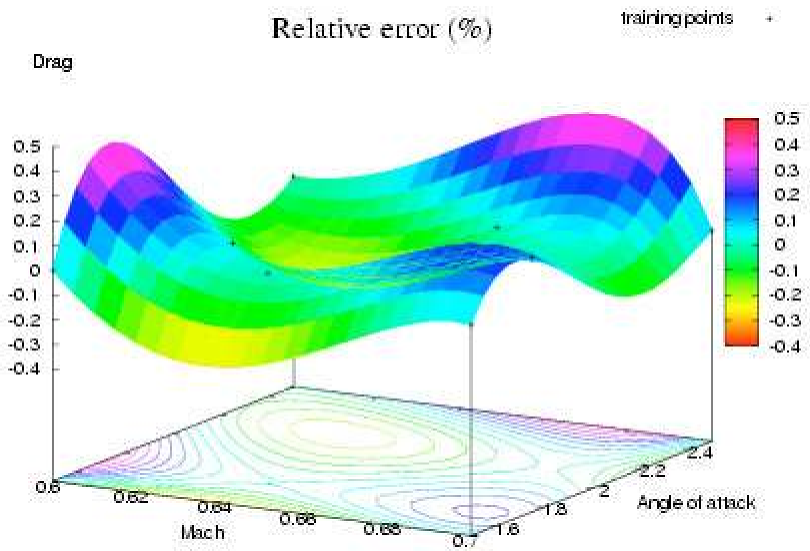




Application : subsonic flow around a wing

Radial Basis Functions with 8 training points :





Application : subsonic flow around a wing

Comparison of statistic values :

Metamodel	Points	Expectation	Expectation error
RBF	8	$6.85507\,10^{-3}$	$2.43491\,10^{-6}$
RBF	23	$6.85762\,10^{-3}$	$1.05410\,10^{-7}$
KRG	8	$6.85257\,10^{-3}$	$4.94454\,10^{-6}$
KRG	23	$6.85762\,10^{-3}$	$1.14150\,10^{-7}$

Metamodel	Points	Variance	Variance error
RBF	8	$1.56176\,10^{-7}$	$8.41631\,10^{-10}$
RBF	23	$1.56595\,10^{-7}$	$1.26098\,10^{-9}$
KRG	8	$1.55076\,10^{-7}$	$2.57555\,10^{-10}$
KRG	23	$1.56575\,10^{-7}$	$1.24084\,10^{-9}$

	Expectation	Expectation error
First-order	$6.83049\,10^{-3}$	$2.70266\,10^{-5}$
Second-order	$6.86056\,10^{-3}$	$3.04649\,10^{-6}$

	Variance	Variance error
First-order	$1.54665\,10^{-7}$	$6.69276\,10^{-10}$
Second-order	$1.55758\,10^{-7}$	$4.23060\,10^{-10}$

Application : subsonic flow around a wing

Comparison of memory requirements :

	Memory in Mb
Flow solver	130
First derivatives	250
Second derivatives	120
GMRES linear solver	250
Preconditionners	340
Total	1090

Comparison of CPU costs :

	CPU time in second
Flow solver	403
Gradient	255
Hessian	278
Total	936

	CPU time in second
Sequential database	3250
Parallel database	440

Application to robust design

Test-case

- Supersonic business jet
- Cruise regime (M=0.83)
- Minimize drag
- Lift constraint
- Uncertain Mach number (Gaussian)

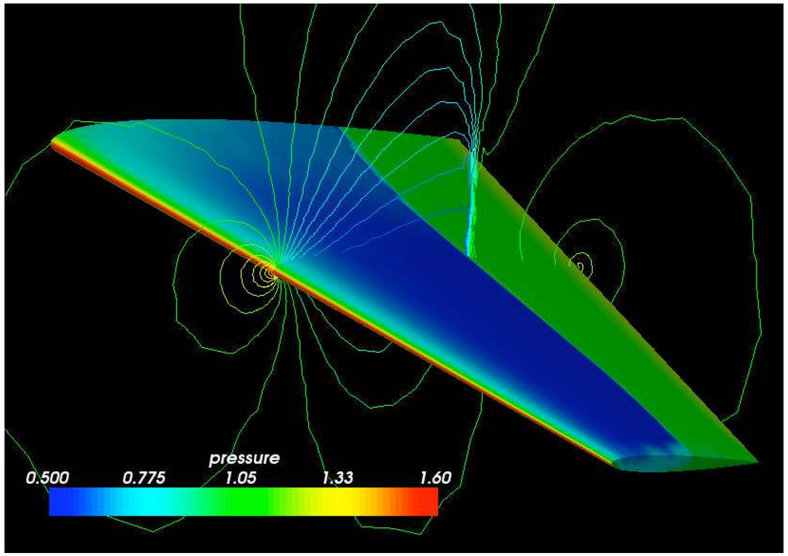
Duvigneau, EUROGEN, 2007

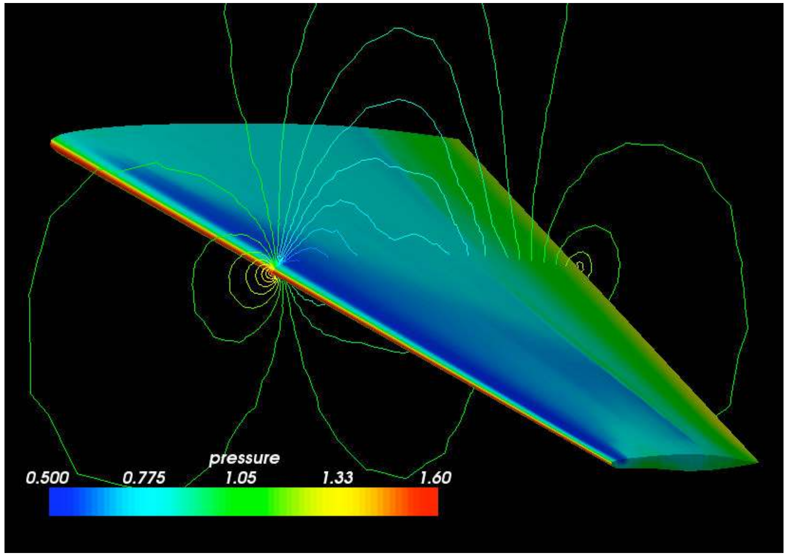
Duvigneau, Martinelli & Chandrashekar, ISTE-Wiley, 2010

Application to robust design

Single-point optimization

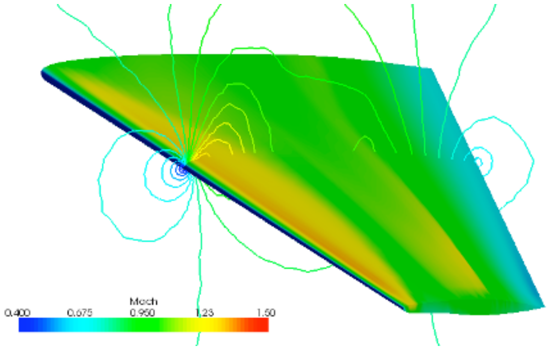
- Optimization at cruise regime only (M=0.83)
- Cost function (drag) : 0.02633 to 0.01139
- Constraint (lift) : 0.3190 to 0.3188



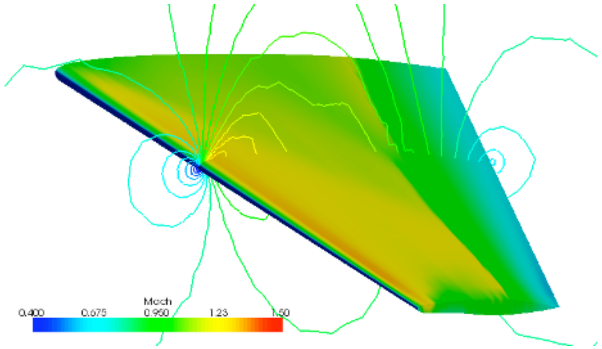


Application to robust design

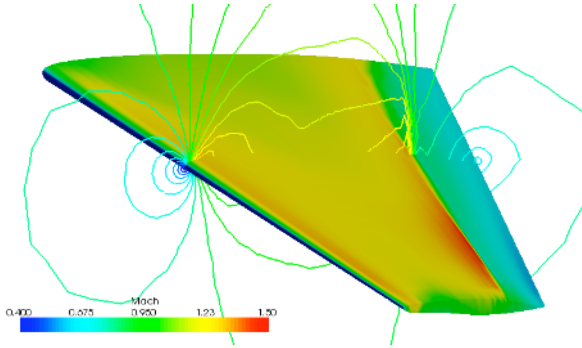
Effect of Mach perturbation for the optimum shape



- M=0.81



- M=0.83

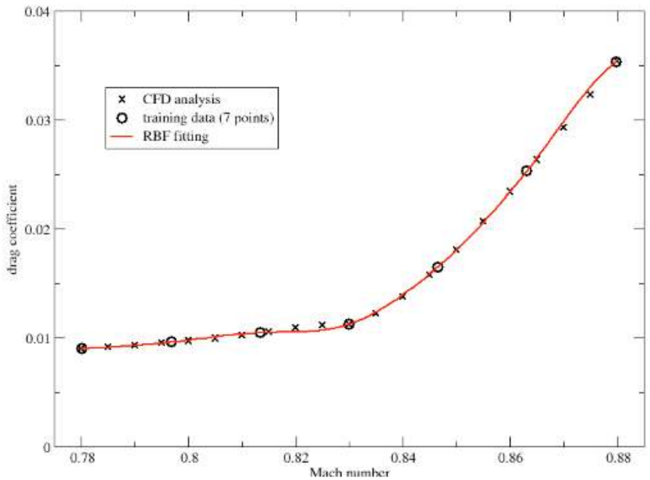


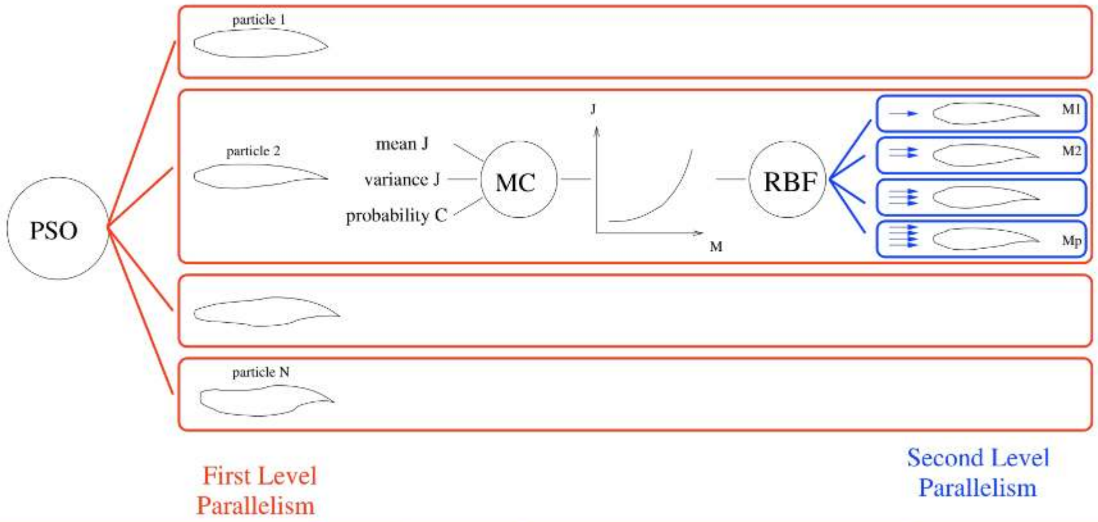
- M=0.85

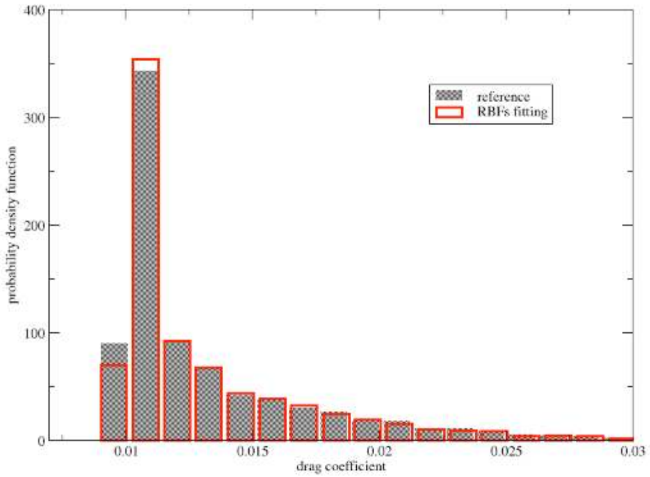
Application to robust design

Robust design procedure

- Minimize both drag mean and variance
- Lift constraint (probabilistic)
- Uncertain Mach number (Gaussian)
- Meta-model based approach (7 points)
- Calcul : 256 proc. 10H (IBM Power4)



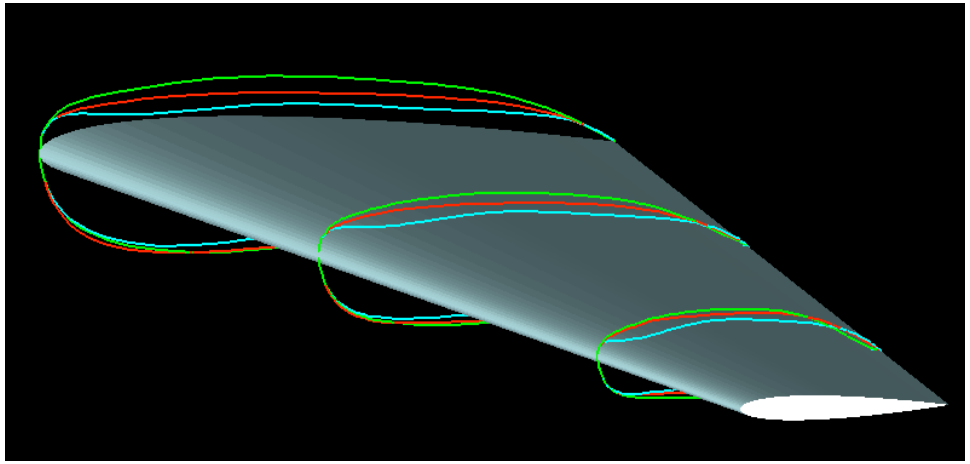


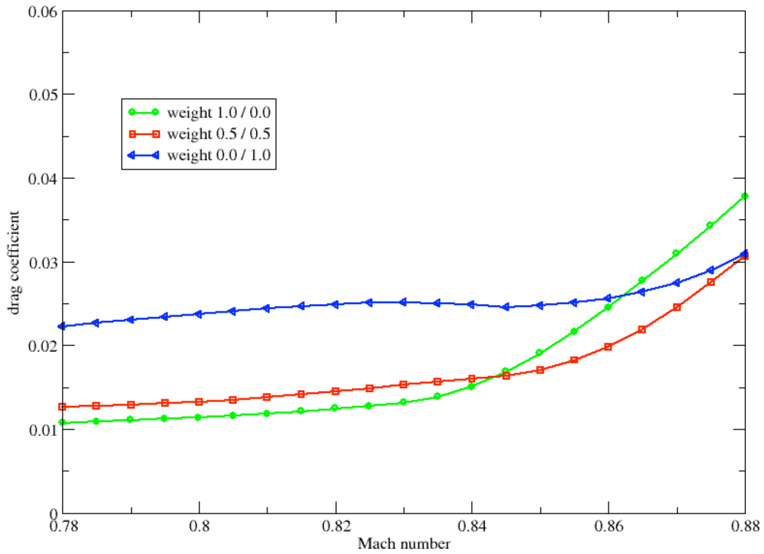


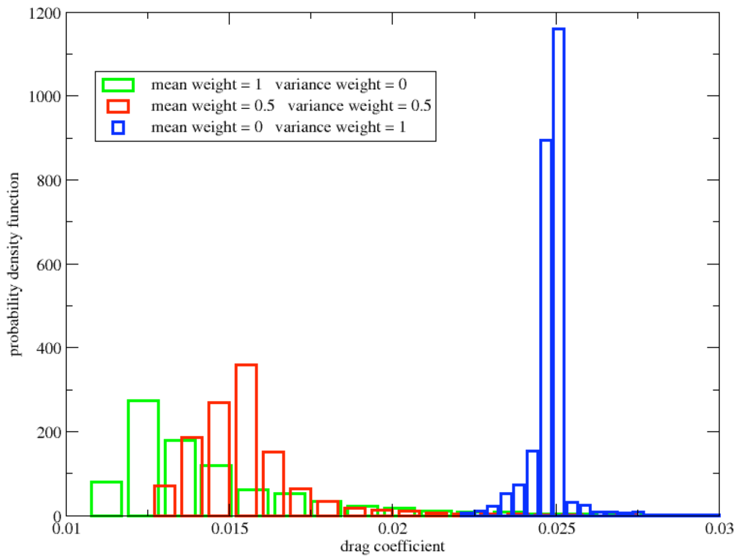
Application to robust design

Solution

- Minimization of drag mean only
- Minimization of drag variance only
- Minimization of drag variance and mean

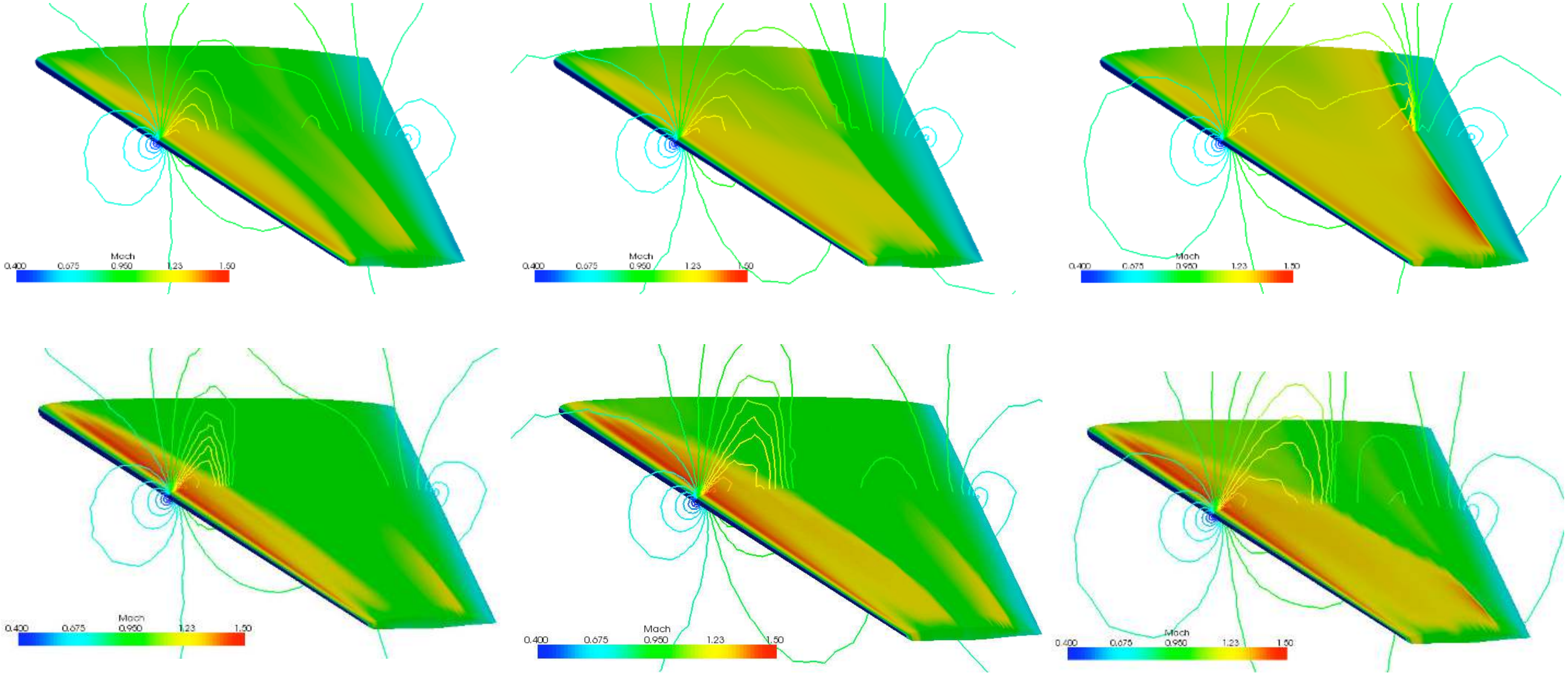






Application to robust design

Effect of Mach perturbation for the optimum shape (bottom : robust)

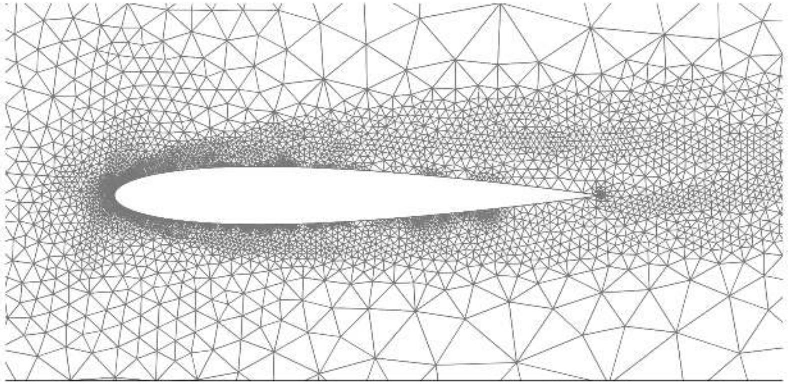


- M=0.81
- M=0.83
- M=0.85

Application to uncertainty field estimation

Test-case

- 2D incompressible Navier-Stokes
- Adaptive finite-element method
- Laminar flow around airfoil
- Uncertainty due to thickness, camber and incidence



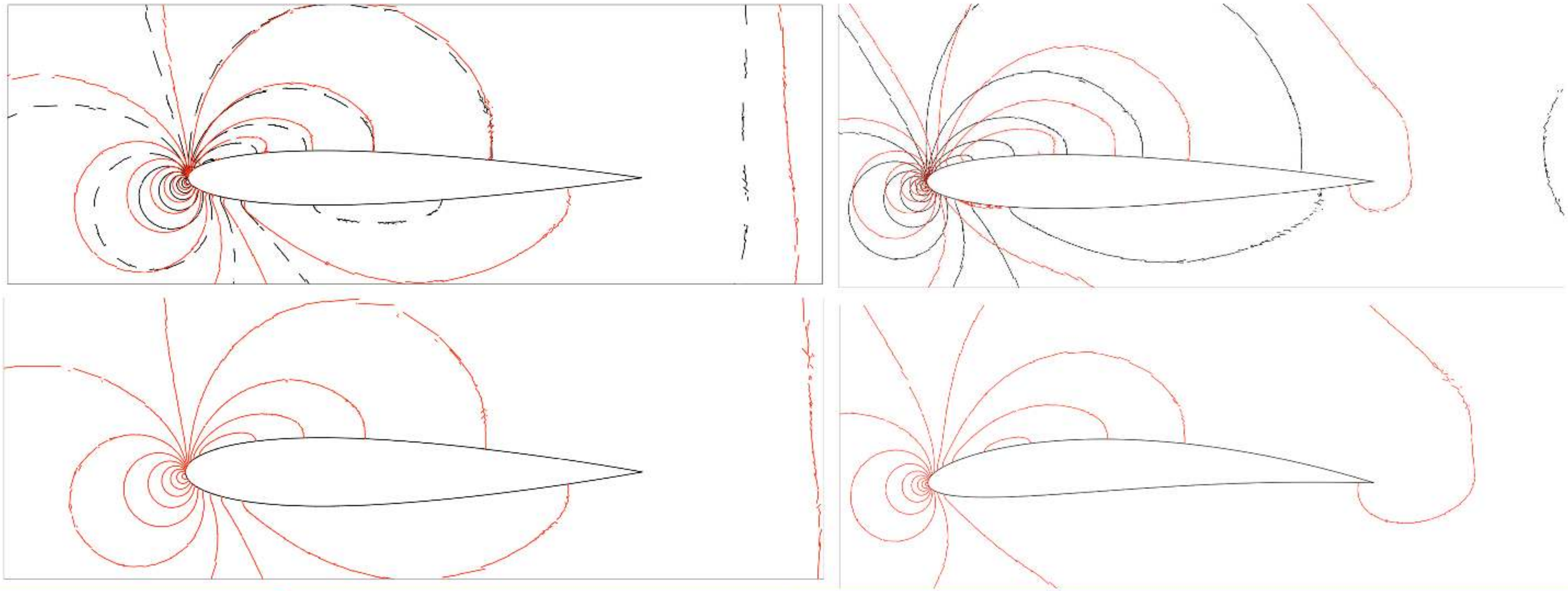
Duvigneau & Pelletier, AIAA-0127, 2005

Duvigneau & Pelletier, Int. J. Comp. Fluid Dynamics, 2006

Application to uncertainty field estimation

Validation by estimating nearby solution

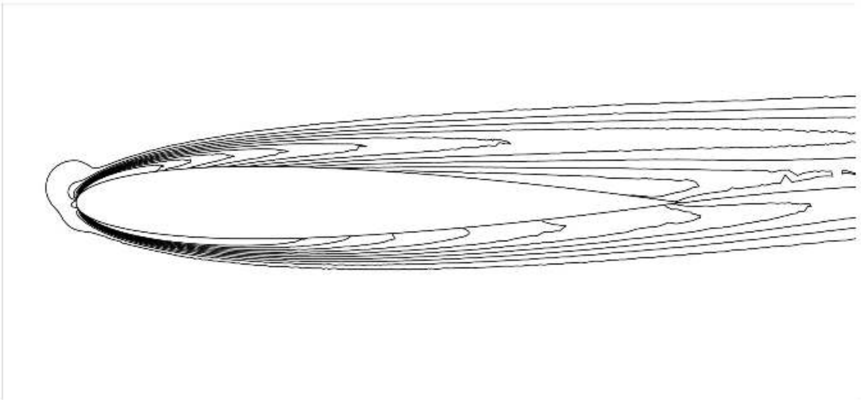
- Change of thickness (12% to 15%)
- Change of camber (0% to 4%)

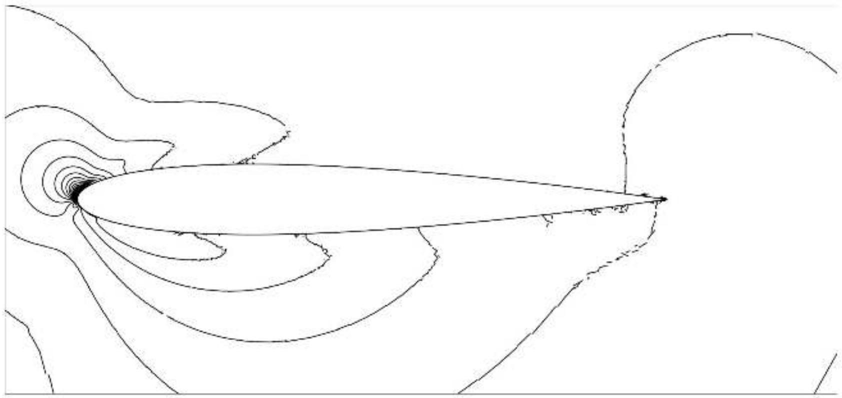


Application to uncertainty field estimation

Uncertainty estimation

- Uncertainty of thickness : 1%
- Uncertainty of camber : 1%
- Uncertainty of incidence : 0.5 degree

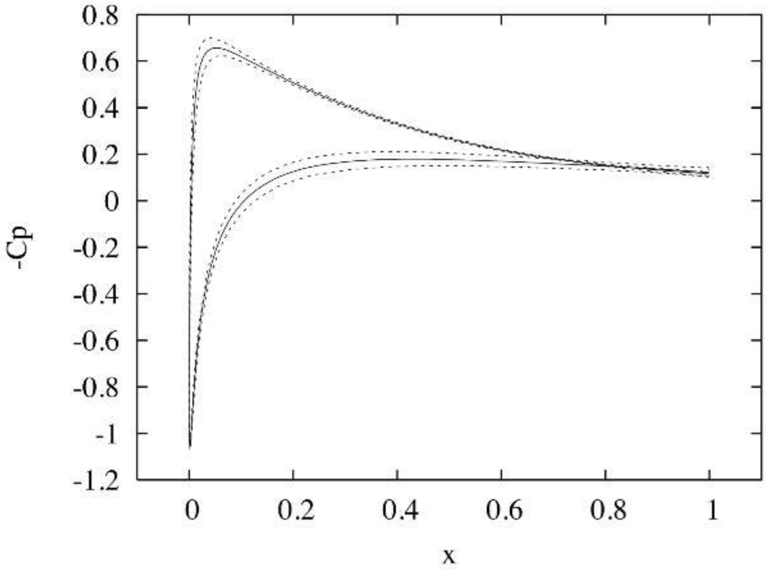


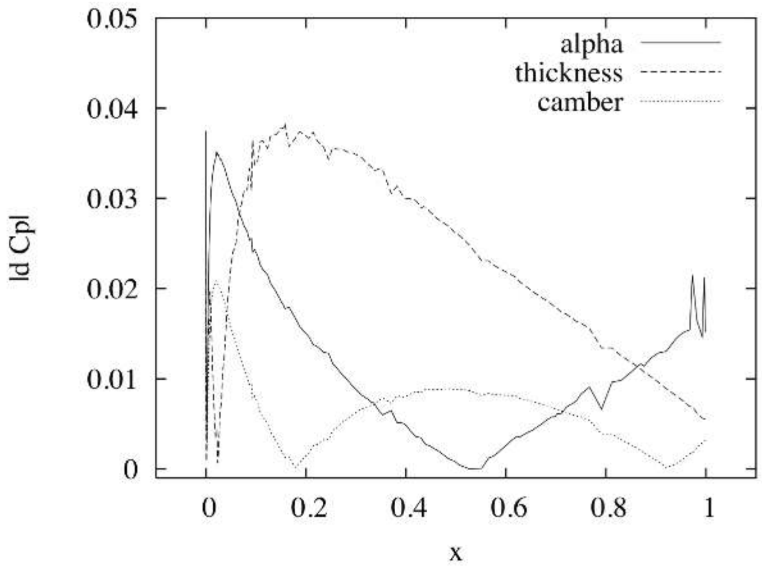


Application to uncertainty field estimation

Uncertainty estimation

- Uncertainty of thickness : 1%
- Uncertainty of camber : 1%
- Uncertainty of incidence : 0.5 degree





Conclusion

AD-based approach :

- AD usefull to reduce human cost
- Difficulties in transonic regime (limiters not differentiable)
- Low CPU cost
- High memory requirements

CSEM-based approach :

- No differentiation of discrete entities
- Human cost higher

Metamodel-based approach :

- Easy to use
- CPU cost : only due to database building
- Strong increase of CPU cost with the number of parameters
- Parallel computing