NATIONAL TECHNICAL UNIVERSITY OF ATHENS
Parallel CFD & Optimization Unit
Laboratory of Thermal Turbomachines

Computation of second-order derivatives in aerodynamic optimization

VARNA-Sep. 2010

Kyriakos C. GIANNAKOGLOU, Professor NTUA
kgianna@central.ntua.gr
http://velos0.ltt.mech.ntua.gr/research/

Development of Adjoint Methods

OBJECTIVES:

- **Development of Discrete/Continuous Adjoint Methods for the computation of first/second/third order (exact) sensitivities of various objective functions.**

TOPICS:

- **Metrics-free adjoint formulations**
- **Validated vs finite-differences, direct differentiation, complex variables**
- **Various parameterizations (including normal sensitivities, free form deformation, NURBS, etc)**
- **Based on in-house CFD tools and OpenFOAM**
- **The adjoint to the turbulence model equations**
- **Adjoint wall function techniques**
- **Computation of exact or “exact” Hessian matrices**
- **Robust Design**
- **Adjoint for the optimization of flow control**
- **Adjoint methods for Cluster- & Grid-Computing.**
- **Adjoint method implementations on Graphics Processing Units (GPUs)**

FUNDED BY:

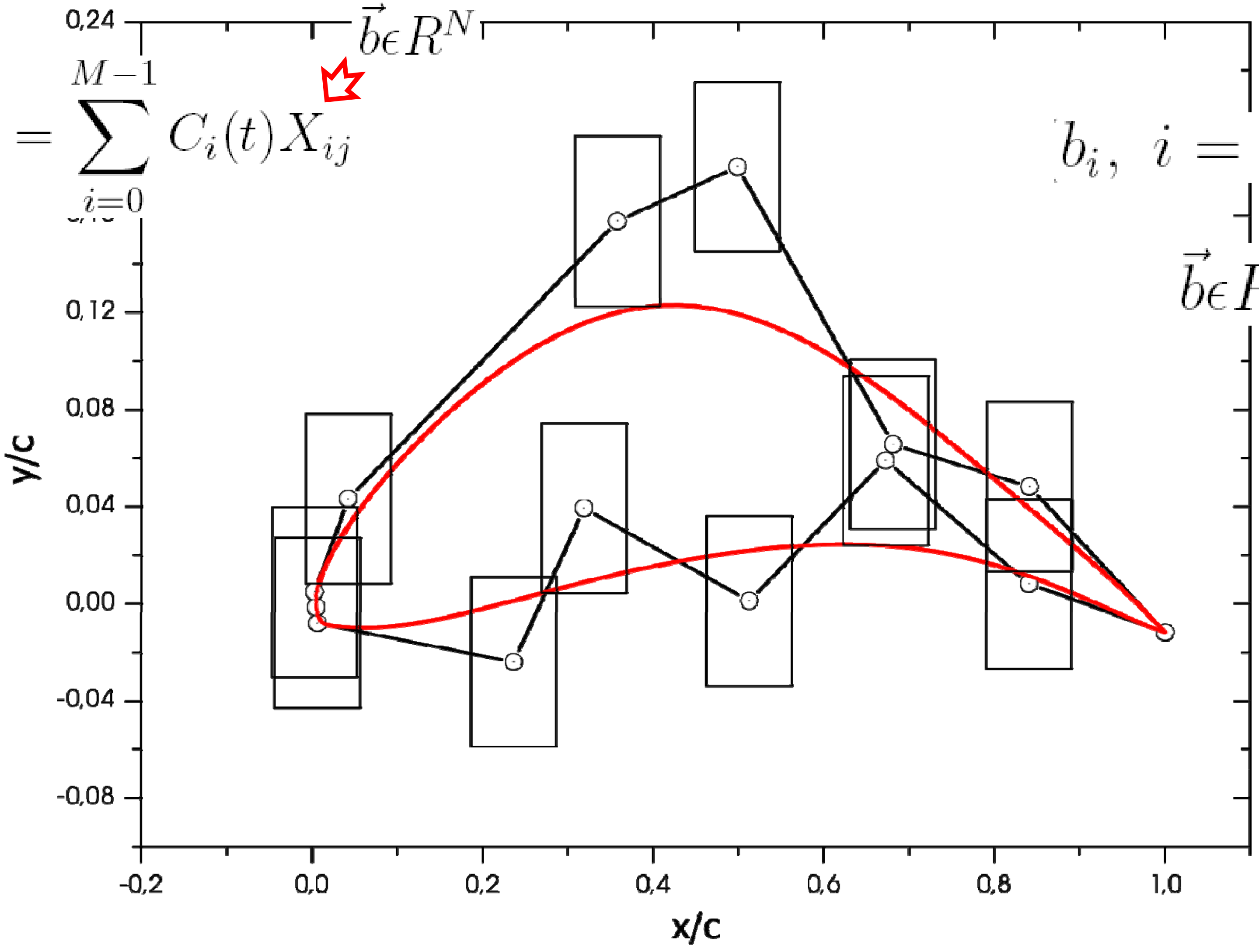
NTUA

A.S. Onassis Foundation

State Scholarship Foundation

The Continuous Adjoint Approach

Shape Parameterization - Design Variables



$\vec{b}\epsilon R^N$

$$x_j(t)=\sum_{i=0}^{M-1} C_i(t)X_{ij}$$

$b_i,\ i=1,\ldots,N$

$\vec{b}\epsilon R^N$

Flow Model - State Equations & Discretization

Flow Model (State Equations) :

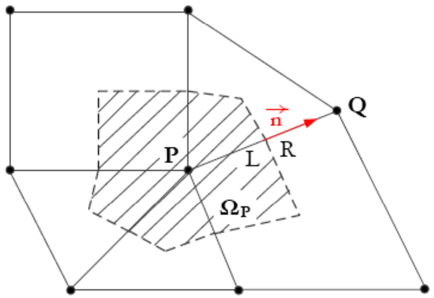
Example: Euler equations, compressible fluid

$$R_n = \frac{\partial U_n}{\partial t} + \frac{\partial f^{inv}_{nk}}{\partial x_k} = 0 \qquad \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{bmatrix} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ E \end{bmatrix} , \quad \begin{bmatrix} f^{inv}_{1k} \\ f^{inv}_{2k} \\ f^{inv}_{3k} \\ f^{inv}_{4k} \\ f^{inv}_{5k} \end{bmatrix} = \begin{bmatrix} \rho u_k \\ \rho u_1 u_k + p\delta_{k1} \\ \rho u_2 u_k + p\delta_{k2} \\ \rho u_3 u_k + p\delta_{k3} \\ u_k(E+p) \end{bmatrix}$$

Solid Wall Boundary Conditions : $u_k n_k = 0$

Discretization :

$$\sum_{Q\epsilon nei(P)} h_{n,PQ} \Delta S_{PQ} = 0$$



$$h_n|_{PQ} = \frac{1}{2}\left(A_{nk}|_P U_k|_L + A_{nk}|_Q U_k|_R\right) - \frac{1}{2}|A_{nk}|_{PQ}\left(U_k|_L - U_k|_R\right)$$

Objective Functions

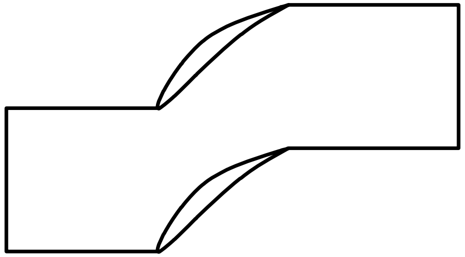
◆

$$F=\frac{1}{2}\int_{S_w}(p-p_{target})^2 dS$$

- Inverse design.
- Functional and design variables correspond to the same boundary !!!

◆

$$F=\int_{S_{in}}\rho V_n p_t dS-\int_{S_{out}}\rho V_n p_t dS$$



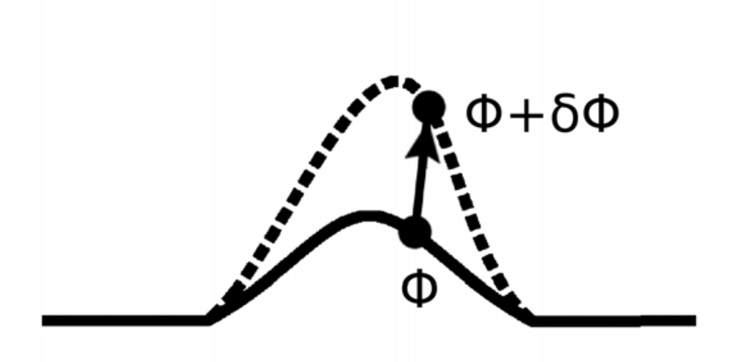
- Losses Minimization.
- Functional and design variables correspond to different boundaries !!!

◆

$$F=\int_{S_{out}}\rho V_n s dS-\int_{S_{in}}\rho V_n s dS=\int_{\Omega}\rho u_i\frac{\partial s}{\partial x_i}d\Omega=\int_{\Omega}\frac{1}{\mathrm{T}}\tau_{ij}\frac{\partial u_i}{\partial x_j}d\Omega$$

- Losses Minimization.
- Transformation of the inlet/outlet integral to a field integral !!!

Direct Differentiation (DD) Approach



$$\underbrace{\frac{\delta \Phi}{\delta b_i}}_{\text{Total Variation}} = \underbrace{\frac{\partial \Phi}{\partial b_i}}_{\text{Partial Variation}} + \frac{\partial \Phi}{\partial x_l}\frac{\delta x_l}{\delta b_i}$$

$$\frac{\delta}{\delta b_i}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right) = 0 \Rightarrow \frac{\partial}{\partial b_i}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right) + \frac{\partial^2 f_{nk}^{inv}}{\partial x_k \partial x_l}\frac{\delta x_l}{\delta b_i} = 0$$

$$\frac{\partial}{\partial x_k}\left(A_{nmk}\boxed{\frac{\partial U_m}{\partial b_i}}\right) + \frac{\partial^2 f_{nk}^{inv}}{\partial x_k \partial x_l}\frac{\delta x_l}{\delta b_i} = 0$$

… add the pseudo-time derivative and discretize (FV) …

Solid Wall Boundary Conditions :

$$\frac{\delta(u_k n_k)}{\delta b_i} = 0 \Rightarrow \frac{\partial u_k}{\partial b_i}n_k = -\frac{\partial u_k}{\partial x_l}\frac{\delta x_l}{\delta b_i}n_k - u_k\frac{\delta n_k}{\delta b_i}$$

Direct Differentiation (DD) Approach

Example : Inverse Design Problem :

$$F = \frac{1}{2}\int_{S_w}(p-p_{tar})^2 dS$$

$$\frac{\delta F}{\delta b_i} = \int_{S_w}(p-p_{tar})\frac{\delta p}{\delta b_i}dS + \frac{1}{2}\int_{S_w}(p-p_{tar})^2\frac{\delta(dS)}{\delta b_i}$$

F variations with respect to the design variables b

Variations in the flow variables U with respect to b
THE OUTCOME OF DD

Variations in geometrical quantities with respect b

►CPU cost: N Equivalent Flow Solutions (N EFS)

Continuous Adjoint Approach

<u>Example :</u> Compressible Fluid Flow Equations

$$F_{aug} = F + \int_{\Omega} \Psi_n R_n^{inv} d\Omega = F + \int_{\Omega} \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k} d\Omega$$

$$\frac{\delta F_{aug}}{\delta b_i} = \frac{\delta F}{\delta b_i} + \frac{\delta}{\delta b_i} \int_{\Omega} \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k} d\Omega$$

$$\frac{\delta}{\delta b_i} \int_{\Omega} \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k} d\Omega = \int_{\Omega} \frac{\delta \Psi_n}{\delta b_i} \frac{\partial f_{nk}^{inv}}{\partial x_k} d\Omega + \int_{\Omega} \Psi_n \frac{\delta}{\delta b_i} \left(\frac{\partial f_{nk}^{inv}}{\partial x_k} \right) d\Omega + \int_{\Omega} \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k} \frac{\delta (d\Omega)}{\delta b_i}$$

$$\frac{\delta (d\Omega)}{\delta b_i} = \frac{\partial}{\partial x_l} \left(\frac{\delta x_l}{\delta b_i} \right) d\Omega$$

Continuous Adjoint Approach

Final Expression :

$$
\begin{aligned}
\frac{\delta F_{aug}}{\delta b_i} &= \frac{1}{2}\int_{S_w}(p-p_{tar})^2\frac{\delta(dS)}{\delta b_i} + \underbrace{\int_{S_w}(p-p_{tar})\,\frac{\delta p}{\delta b_i}dS}_{SWCR} \\
&- \underbrace{\int_{\Omega} A_{nmk}\frac{\partial \Psi_n}{\partial x_k}\frac{\partial U_m}{\partial b_i}d\Omega}_{FAE} + \underbrace{\int_{S_{I,O}} \Psi_n\frac{\partial f^{inv}_{nk}}{\partial b_i}n_k dS}_{IOBC} + \underbrace{\int_{S_w} \Psi_{k+1}n_k\frac{\delta p}{\delta b_i}dS}_{SWCR} \\
&+ \int_{S_w}\left(\Psi_{k+1}p - \Psi_n f^{inv}_{nk}\right)\frac{\delta\left(n_k dS\right)}{\delta b_i} - \int_{S_w}\Psi_n\frac{\partial f^{inv}_{nk}}{\partial x_l}\frac{\delta x_l}{\delta b_i}n_k dS \\
&+ \int_{S_w}\Psi_n\frac{\partial f^{inv}_{nk}}{\partial x_k}\frac{\delta x_l}{\delta b_i}n_l dS + \int_{\Omega}\left(\frac{\delta \Psi_n}{\delta b_i} - \frac{\partial \Psi_n}{\partial x_l}\frac{\delta x_l}{\delta b_i}\right)\frac{\partial f^{inv}_{nk}}{\partial x_k}d\Omega
\end{aligned}
$$

Field Adjoint Equations :

$$
\frac{\partial \Psi_m}{\partial t} - A_{nmk}\frac{\partial \Psi_n}{\partial x_k} = 0
$$

… add discretize (FV) …

Continuous Adjoint Approach

Sensitivity Derivatives :

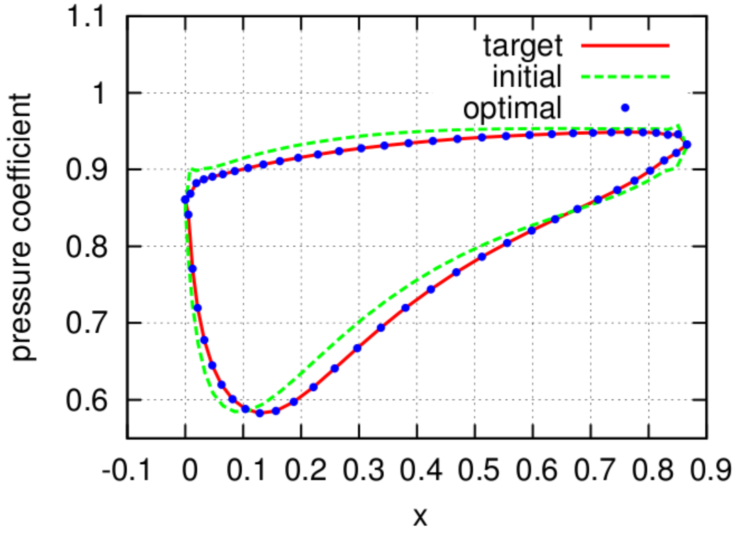
$$
\begin{aligned}
\frac{\delta F_{aug}}{\delta b_i} &= \frac{1}{2}\int_{S_w}(p-p_{tar})^2\frac{\delta(dS)}{\delta b_i} + \int_{S_w}\left(\Psi_{k+1}p-\Psi_n f_{nk}^{inv}\right)\frac{\delta\left(n_k dS\right)}{\delta b_i} \\
&- \int_{S_w}\Psi_n\frac{\partial f_{nk}^{inv}}{\partial x_l}\frac{\delta x_l}{\delta b_i}n_k dS + \int_{S_w}\Psi_n\frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\delta x_l}{\delta b_i}n_l dS
\end{aligned}
$$

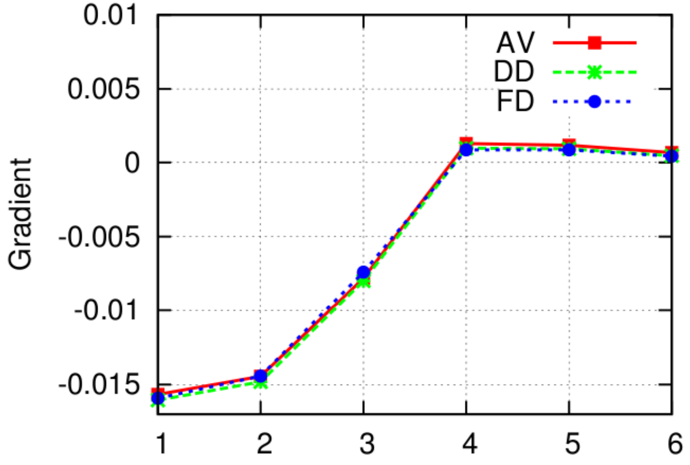
Sensitivity derivatives based exclusively on boundary integrals
(valid, even if the objective function was a field integral !!!)

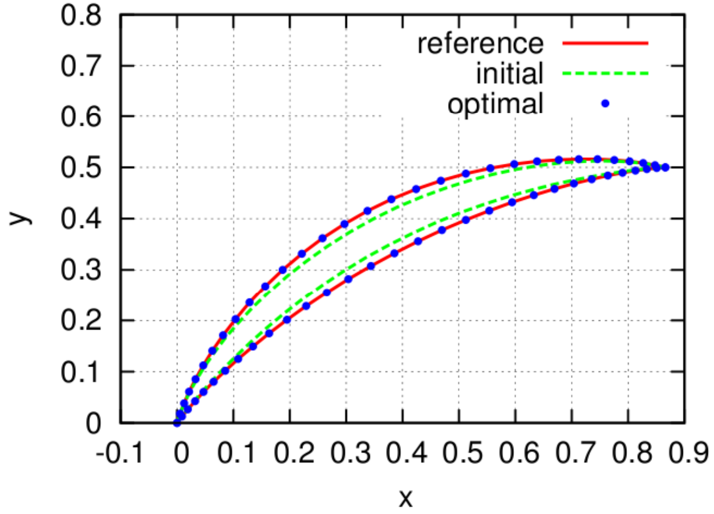
D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'A Continuous Adjoint Method with Objective Function Derivatives Based on Boundary Integrals for Inviscid and Viscous Flows', Computers & Fluids, Vol. 36, pp. 325-341, 2007.

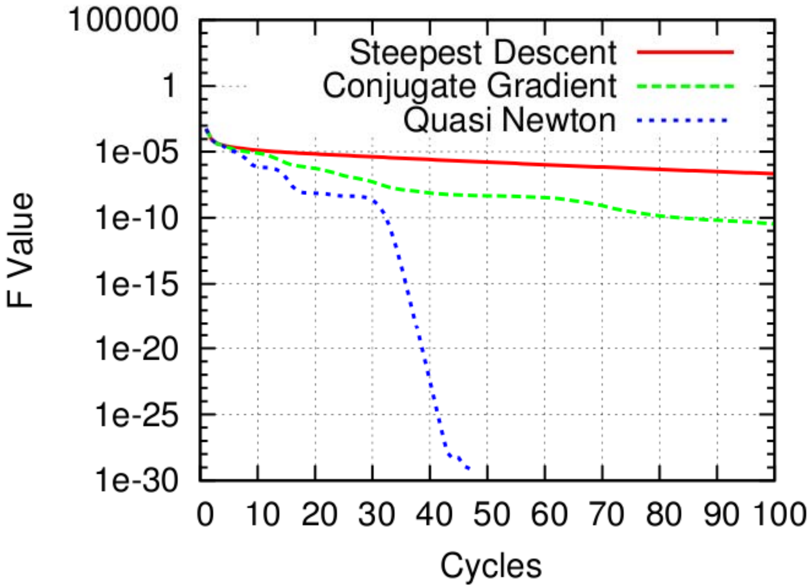
D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'Total Pressure Losses Minimization in Turbomachinery Cascades, Using a New Continuous Adjoint Formulation', Proc. IMechE, Part A: Journal of Power and Energy (Special Issue on Turbomachinery), Vol. 221, pp. 865-872, 2007.

Inverse Design of a 2D Compressor Cascade

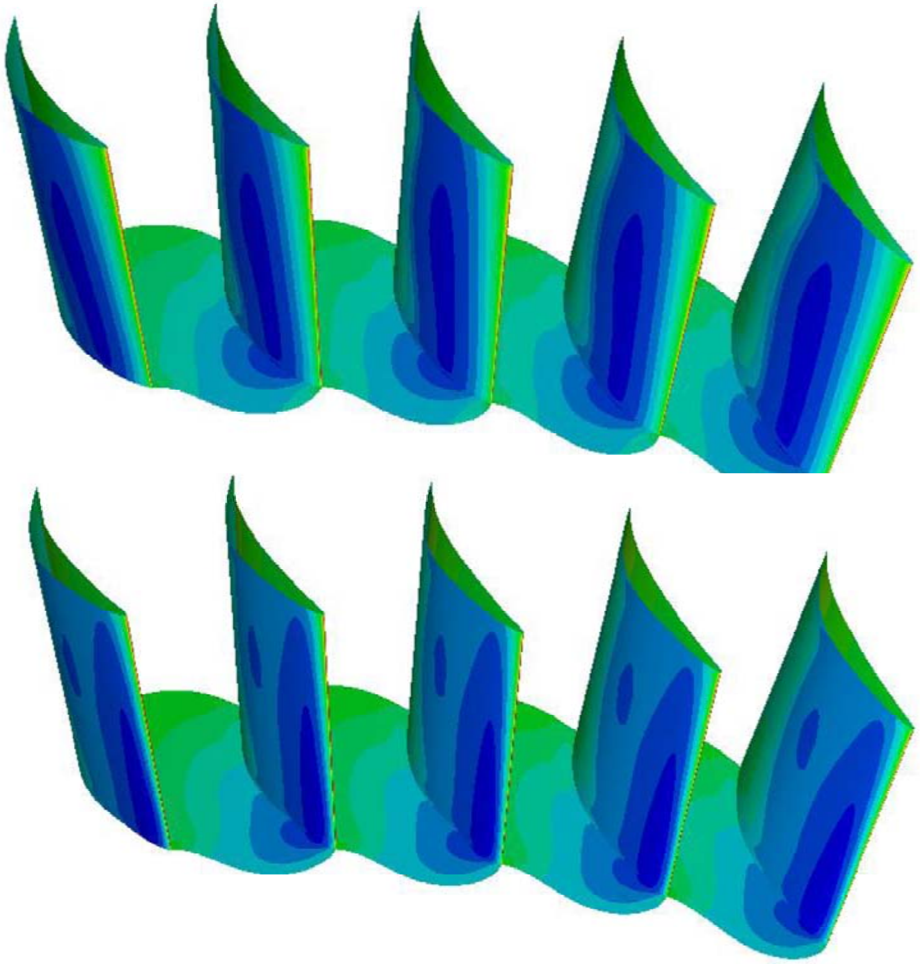


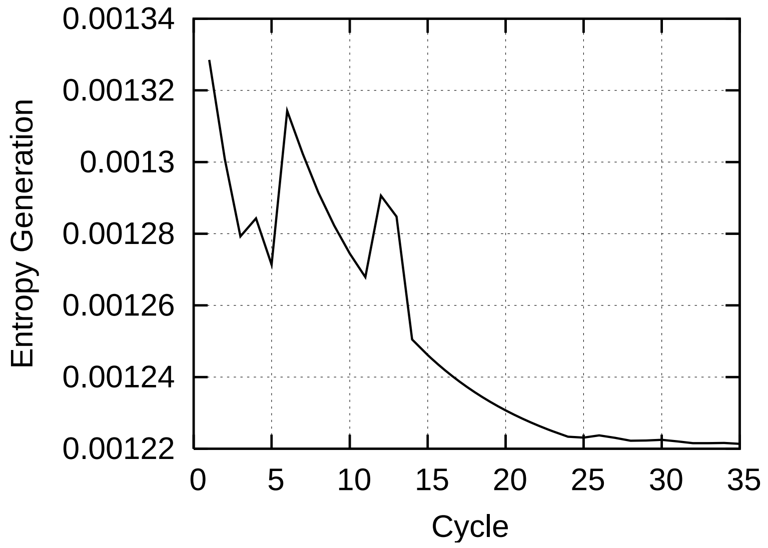






Application: Minimization of Losses





Design-Optimization of a 3D peripheral compressor blade cascade, for minimal viscous losses, with geometrical constraints, using the continuous adjoint method.
Turbulence model : Low-Re Spalart Allmaras

(Continuous) Adjoint Methods for Turbulent Flows

The Adjoint to the Spalart-Allmaras Turbulence Model (for Incompressible Flows)

FUNDED BY:

Technology● Process● Consulting

www.iconCFD.com

Adjoint to the Spalart-Allmaras (SA) Turbulence Model

State Equations : *(Incompressible Fluid Flows)*

$$R^p = \frac{\partial v_j}{\partial x_j} = 0$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] = 0 \ , \qquad i = 1, 2(, 3)$$

$$R^{\widetilde{\nu}} = \frac{\partial (v_j \widetilde{\nu})}{\partial x_j} - \frac{1}{\sigma}\frac{\partial}{\partial x_j}\left[(\nu + \widetilde{\nu})\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\, P\left(\widetilde{\nu}\right) + \widetilde{\nu}\, D\left(\widetilde{\nu}\right) = 0$$

$$\nu_t = \widetilde{\nu} f_{v_1}$$

The "Usual" Assumption (at least in continuous adjoint):

$$\cancel{\frac{\partial \nu_t}{\partial b_m} = 0}$$

Adjoint to the Spalart-Allmaras (SA) Turbulence Model

$$F_{aug} = F + \int_{\Omega} u_i R_i^v d\Omega + \int_{\Omega} q R^p d\Omega + \boxed{\int_{\Omega} \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega}$$

New terms or, even, equations appear!!!!

- **Field Adjoint Equations :**

$$\frac{\partial F_{\Omega}}{\partial v_i} + R^u_{i,v} + R^u_{i,p} + \boxed{R^u_{i,\widetilde{\nu}}} = 0 \quad ★$$

$$\frac{\partial F_{\Omega}}{\partial p} + R^q = 0$$

$$\boxed{\frac{\partial F_{\Omega}}{\partial \widetilde{\nu}} + R_v^{\widetilde{\nu_a}} + R_{\widetilde{\nu}}^{\widetilde{\nu_a}} = 0} \quad ★$$

- **Adjoint BCs :**

$$\frac{\partial F_S}{\partial v_i} + B^u_{i,v} + B^u_{i,p} + \boxed{B^u_{i,\widetilde{\nu}}} = 0 \quad ★$$

$$\frac{\partial F_S}{\partial p} + B^q = 0$$

$$\boxed{\frac{\partial F_S}{\partial \widetilde{\nu}} + B_v^{\widetilde{\nu_a}} + B_{\widetilde{\nu}}^{\widetilde{\nu_a}} = 0} \quad ★$$

- **Sensitivity Derivatives :**

$$\frac{\delta F}{\delta b_m} = SD_v + SD_p \boxed{+ SD_{\widetilde{\nu}}} \quad ★$$

A.S. ZYMARIS, D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU, C. OTHMER: ' Continuous Adjoint Approach to the Spalart-Allmaras Turbulence Model for Incompressible Flows', Computers & Fluids, 38, pp. 1528-1538, 2009.

Adjoint to the Spalart-Allmaras (SA) Turbulence Model

Mean-Flow Field Adjoint Equations :

$$\frac{\partial u_j}{\partial x_j} = -\frac{\partial F_\Omega}{\partial p}$$

$$-v_j\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{\partial}{\partial x_j}\left[(\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]+\frac{\partial q}{\partial x_i}\underbrace{\boxed{-\widetilde{\nu}\frac{\partial \widetilde{\nu_a}}{\partial x_i}-\frac{\partial}{\partial x_l}\left(e_{jli}e_{jmq}\frac{\mathcal{C}_S}{S}\frac{\partial v_q}{\partial x_m}\widetilde{\nu}\widetilde{\nu_a}\right)}}_{Term\ A1}=-\frac{\partial F_\Omega}{\partial v_i}$$

★

Turbulence Model Field Adjoint Equations :

$$\boxed{\begin{aligned}\frac{\partial \widetilde{\nu_a}}{\partial x_j}v_j+\frac{\partial}{\partial x_j}\left[\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu_a}}{\partial x_j}\right] &= \frac{1}{\sigma}\frac{\partial \widetilde{\nu_a}}{\partial x_j}\frac{\partial \widetilde{\nu}}{\partial x_j}+2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu_a}\frac{\partial \widetilde{\nu}}{\partial x_j}\right)+\widetilde{\nu_a}\widetilde{\nu}\mathcal{C}_{\widetilde{\nu}}(\widetilde{\nu},\vec{v})\\ &+\frac{\delta\nu_t}{\delta\widetilde{\nu}}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)+(-P+D)\widetilde{\nu_a}+\frac{\partial F_\Omega}{\partial \widetilde{\nu}}.\end{aligned}}$$

★

Adjoint to the Spalart-Allmaras (SA) Turbulence Model

<u>Adjoint BCs (INLET) :</u>

$$u_{\langle n\rangle} = -\frac{\partial F_{S_I}}{\partial p} \quad , \qquad \mathbf{u}_{\langle t\rangle} = 0 \quad , \qquad \widetilde{\nu_a} = 0$$

<u>Adjoint BCs (OUTLET) :</u>

★

$$q = u_j v_j + u_{\langle n\rangle} v_{\langle n\rangle} + (\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j n_i + \boxed{\underbrace{\widetilde{\nu_a}\widetilde{\nu} - \widetilde{\nu_a}\widetilde{\nu}\frac{C_S}{S} e_{jmq} e_{jli} \frac{\partial v_q}{\partial x_m} n_l n_i}_{\text{Term A2}}} + \frac{\partial F_{S_O}}{\partial v_{\langle n\rangle}}$$

★

$$0 = \mathbf{u}_{\langle t\rangle} v_{\langle n\rangle} + (\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j t_i + \boxed{\underbrace{\widetilde{\nu_a}\widetilde{\nu}\frac{C_S}{S} e_{jmq} e_{jli} \frac{\partial v_q}{\partial x_m} n_l t_i}_{\text{Term A3}}} + \frac{\partial F_{S_O}}{\partial \mathbf{v}_{\langle t\rangle}}$$

★

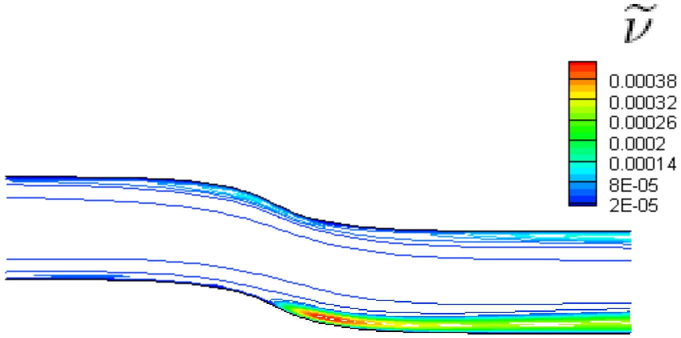
$$\boxed{0 = -\frac{\delta \nu_t}{\delta \widetilde{\nu}} u_i\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right) n_j + \widetilde{\nu_a} v_j n_j + \left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu_a}}{\partial x_j} n_j + \frac{\partial F_{S_O}}{\partial \widetilde{\nu}}}$$

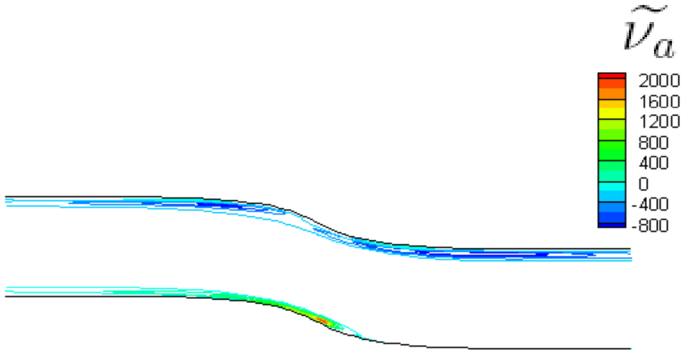
<u>Adjoint BCs (SOLID WALLS) :</u>

$$u_{\langle n\rangle} = -\frac{\partial F_{S_W}}{\partial p} \quad , \qquad \mathbf{u}_{\langle t\rangle} = 0 \quad , \qquad \widetilde{\nu_a} = 0$$

Adjoint to the Spalart-Allmaras (SA) Turbulence Model

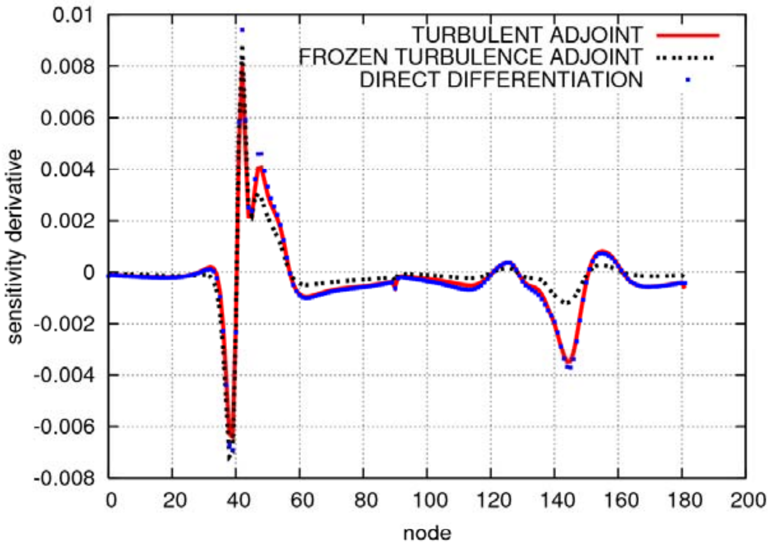
$$\boxed{F = \int_{S_{in}} \rho V_n p_t dS - \int_{S_{out}} \rho V_n p_t dS}$$



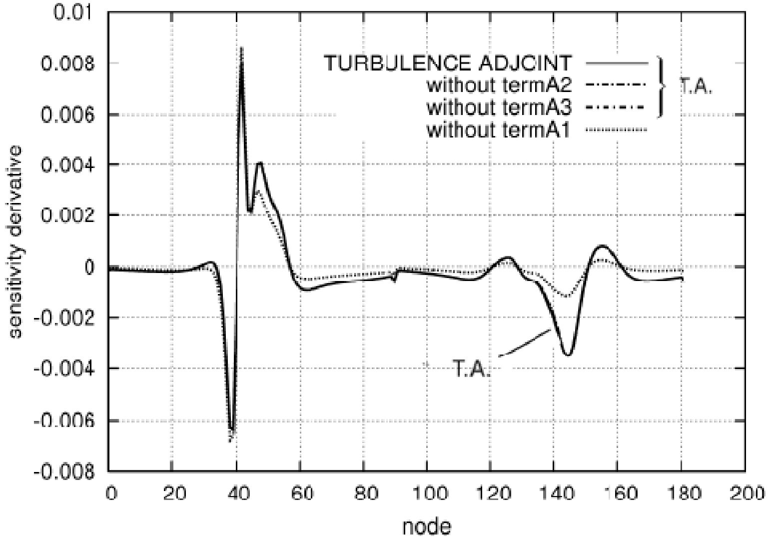


Term (A1), i.e. the new term in the field adjoint momentum eqs. is by far the most important among (A1), (A2) and (A3).

<u>Thus, (A2) and (A3) can safely be neglected.</u>



<u>**Investigations of the roles of (A1), (A2), (A3):**</u>



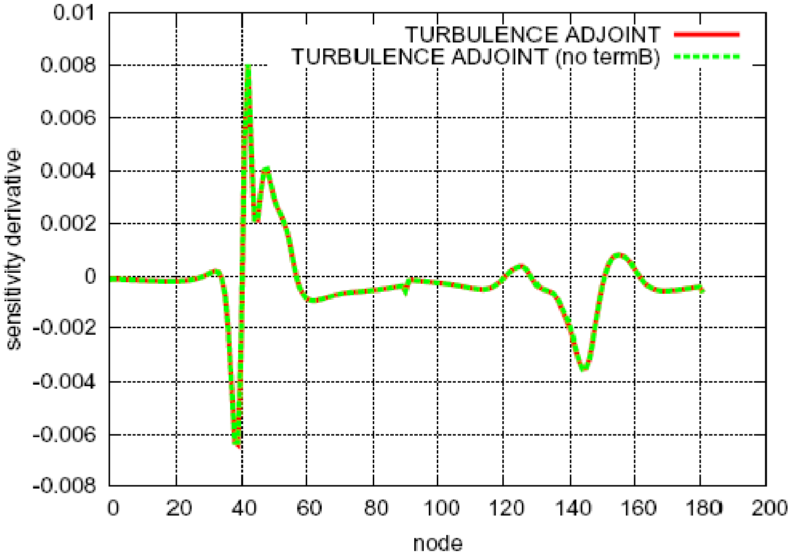
Adjoint to the Spalart-Allmaras (SA) Turbulence Model

$$\frac{\delta F}{\delta b_m}=\int_{S_w}\left(\frac{\partial F_S}{\partial x_k}+F_\Omega n_k\right)\frac{\delta x_k}{\delta b_m}dS+\int_{S_w}F_S\frac{\delta(dS)}{\delta b_m}-\int_{S_w}\left[\nu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j-qn_i\right]\frac{\partial v_i}{\partial x_k}\frac{\delta x_k}{\delta b_m}dS$$

$$\boxed{\underbrace{-\int_{S_w}\nu\frac{\partial\widetilde{\nu}_a}{\partial x_j}n_j\frac{\partial\widetilde{\nu}}{\partial x_k}\frac{\delta x_k}{\delta b_m}dS+\int_{\Omega}\widetilde{\nu}_a\widetilde{\nu}C_d(\widetilde{\nu},\vec{v})\frac{\partial d}{\partial b_m}d\Omega}_{\textit{Term B}}}\ \bigstar$$

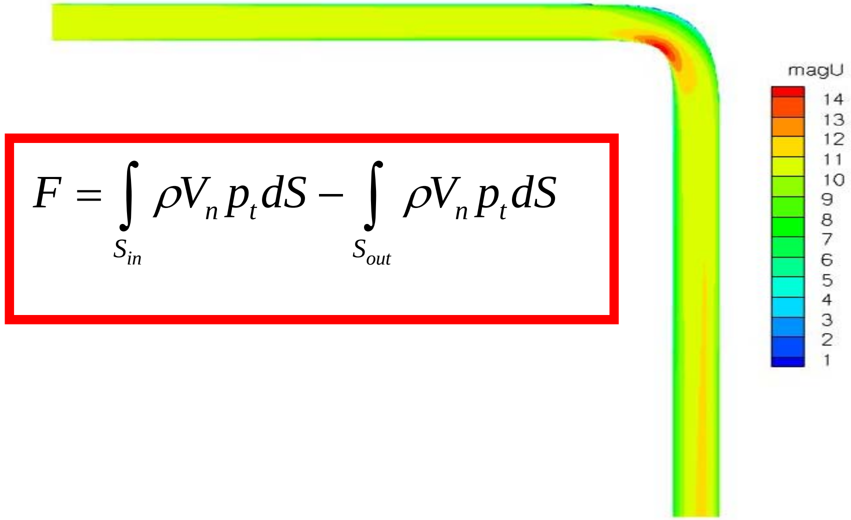
$$+\int_{S_w}u_iR_i^v\frac{\delta x_k}{\delta b_m}n_kdS+\int_{S_w}qR^p\frac{\delta x_k}{\delta b_m}n_kdS$$

Investigation of the role of term(B) :

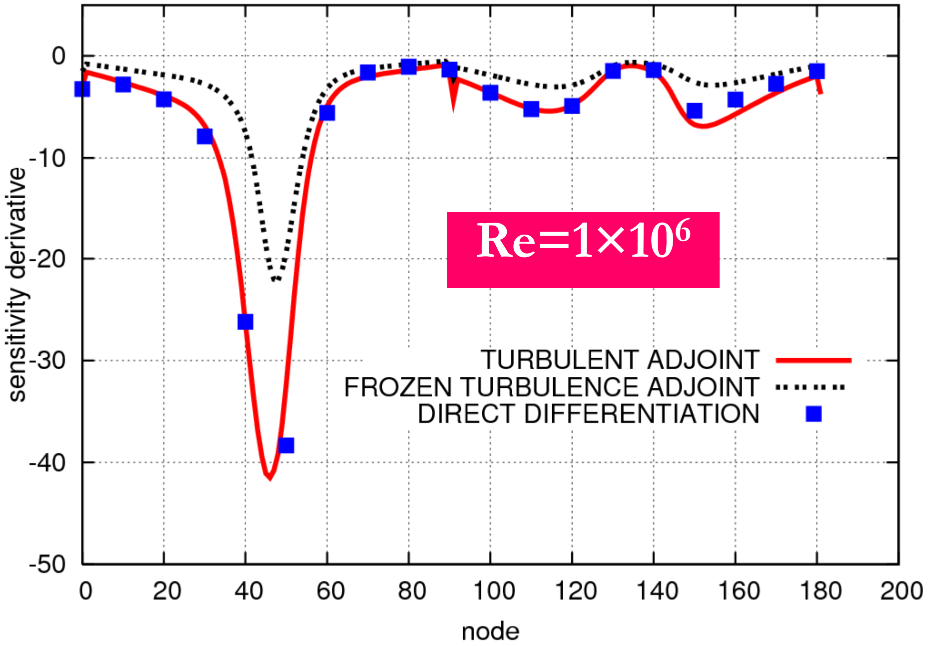


Term (B), or its (computationally intensive) field integral (including DISTANCES FROM THE WALL!!!) in particular, can be neglected.

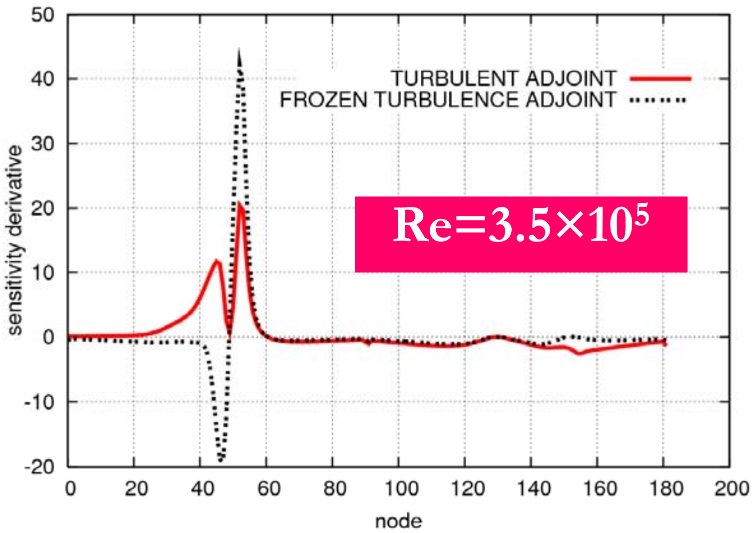
Adjoint to the Spalart-Allmaras (SA) Turbulence Model



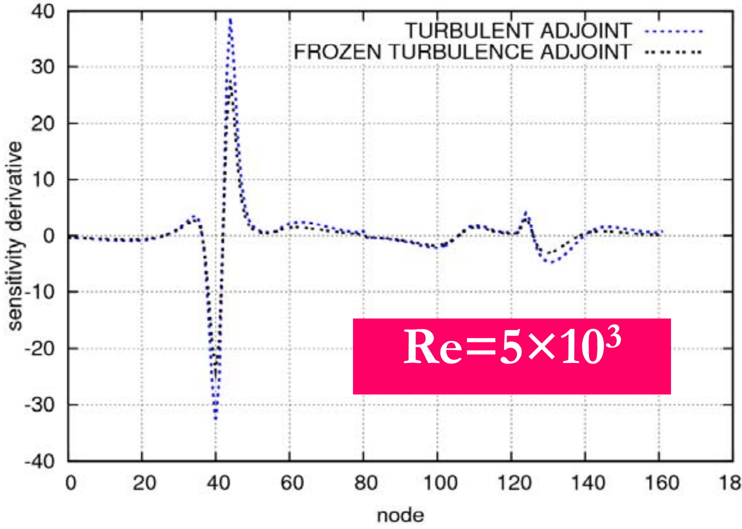
$$F = \int_{S_{in}} \rho V_n p_t dS - \int_{S_{out}} \rho V_n p_t dS$$



Re=1×10^6

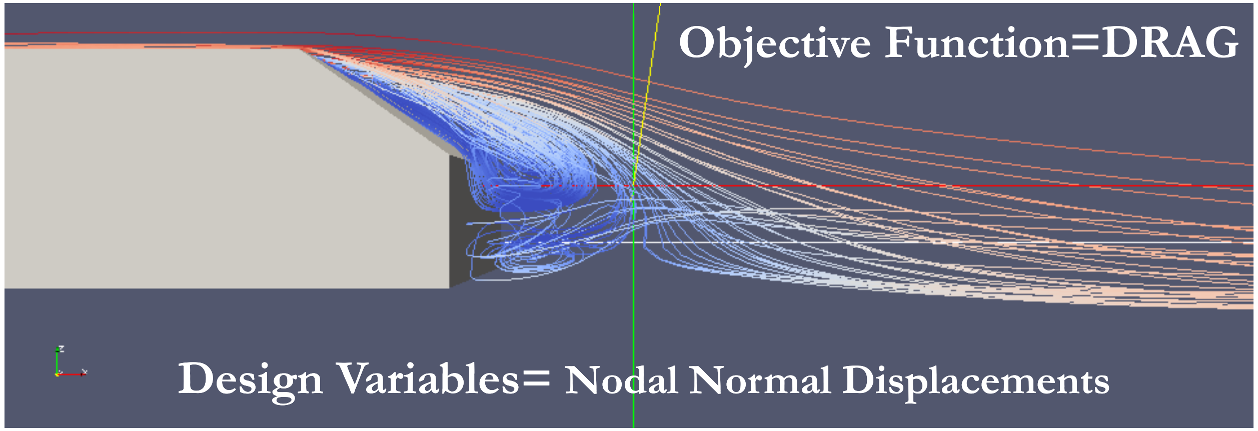


Re=3.5×10^5



Re=5×10^3

Applications

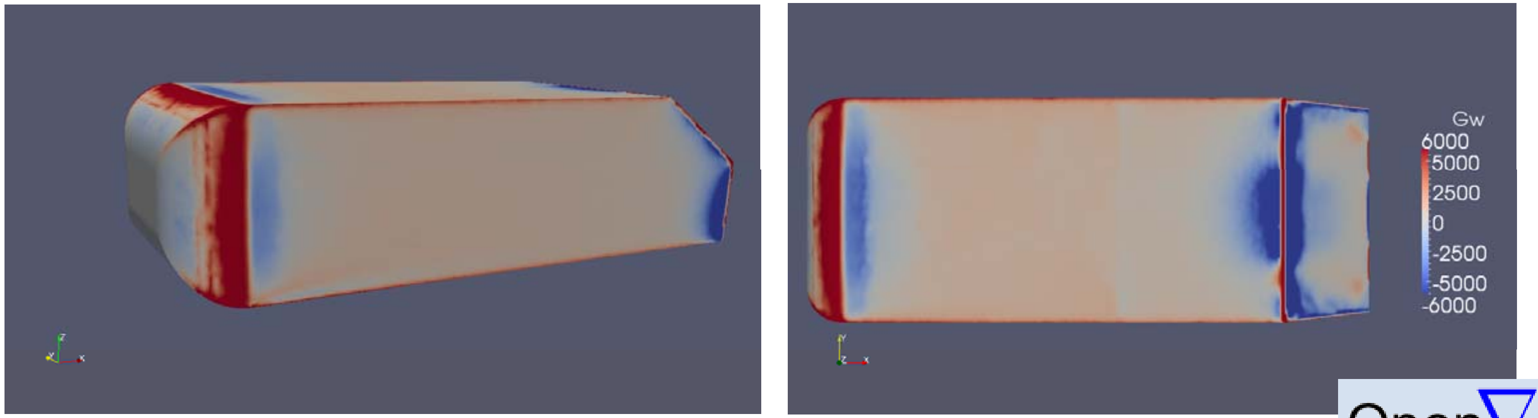


Objective Function=DRAG

Design Variables= Nodal Normal Displacements

The drag sensitivity map shows
beneficial areas for:

- Inward surface movement OR suction jets (red)
- Outward surface movement OR blowing jets (blue)



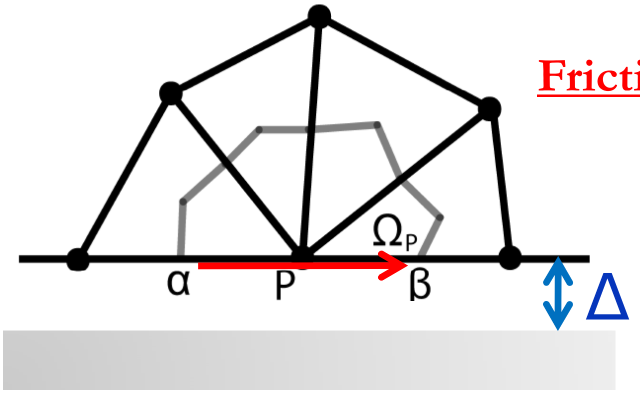
FUNDED BY:

The (Continuous) Adjoint to the High-Re Turbulence Model -

k-ε Model with Wall Functions

Adjoint to High–Re Models – Adjoint Wall Functions

State Equations : *(Incompressible Fluid Flows, k-ε model)*



Friction velocity : $$\upsilon_\tau^2 = (\nu + \nu_t)\left(\frac{\partial \upsilon_i}{\partial x_j} + \frac{\partial \upsilon_j}{\partial x_i}\right) n_j t_i$$

$y^+ < y_c^+$	$y^+ \geq y_c^+$
$\upsilon^+ = y^+$	$\upsilon^+ = \frac{1}{\kappa} ln y^+ + B$
$k_P = \frac{\upsilon_\tau^2}{\sqrt{c_\mu}}\left(\frac{y^+}{y_c^+}\right)^2$	$k_P = \frac{\upsilon_\tau^2}{\sqrt{c_\mu}}$
$\varepsilon_P = k_P^{\frac{3}{2}} \frac{1+\frac{5.3\nu}{\sqrt{k_P \Delta}}}{\kappa c_\mu^{-\frac{3}{4}} \Delta}$	$\varepsilon_P = \frac{\upsilon_\tau^3}{\kappa \Delta}$

Example: **Unstructured grids, finite-volumes, k-ε turbulence model, with slip velocity at the wall**

$$y^+ = \frac{\Delta \upsilon_\tau}{\nu} \qquad \upsilon^+ = \frac{\upsilon_t}{\upsilon_\tau}$$

$$\frac{\delta F_{aug}}{\delta b_m} = \frac{\delta F}{\delta b_m} + \int_\Omega u_i \frac{\delta R_i^v}{\delta b_m} d\Omega + \int_\Omega q \frac{\delta R^p}{\delta b_m} d\Omega + \int_\Omega k_a \frac{\delta R^k}{\delta b_m} d\Omega + \int_\Omega \varepsilon_a \frac{\delta R^\varepsilon}{\delta b_m} d\Omega$$
$$+ \int_\Omega u_i R_i^v \frac{\delta(d\Omega)}{\delta b_m} + \int_\Omega q R^p \frac{\delta(d\Omega)}{\delta b_m} + \int_\Omega k_a R^k \frac{\delta(d\Omega)}{\delta b_m} + \int_\Omega \varepsilon_a R^\varepsilon \frac{\delta(d\Omega)}{\delta b_m}$$

Adjoint to High–Re Models – Adjoint Wall Functions

…

… after satisfying the field adjoint equations and eliminating the field integrals,

it can be shown that

$$\frac{\delta F}{\delta b_m} = \int_{S_w} \underset{\blacktriangle}{E^1} \frac{\delta v_\tau}{\delta b_m} dS + \int_{S_w} E^2 \frac{\delta t_i^l}{\delta b_m} dS + \int_{S_w} E^3 \frac{\delta n_i}{\delta b_m} dS + \int_{S_w} E^4 \frac{\delta x_k}{\delta b_m} dS$$

Definition :

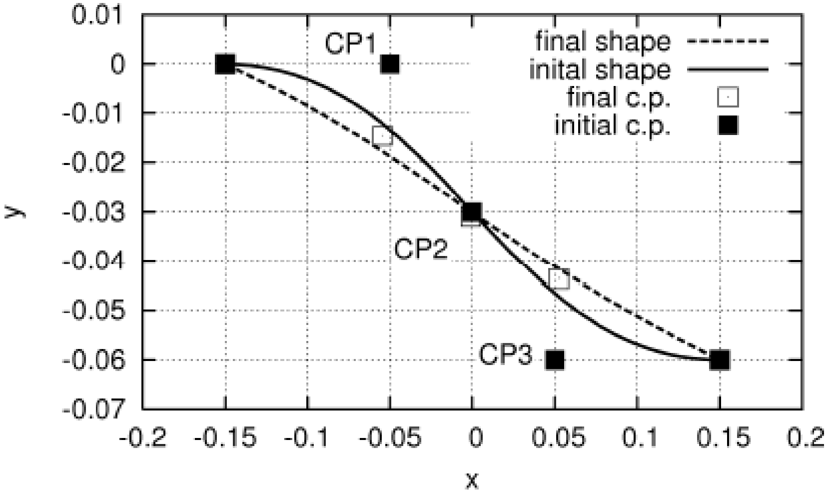
Adjoint friction velocity:

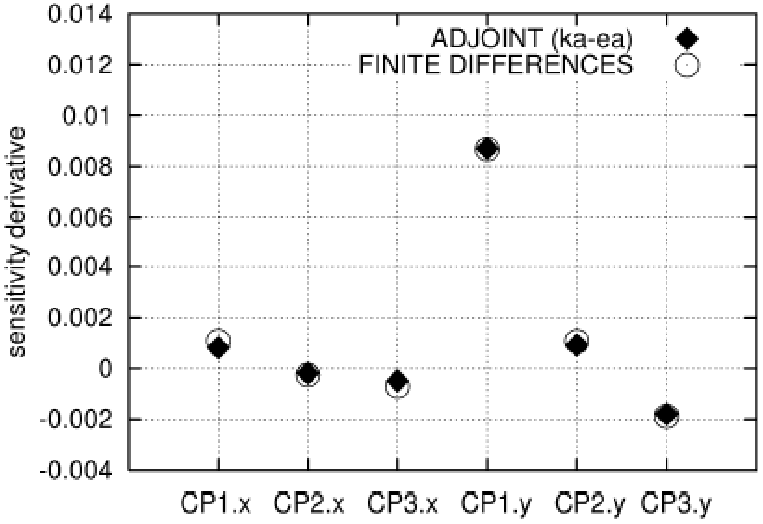
$$u_\tau^2 = (\nu + \nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) n_j t_i$$

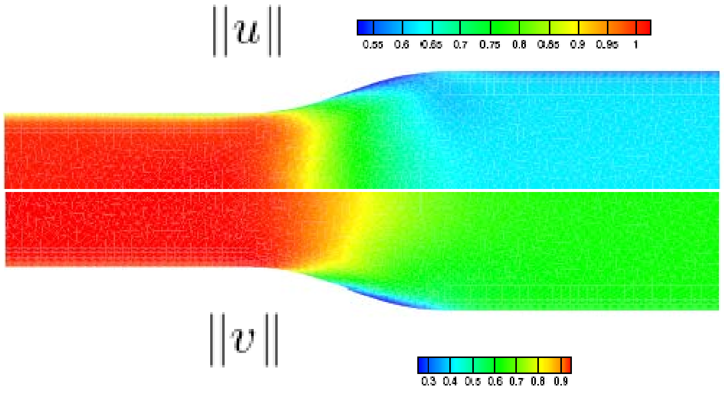
$$u_\tau^2 = \frac{1}{c_v}\left[2u_k t_k v_\tau - \left(\nu + \frac{\nu_t}{Pr_k}\right)\frac{\partial k_a}{\partial x_j} n_j \frac{\delta k}{\delta v_\tau} - \left(\nu + \frac{\nu_t}{Pr_\varepsilon}\right)\frac{\partial \varepsilon_a}{\partial x_j} n_j \frac{\delta \varepsilon}{\delta v_\tau}\right]$$

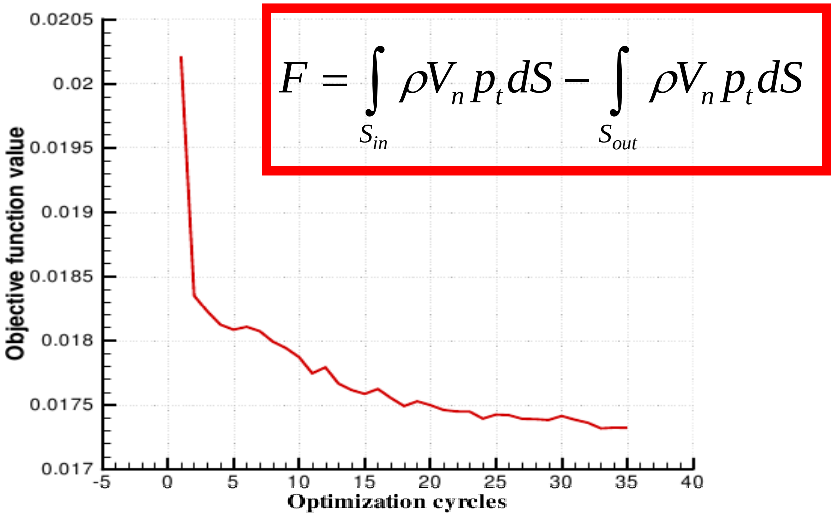
A.S. ZYMARIS, D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU, C. OTHMER: 'Adjoint Wall Functions: A New Concept for Use in Aerodynamic Shape Optimization', Journal of Computational Physics, Vol. 229, pp. 5228–5245 , 2010.

Adjoint to High–Re Models – Adjoint Wall Functions









$$F=\int_{S_{in}}\rho V_n p_t dS-\int_{S_{out}}\rho V_n p_t dS$$

Design of an axial diffuser with minimum total pressure losses (Re=1x10^6)

Applications

(See Presentation by Dr. C. Othmer, VW)

Drag sensitivity map.

FUNDED BY:

NTUA

A.S. Onassis
Foundation

Hessian Matrix Computation,

using the Adjoint Technique

Presentation based on discrete adjoint !!!

Formulation & s/w for both discrete & continuous AV !!!

Hessian Matrix Computation

Newton Methods :

$$b_i^{n+1} = b_i^n + db_i$$

$$\frac{d^2F}{db_i db_j} db_j = -\frac{dF}{db_i}$$

Hessian Matrix Computation

Using Discrete Adjoint :

$$\frac{dF}{db_i} = \frac{\partial F}{\partial b_i} + \frac{\partial F}{\partial U_k}\frac{dU_k}{db_i}$$

k=1,…,N
design variables

$$\frac{d^2F}{db_i db_j} = \frac{\partial^2 F}{\partial b_i \partial b_j} + \frac{\partial^2 F}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 F}{\partial U_k \partial b_j}\frac{dU_k}{db_i} + \frac{\partial^2 F}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \frac{\partial F}{\partial U_k}\frac{d^2U_k}{db_i db_j}$$

Direct Differentiation (DD)

$$\frac{dR_m}{db_i} = \frac{\partial R_m}{\partial b_i} + \frac{\partial R_m}{\partial U_k}\frac{dU_k}{db_i} = 0$$

m=1,…,M
nodes x eqs.

$$\frac{d^2R_n}{db_i db_j} = \frac{\partial^2 R_n}{\partial b_i \partial b_j} + \frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i} + \frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \frac{\partial R_n}{\partial U_k}\frac{d^2U_k}{db_i db_j} = 0$$

Computation of the Hessian Matrix

The DD-AV Scheme:

$$\frac{dR_m}{db_i} = \frac{\partial R_m}{\partial b_i} + \frac{\partial R_m}{\partial U_k}\frac{dU_k}{db_i} = 0 \quad \Rightarrow \quad \boxed{\frac{dU_k}{db_i}} \qquad \boxed{N}$$

System solutions (EFS)

$$\frac{\partial F}{\partial U_k} + \hat{\Psi}_n \frac{\partial R_n}{\partial U_k} = 0 \quad \Rightarrow \quad \boxed{\hat{\Psi}_n} \qquad \boxed{1}$$

EFS

$$\begin{aligned}
\frac{d^2\hat{F}}{db_i db_j} = {} & \frac{\partial^2 F}{\partial b_i \partial b_j} + \hat{\Psi}_n \frac{\partial^2 R_n}{\partial b_i \partial b_j} + \frac{\partial^2 F}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \hat{\Psi}_n \frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} \\
& + \frac{\partial^2 F}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \hat{\Psi}_n \frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 F}{\partial U_k \partial b_j}\frac{dU_k}{db_i} + \hat{\Psi}_n \frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i} \\
& + \cancel{\left(\frac{\partial F}{\partial U_k} + \hat{\Psi}_n \frac{\partial R_n}{\partial U_k}\right)} \frac{d^2 U_k}{db_i db_j}
\end{aligned}$$

$$\frac{d^2 F^\lambda}{db_i db_j} db_j^\lambda = -\frac{dF^\lambda}{db_i} \quad \Rightarrow \quad \boxed{N+1}$$

EFS

Computation of the Hessian Matrix

D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'Direct, Adjoint and Mixed Approaches for the Computation of Hessian in Airfoil Design Problems', International Journal for Numerical Methods in Fluids, Vol. 56, pp. 1929-1943, 2008.

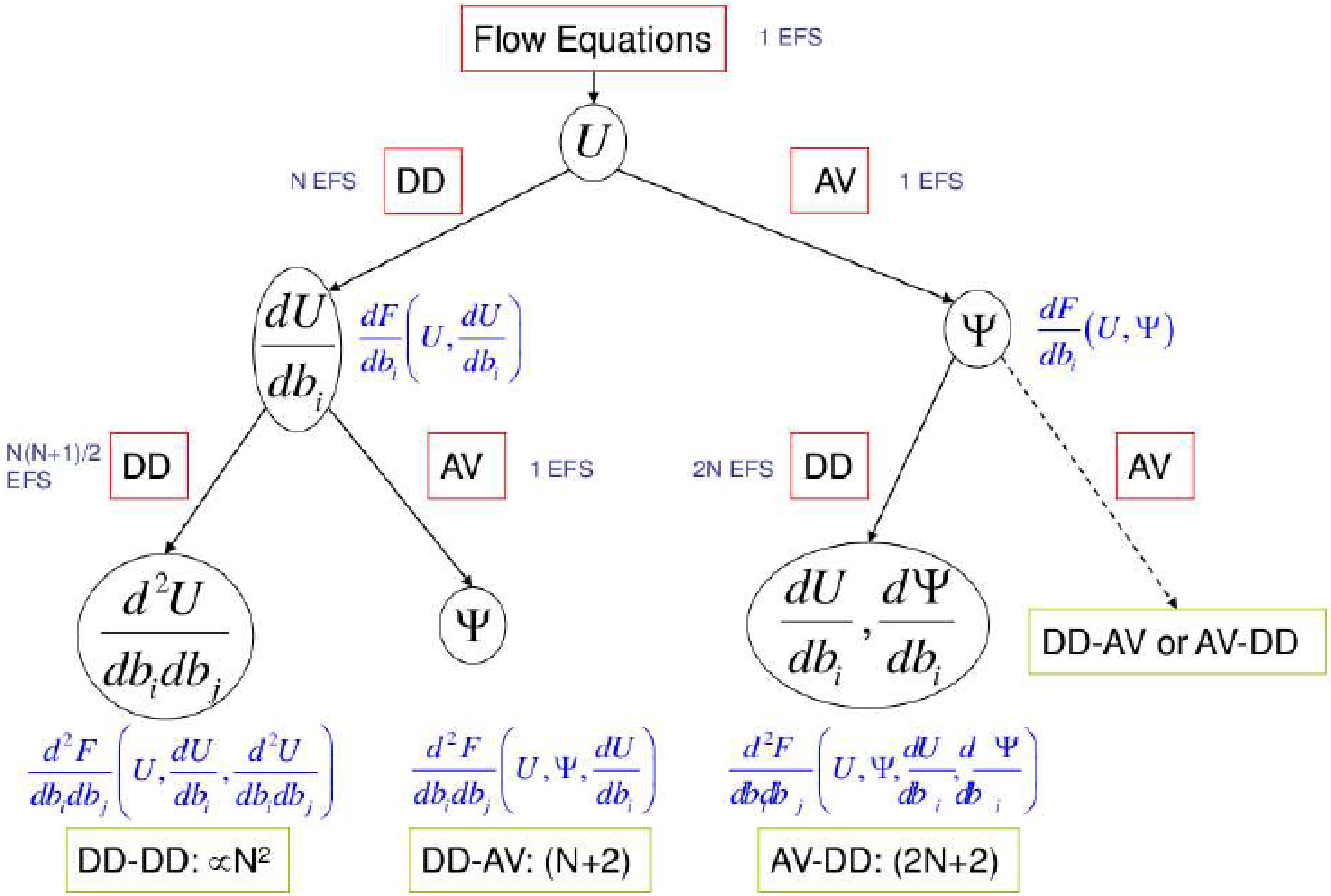
D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'Computation of the Hessian Matrix in Aerodynamic Inverse Design using Continuous Adjoint Formulations', Computers & Fluids, Vol. 37, pp. 1029-1039, 2008.

K.C. GIANNAKOGLOU, D.I. PAPADIMITRIOU: 'Adjoint Methods for gradient- and Hessian-based Aerodynamic Shape Optimization', EUROGEN 2007, Jyvaskyla, Finland, June 11-13, 2007.

D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'Aerodynamic Shape Optimization using Adjoint and Direct Approaches', Archives of Computational Methods in Engineering (State of the Art Reviews), Vol. 15(4), pp. 447-488, 2008

D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'The Continuous Direct-Adjoint Approach for Second Order Sensitivities in Viscous Aerodynamic Inverse Design Problems', Computers & Fluids, 38, pp. 1539-1548, 2009.

Computation of the Hessian Matrix – Alternative Ways



Flow Equations 1 EFS

U

N EFS DD

AV 1 EFS

$\frac{dU}{db_i}$ $\frac{dF}{db_i}\left(U,\frac{dU}{db_i}\right)$

Ψ $\frac{dF}{db_i}(U,\Psi)$

N(N+1)/2 EFS DD

AV 1 EFS

2N EFS DD

AV

$\frac{d^2U}{db_idb_j}$

Ψ

$\frac{dU}{db_i},\frac{d\Psi}{db_i}$

DD-AV or AV-DD

$\frac{d^2F}{db_idb_j}\left(U,\frac{dU}{db_i},\frac{d^2U}{db_idb_j}\right)$

DD-DD: $\propto N^2$

$\frac{d^2F}{db_idb_j}\left(U,\Psi,\frac{dU}{db_i}\right)$

DD-AV: (N+2)

$\frac{d^2F}{db_idb_j}\left(U,\Psi,\frac{dU}{db_i},\frac{d\Psi}{db_i}\right)$

AV-DD: (2N+2)

Computation of the Hessian Matrix

Using Continuous Adjoint – The DD-AV Scheme:

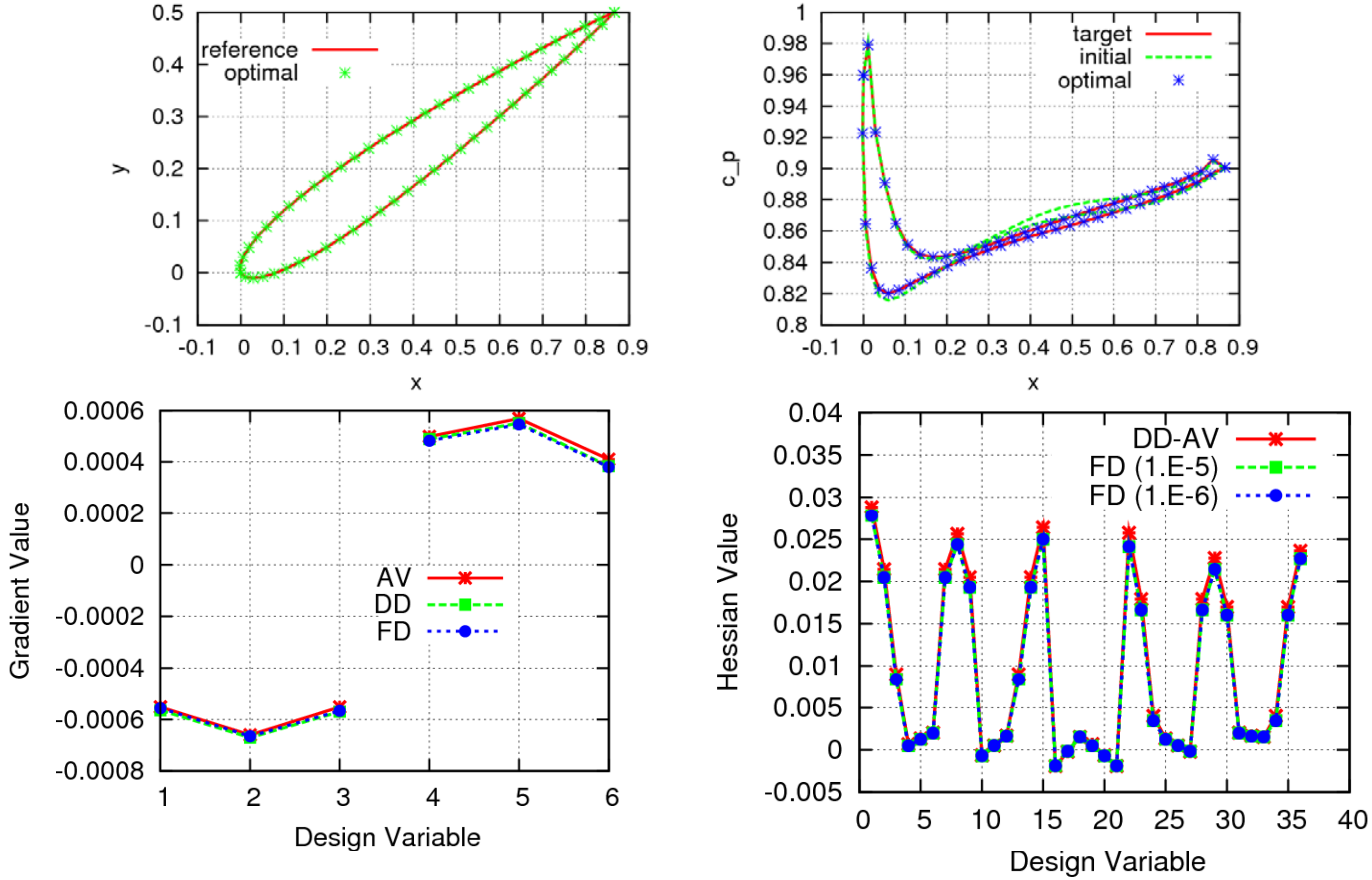
$$\frac{\delta F_{aug}}{\delta b_j} = \frac{\delta F}{\delta b_j} + \int_\Omega \Psi_n \frac{\partial}{\partial b_j}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right) d\Omega + \int_S \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\delta x_l}{\delta b_j} n_l dS + \int_\Omega \frac{\partial \Psi_n}{\partial b_j}\frac{\partial f_{nk}^{inv}}{\partial x_k} d\Omega$$

…

Sensitivity Derivatives :

$$\begin{aligned}
\frac{\delta^2 F_{aug}}{\delta b_i \delta b_j} &= \int_{S_w} \frac{\delta p}{\delta b_i}\frac{\delta p}{\delta b_j} dS + \int_{S_w} (p-p_{tar}) \frac{\delta p}{\delta b_i}\frac{\delta (dS)}{\delta b_j} + \int_{S_w} (p-p_{tar}) \frac{\delta p}{\delta b_j}\frac{\delta (dS)}{\delta b_i} \\
&+ \frac{1}{2}\int_{S_w} (p-p_{tar})^2 \frac{\delta^2 (dS)}{\delta b_i \delta b_j} + \int_{S_w} (\Psi_{k+1} p - \Psi_n f_{nk}^{inv}) \frac{\delta^2 n_k}{\delta b_i \delta b_j} dS \\
&+ \int_{S_w} \left(\Psi_{k+1}\frac{\delta p}{\delta b_i} - \Psi_n \frac{\delta f_{nk}^{inv}}{\delta b_i}\right)\frac{\delta n_k}{\delta b_j} dS + \int_{S_w} \left(\Psi_{k+1}\frac{\delta p}{\delta b_j} - \Psi_n \frac{\delta f_{nk}^{inv}}{\delta b_j}\right)\frac{\delta n_k}{\delta b_i} dS \\
&+ \int_\Omega \frac{\partial A_{nmk}}{\partial U_l}\frac{\partial U_m}{\partial b_i}\frac{\partial U_l}{\partial b_j}\frac{\partial \Psi_n}{\partial x_k} d\Omega \\
&- \int_{S_w} \Psi_n \left(\frac{\partial^2 f_{nk}}{\partial b_i \partial x_l}\frac{\delta x_l}{\delta b_j} + \frac{\partial^2 f_{nk}}{\partial b_j \partial x_l}\frac{\delta x_l}{\delta b_i} + \frac{\partial^2 f_{nk}}{\partial x_l \partial x_m}\frac{\delta x_l}{\delta b_i}\frac{\delta x_m}{\delta b_j} + \frac{\partial f_{nk}}{\partial x_l}\frac{\delta^2 x_l}{\delta b_i \delta b_j}\right) n_k dS \\
&+ \int_S \Psi_n \frac{\partial}{\partial b_i}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right)\frac{\delta x_l}{\delta b_j} n_l dS + \int_S \Psi_n \frac{\partial}{\partial b_j}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right)\frac{\delta x_l}{\delta b_i} n_l dS \\
&+ \int_S \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\delta^2 x_l}{\delta b_i \delta b_j} n_l dS + \int_S \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\partial}{\partial x_m}\left(\frac{\delta x_m}{\delta b_i}\right)\frac{\delta x_l}{\delta b_j} n_l dS \\
&+ \int_S \frac{\partial \Psi_n}{\partial x_m}\frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\delta x_m}{\delta b_i}\frac{\delta x_l}{\delta b_j} n_l dS - \int_S \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_k}\frac{\partial}{\partial x_m}\left(\frac{\delta x_l}{\delta b_i}\right)\frac{\delta x_m}{\delta b_j} n_l dS \\
&+ \int_S \Psi_n \frac{\partial^2 f_{nk}^{inv}}{\partial x_k \partial x_l}\frac{\delta x_l}{\delta b_i}\frac{\delta x_m}{\delta b_j} n_m dS
\end{aligned}$$

Accuracy of the Computed Gradient/Hessian

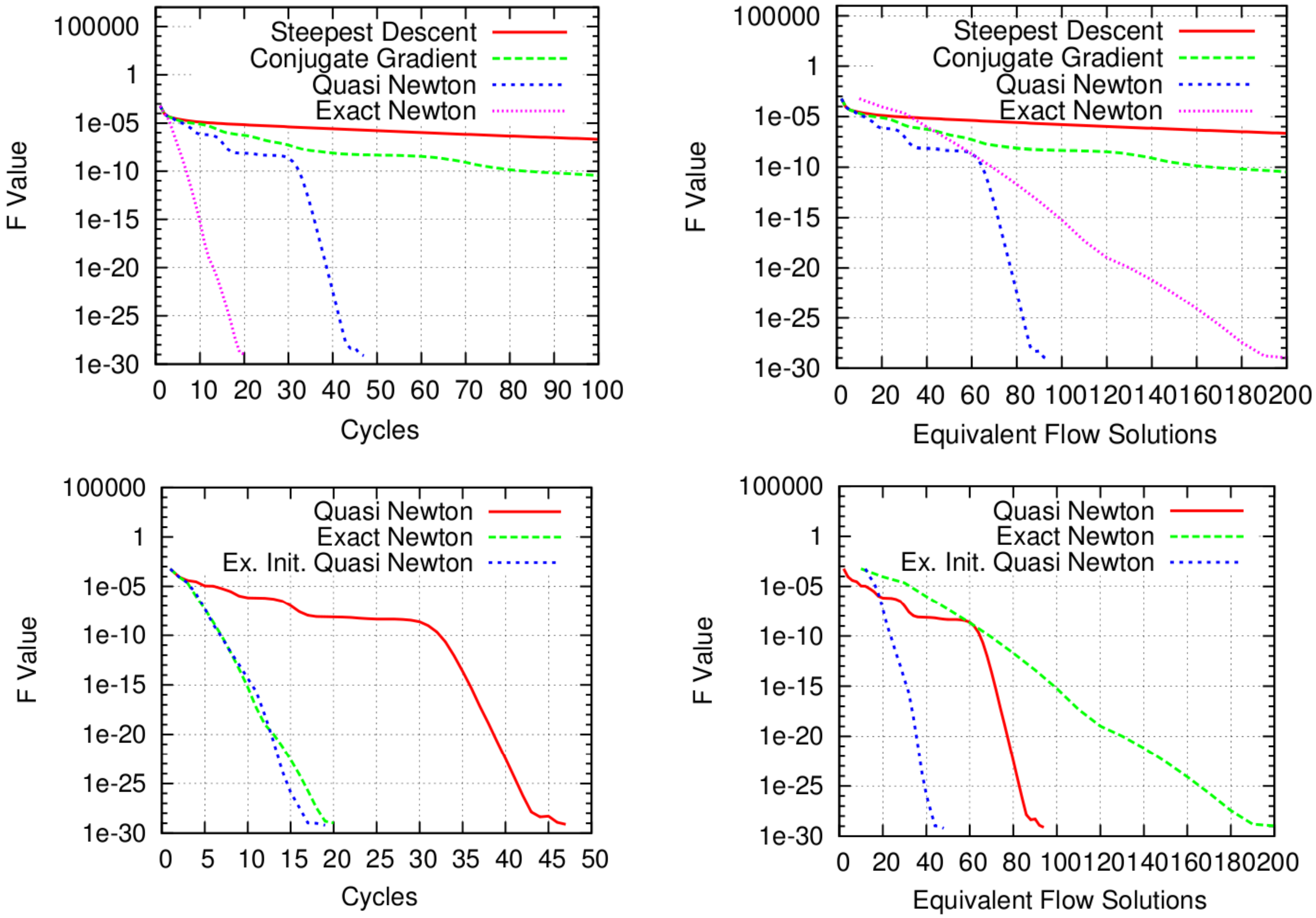


Inverse Design of a Compressor Cascade, Inviscid Flow (6 design variables)

$\frac{5\times6}{2}\times 4 = 60$ vs. $6+1=7$ EFS

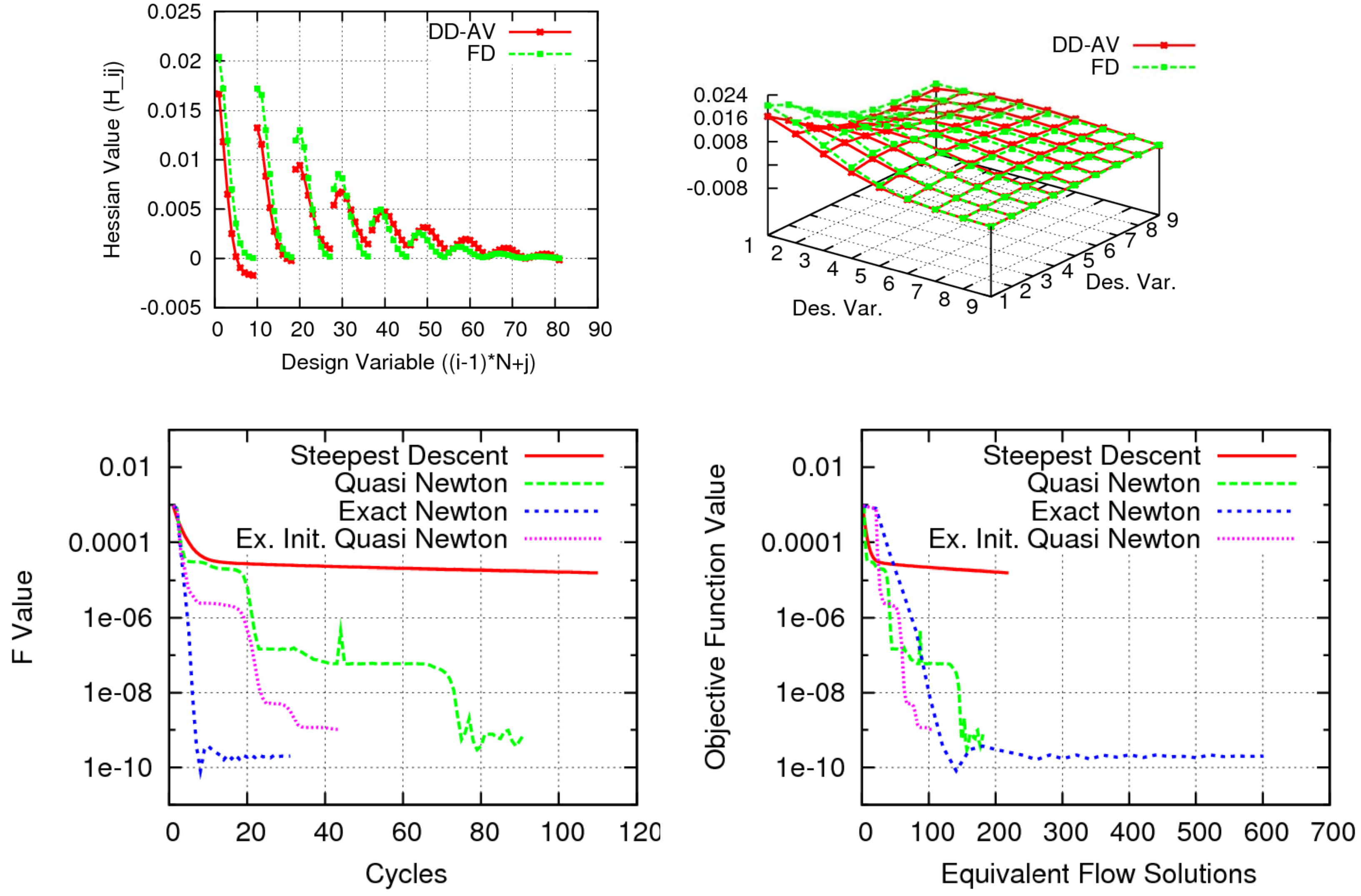
(FD) (DD-AV)

Newton methods: About CPU Cost



Inverse Design of a Compressor Cascade, Inviscid Flow

Computation of the Hessian Matrix

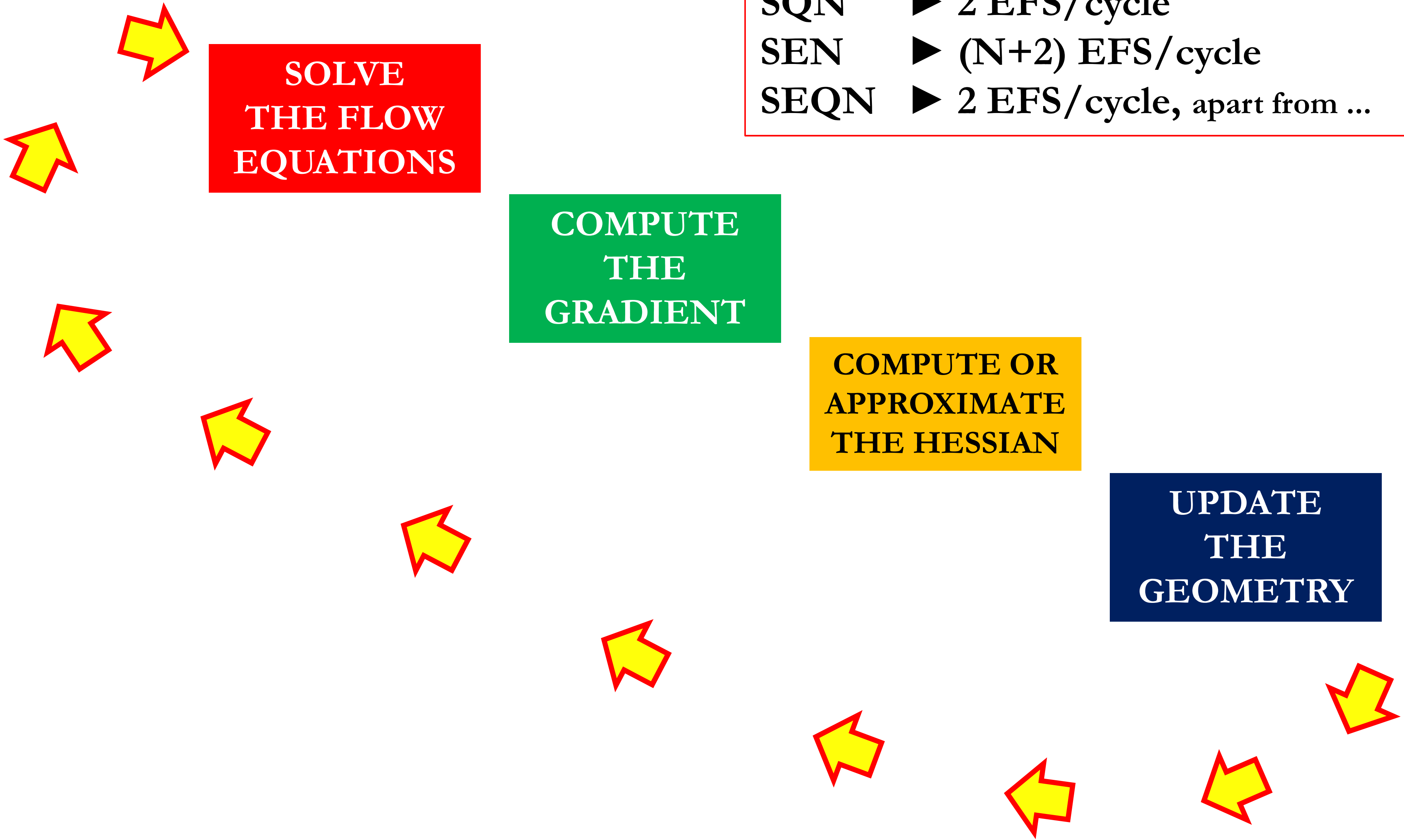


Inverse Design of a Plane Diffuser, Viscous Flow

One-Shot Newton Methods

Newton Methods – Segregated (S) Variants

SSD	▶ 2 EFS/cycle
SQN	▶ 2 EFS/cycle
SEN	▶ (N+2) EFS/cycle
SEQN	▶ 2 EFS/cycle, apart from ...



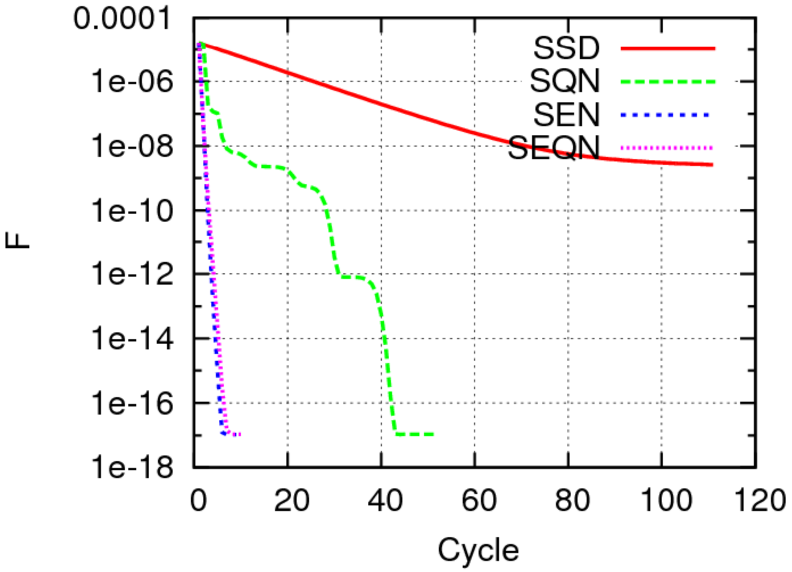
SOLVE THE FLOW EQUATIONS

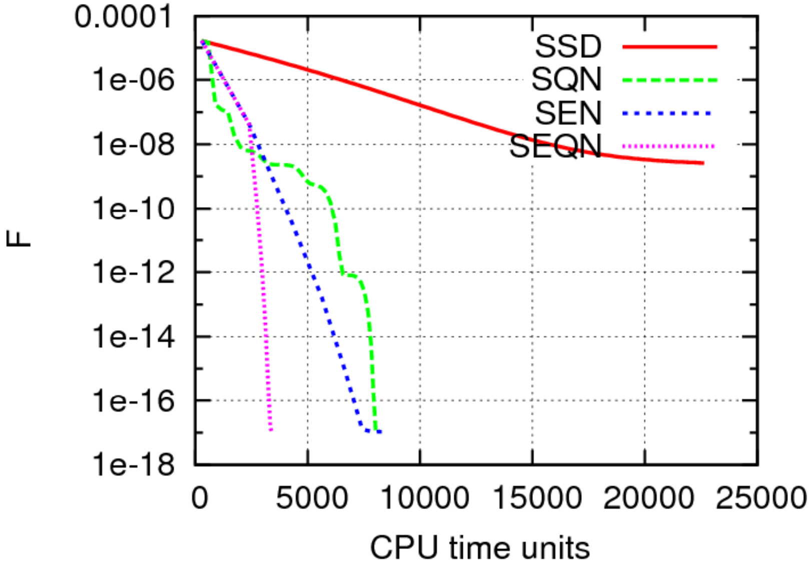
COMPUTE THE GRADIENT

COMPUTE OR APPROXIMATE THE HESSIAN

UPDATE THE GEOMETRY

Segregated Variants – CPU time comparison





SSD ► **Segregated Steepest Descent**

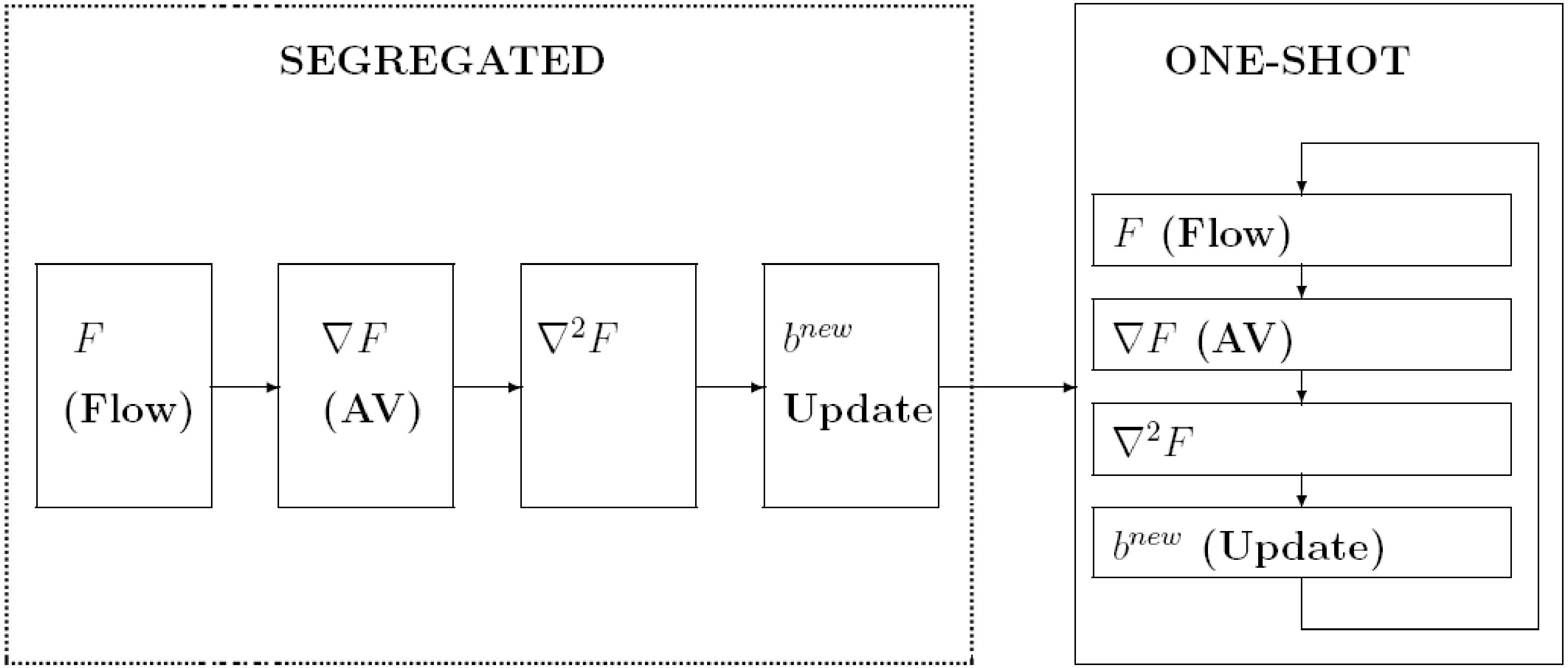
SQN ► **Segregated Quasi-Newton (BFGS)**

SEN ► **Segregated Exact Newton**

SEQN ► **Segregated Exact(*first cycle*)-Quasi(*then*) Newton**

Design of a Compressor Cascade

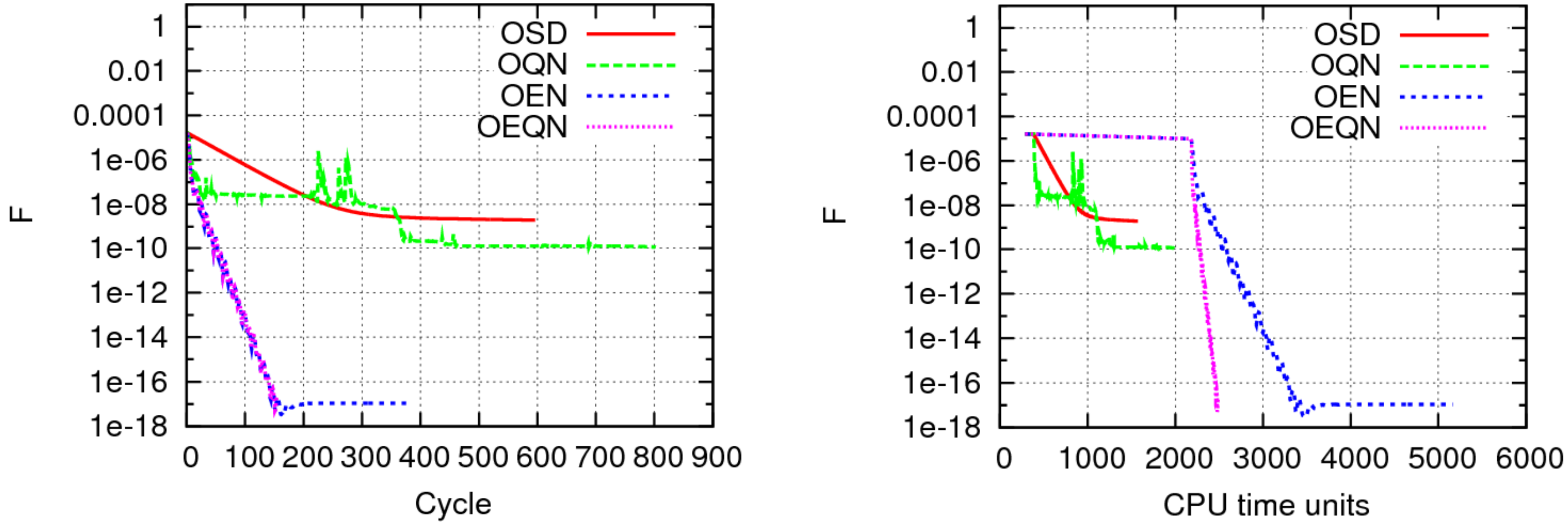
Newton Methods – One-Shot (O) Variants



OSD ► **One-Shot Steepest Descent**
OQN ► **One-Shot Quasi-Newton (BFGS)**
OEN ► **One-Shot Exact Newton**
OEQN ► **One-Shot Exact(*first cycle*)-Quasi(*then*) Newton**

Design of a Compressor Cascade

One-Shot Variants – CPU time comparison



OSD ► **One-Shot Steepest Descent**

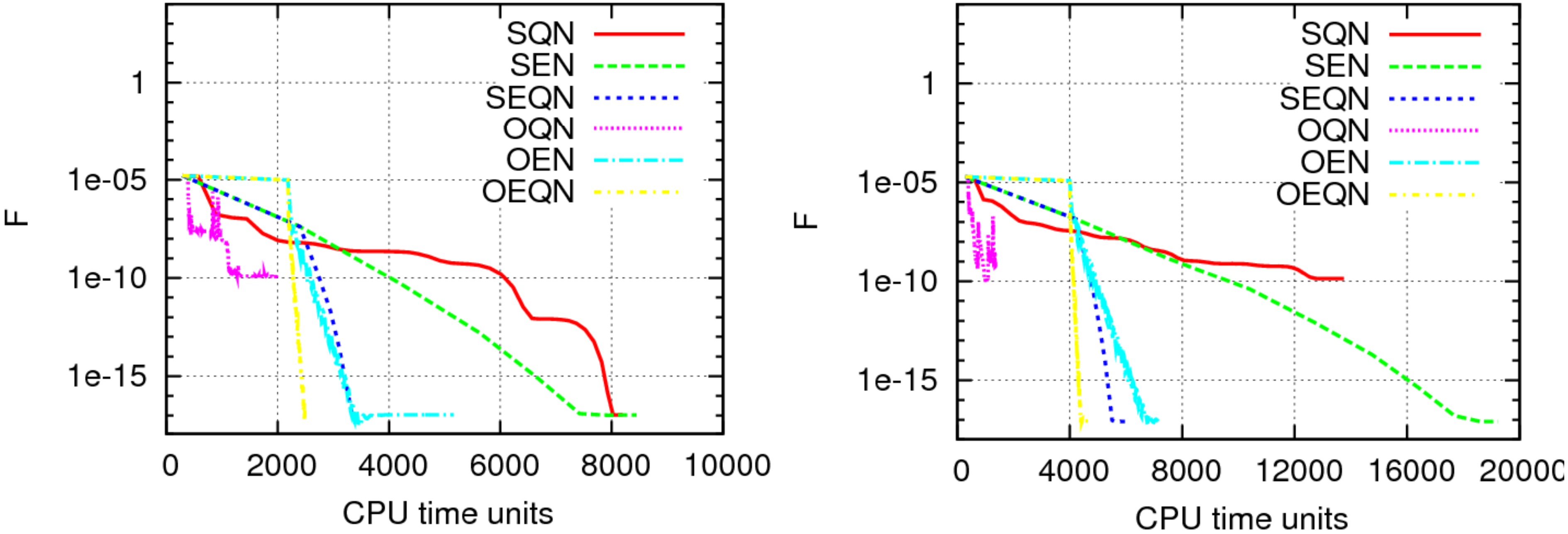
OQN ► **One-Shot Quasi-Newton (BFGS)**

OEN ► **One-Shot Exact Newton**

OEQN ► **One-Shot Exact(*first cycle*)-Quasi(*then*) Newton**

Design of a Compressor Cascade

One-Shot Variants – Overall CPU time comparison



Design of a Compressor Cascade, 6 (Left) & 12 (Right) Design Variables

D.I. PAPADIMITRIOU, K.C. GIANNAKOGLOU: 'One-Shot Shape Optimization Using the Exact Hessian', ECCOMAS CFD 2010, 5th European Conference on CFD, Lisbon, Portugal, June 14-17, 2010.

The Adjoint-Based Truncated Newton Algorithm for Large-Scale Optimization Problems

The Conjugate Gradient method for linear problems

$$\boxed{Ax = q}$$

```
$k \leftarrow 0$
$x \leftarrow \text{init}()$
$r^0 \leftarrow \boxed{Ax} - q;\ p \leftarrow -r^0$
while $r^k \neq 0$ and $k \leq M_{CG}$ do
    $\eta \leftarrow \frac{(r^k)^T r^k}{p^T \boxed{Ap}}$
    $x \leftarrow x + \eta p$
    $r^{k+1} \leftarrow r^k + \eta \boxed{Ap}$
    $\beta \leftarrow \frac{(r^{k+1})^T r^{k+1}}{(r^k)^T r^k}$
    $p \leftarrow -r_{k+1} + \beta p$
    $k \leftarrow k + 1$
end while
```

$$b_i^{n+1} = b_i^n + db_i$$

$$\frac{d^2F}{db_i db_j} db_j = -\frac{dF}{db_i}$$

The AV-DD Approach for Truncated Newton

Compute $\frac{d^2F}{db_i db_j}s_j$

$$\frac{dF}{db_i}=\frac{\partial F}{\partial b_i}+\Psi_m\frac{\partial R_m}{\partial b_i} \quad \Longrightarrow$$

$$\frac{d^2F}{db_i db_j}s_j=\frac{\partial^2 F}{\partial b_i \partial b_j}s_j+\frac{\partial^2 F}{\partial b_i \partial U_k}\left(\frac{dU_k}{db_j}s_j\right)+\underset{\blacktriangle}{\Psi_m}\frac{\partial^2 R_m}{\partial b_i \partial b_j}s_j+$$
$$+\Psi_m\frac{\partial^2 R_m}{\partial b_i \partial U_k}\underset{\blacktriangle}{\left(\frac{dU_k}{db_j}s_j\right)}+\underset{\blacktriangle}{\left(\frac{d\Psi_m}{db_j}s_j\right)}\frac{\partial R_m}{\partial b_i} \qquad (1)$$

▶ $$\frac{\partial F}{\partial U_k}+\underline{\Psi_m}\frac{\partial R_m}{\partial U_k}=0 \qquad (2)$$

▶ $$\frac{\partial R_m}{\partial b_j}s_j+\frac{\partial R_m}{\partial U_k}\underline{\left(\frac{dU_k}{db_j}s_j\right)}=0 \qquad (3)$$

▶ $$\frac{\partial^2 F}{\partial U_n \partial b_j}s_j+\frac{\partial^2 F}{\partial U_n \partial U_k}\left(\frac{dU_k}{db_j}s_j\right)+\Psi_m\frac{\partial^2 R_m}{\partial U_n \partial b_j}s_j+$$
$$+\Psi_m\frac{\partial^2 R_m}{\partial U_n \partial U_k}\left(\frac{dU_k}{db_j}s_j\right)+\underline{\left(\frac{d\Psi_m}{db_j}s_j\right)}\frac{\partial R_m}{\partial U_n}=0 \qquad (4)$$

The AV-DD Truncated Newton Method (with CG)

```
$k \leftarrow 0$
$b_j \leftarrow \text{init}()$
while $k \leq k_{max}$ do
    $U_n \leftarrow$ Flow Equations [1 EFS] ★
    $\Psi_n \leftarrow$ Adjoint Equations [1 EFS] ★
    $r_j^0 = \frac{dF}{db_j} \leftarrow$ Gradient Expression
    $db_j^0 \leftarrow \text{init}(0)$
    $p_j \leftarrow -r_j^0$
    $m \leftarrow 0$
    while $r^m \neq 0$ and $m \leq M_{CG}$ do
        $\frac{dU_n}{db_j}p_j \leftarrow$ DD (Flow Equations) [1 EFS] ★
        $\frac{d\Psi_n}{db_j}p_j \leftarrow$ DD (Adjoint Equations) [1 EFS] ★
        $w_i = \frac{d^2F}{db_i db_j}p_j \leftarrow$ Hessian Expression
        $\eta \leftarrow \frac{r_i^m r_i^m}{p_j w_j}$
        $db_j^{m+1} \leftarrow db_j^m + \eta p_j$
        $r_j^{m+1} \leftarrow r_j^m + \eta w_j$
        $\beta \leftarrow \frac{r_i^{m+1} r_i^{m+1}}{r_j^m r_j^m}$
        $p_j \leftarrow -r_j^{m+1} + \beta p_j$
        $m \leftarrow m + 1$
    end while
    $b_j \leftarrow b_j + db_j$
    $k \leftarrow k + 1$
end while
```

$$b_i^{n+1} = b_i^n + db_i$$

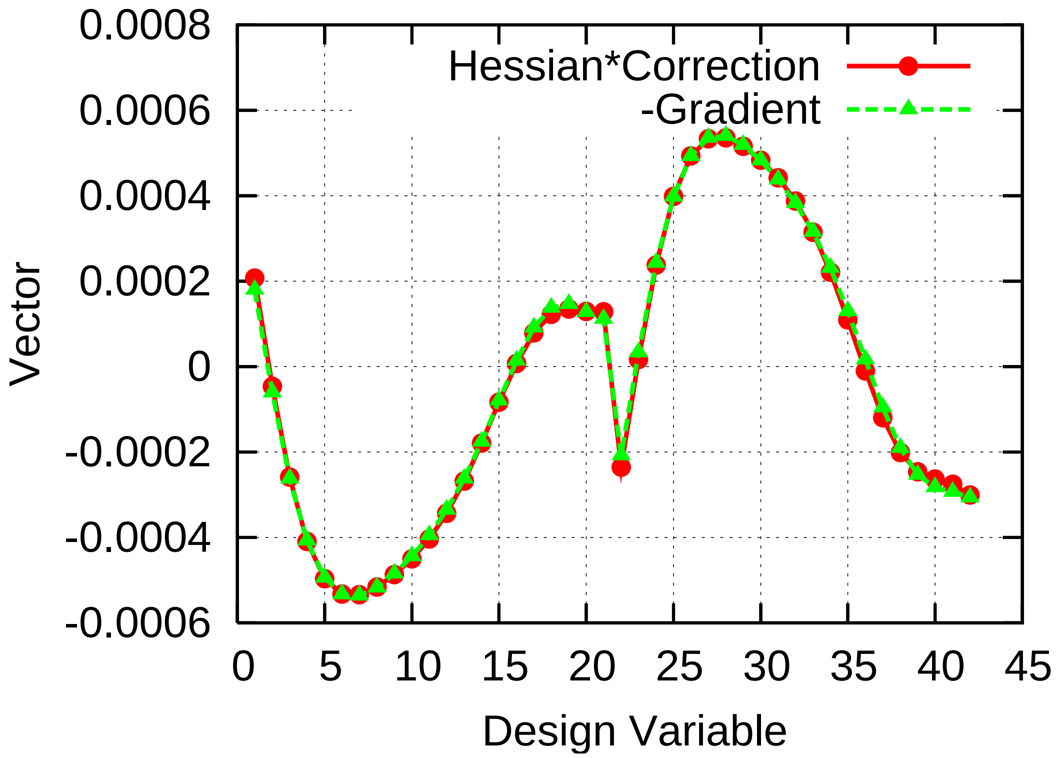
$$\frac{d^2F}{db_i db_j} db_j = -\frac{dF}{db_i}$$

Total Cost= 2+2M$_{CG}$ << N

★ ★
★ ★
★
★
⋮

Accuracy of the AV-DD Truncated Newton method

<u>**Problem:**</u> **Inverse Design of an Isolated Airfoil, 42 degrees of freedom**



$$\frac{d^2F}{db_i db_j} db_j = -\frac{dF}{db_i}$$

●—● $= \frac{d^2F}{db_i db_j} db_j$

db_j computed by the proposed truncated method, with M_{CG}=4, at the cost of 2+2x4=10 EFS

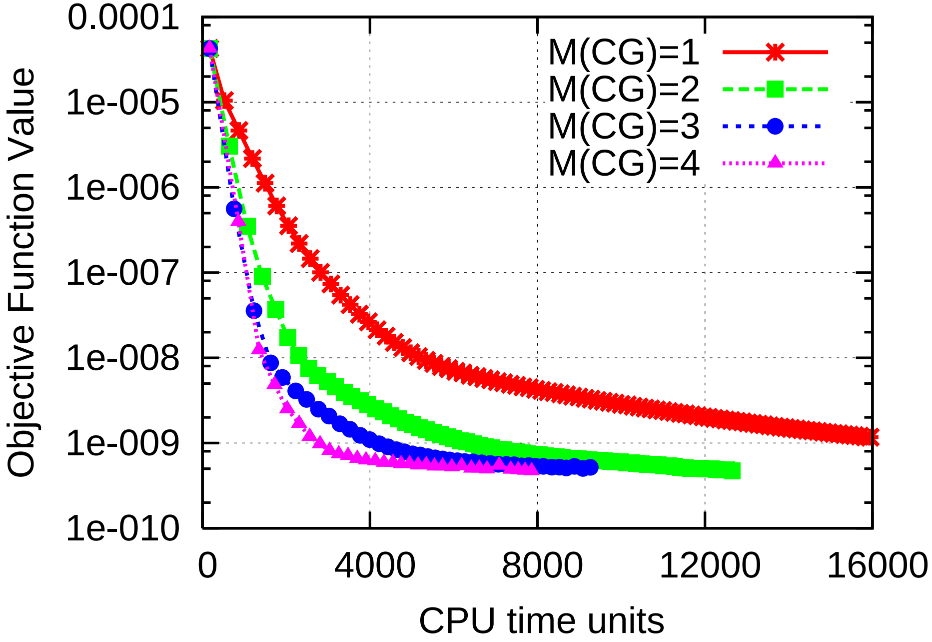
▲—▲ $= -\frac{dF}{db_i}$

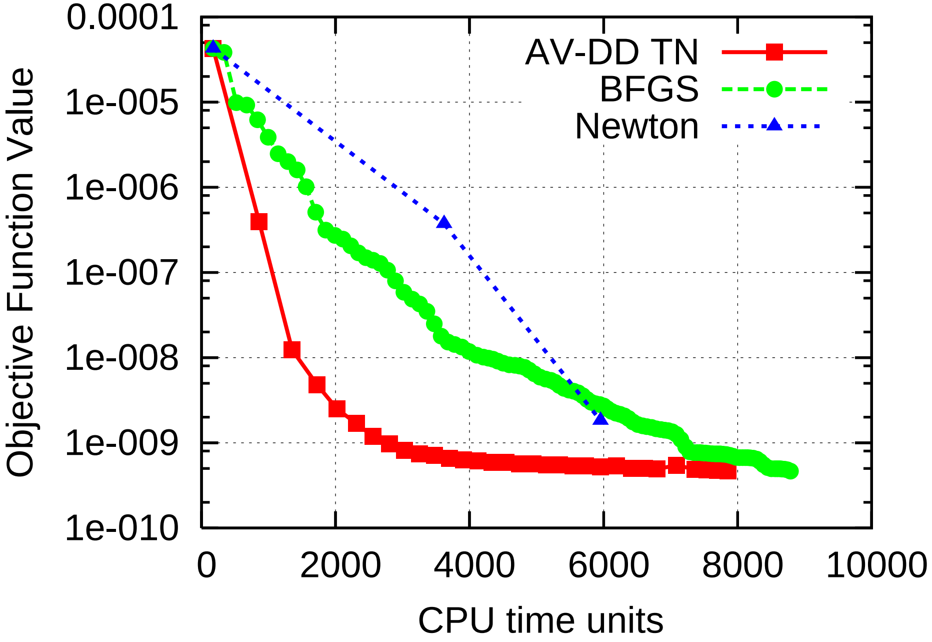
Computed by the “standard” adjoint method

AV-DD Truncated Newton method – Why & How?

Parametric Study of the recommended value of the number of CG steps (M_{CG})

Comparison of
AV-DD Truncated Newton method,
quasi-Newton BFGS &
(exact) Newton



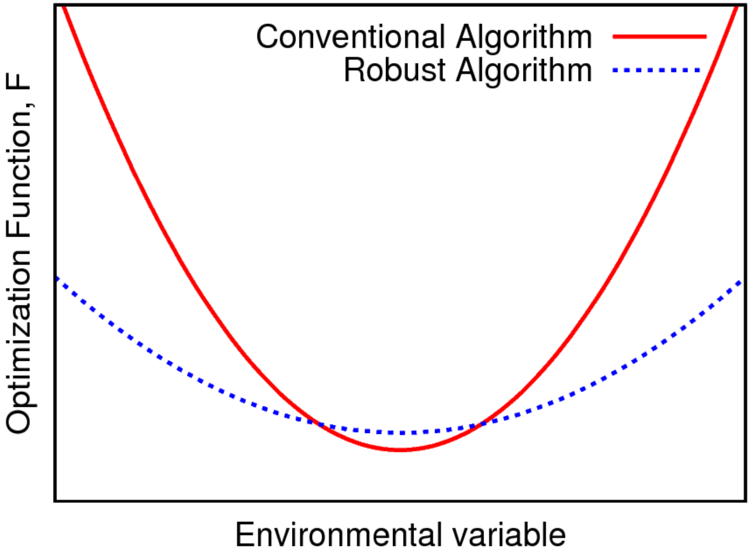


Inverse Design of an Isolated Airfoil, 42 degrees of freedom

Robust Optimization

(Or the need for higher-order derivatives…)

Robust Design – Problem Formulation



"Modified" Cost Function $\widehat{F}$:

$$\widehat{F} = \hat{\mu_F} + k\hat{\sigma_F}$$

- $\hat{\mu_F}$ estimated mean value of F (model ?).
- $\hat{\sigma_F}$ standar deviation of F around $\hat{\mu_F}$ (model ?)
- k constant (engineer's decision).

$$\hat{\mu_F} = F_{\mathrm{D}} + \frac{1}{2}\left[\frac{d^2F}{dc_i^2}\right]_{\mathrm{D}} \sigma_i^2$$

$$\hat{\sigma_F} = \sqrt{\left[\frac{dF}{dc_i}\right]_{\mathrm{D}}^2 \sigma_i^2 + \frac{1}{2}\left[\frac{d^2F}{dc_i dc_j}\right]_{\mathrm{D}}^2 \sigma_i^2 \sigma_j^2}$$

- $\overrightarrow{c}$ robust variables' vector with M elements ($M <<$). Assuming Gaussian distribution of c_i variables.
- σ_i deviation of i-th robust variable.

Robust Design

The steepest descent method is used.

$$b_l^{k+1} = b_l^k - \eta \frac{d\widehat{F^k}}{db_l}$$

- $\overrightarrow{b}$ design variables' vector with elements $(N >> M)$.
- η step.

Computation of the following derivative is required

$$\frac{d\widehat{F}}{db_l} = \underbrace{\frac{dF}{db_l}}_{term1} + \underbrace{\frac{1}{2}\frac{d^3F}{dc_i^2 db_l}\sigma_i^2}_{term2} + \underbrace{k\frac{2\frac{dF}{dc_i}\frac{d^2F}{dc_i db_l}\sigma_i^2 + \frac{d^2F}{dc_i dc_j}\frac{d^3F}{dc_i dc_j db_l}\sigma_i^2\sigma_j^2}{2\sqrt{\left[\frac{dF}{dc_i}\right]^2\sigma_i^2 + \frac{1}{2}\left[\frac{d^2F}{dc_i dc_j}\right]^2\sigma_i^2\sigma_j^2}}}_{term3}$$

So the derivatives of F, up to third order, with respect to $\overrightarrow{c}$ and $\overrightarrow{b}$, must be computed.

How to compute the required derivatives of F

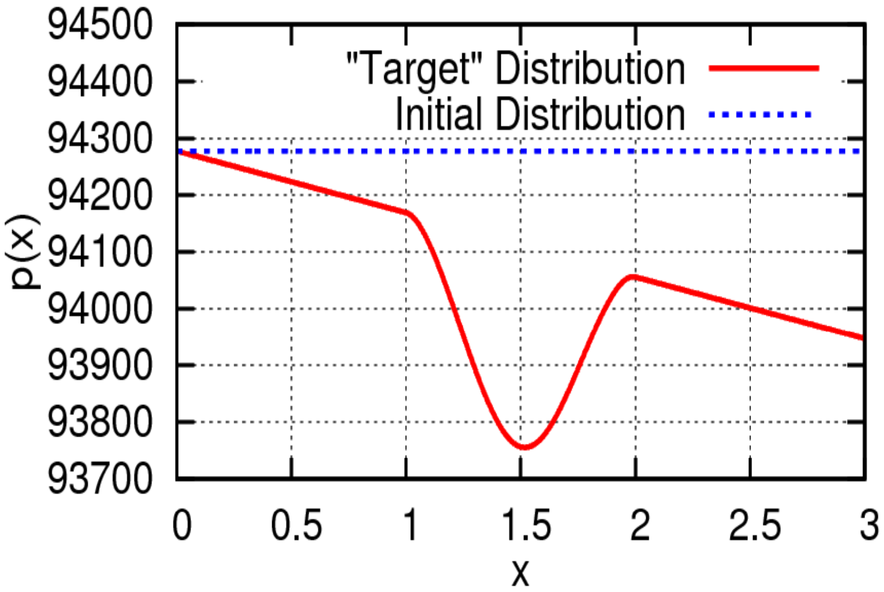
Derivative	**Method**	**Cost**	**Cost when $M = 1$**
$\frac{dF}{db_l}$	AV	1	1
$\frac{dF}{dc_i}$	DD	M	1
$\frac{d^2F}{dc_i dc_j}$	DD-DD	$\frac{M(M+1)}{2} + M$	2
$\frac{d^2F}{dc_i db_l}$	DD-AV	$1 + M$	2
$\frac{d^3F}{dc_i dc_j db_l}$	DD-DD-AV	$2 + 3M + M^2$	6

All costs are independent on the design variables' number.

Robust Design – A Pseudo 1D Example

$$\frac{\partial \vec{f}}{\partial x} = \vec{q_s} + \vec{q_\nu}$$

$$\vec{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix}, \vec{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(\rho E + p) \end{bmatrix}, \vec{q_s} = -\frac{1}{S}\frac{dS}{dx}\begin{bmatrix} \rho u \\ \rho u^2 \\ u(\rho E + p) \end{bmatrix}, \quad \vec{q_\nu} = -\lambda \frac{dx_i}{2D_i}\begin{bmatrix} 0 \\ \rho u^2 \\ \rho u^3 \end{bmatrix}$$

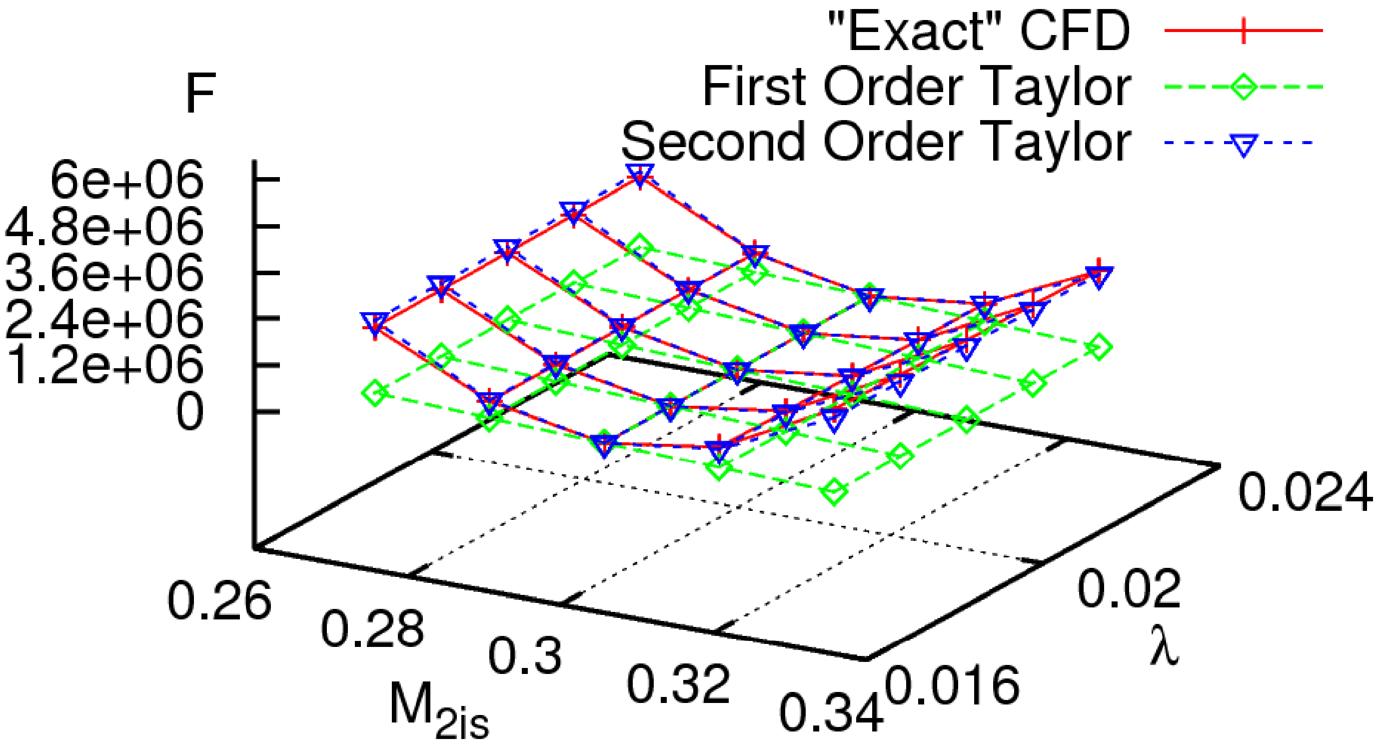


Two environmental variables:

- **Outlet Mach number M_2**
- **Darcy friction loss coefficient λ**

(Solution to be obtained depends on *k*)

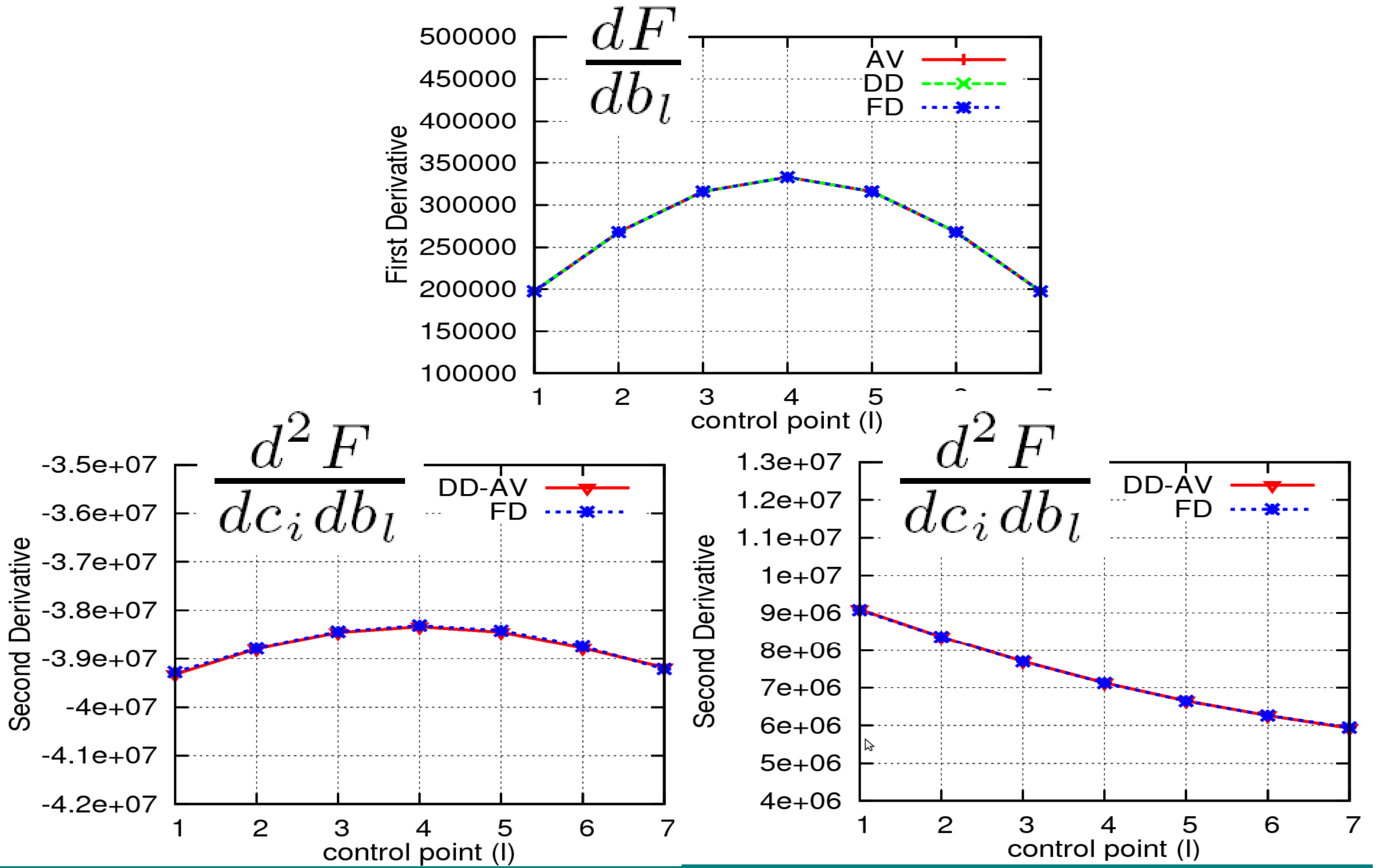
Robust Design – A Pseudo 1D Example



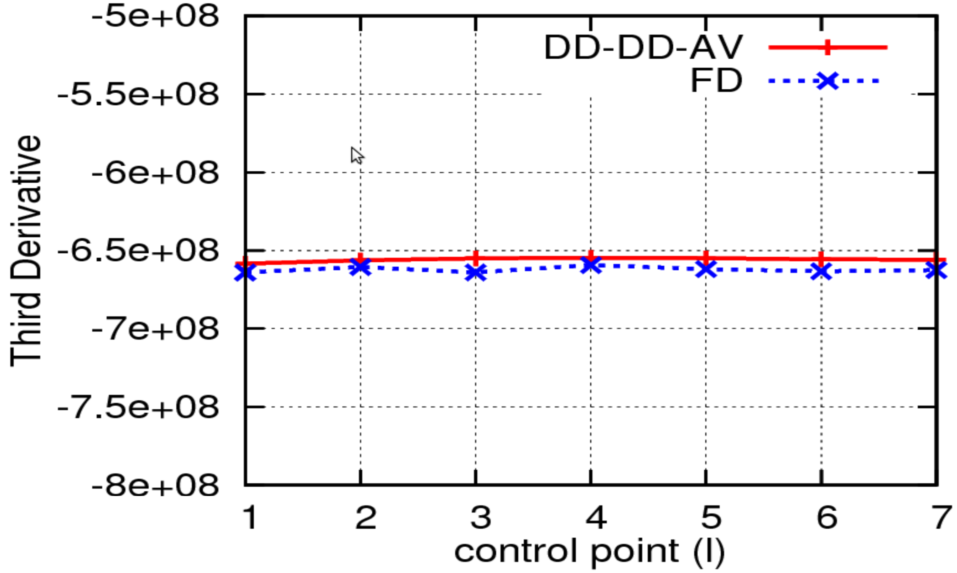
Why second-order Taylor expansion?

First Order Taylor expansion fails to capture the non-linearity of the problem

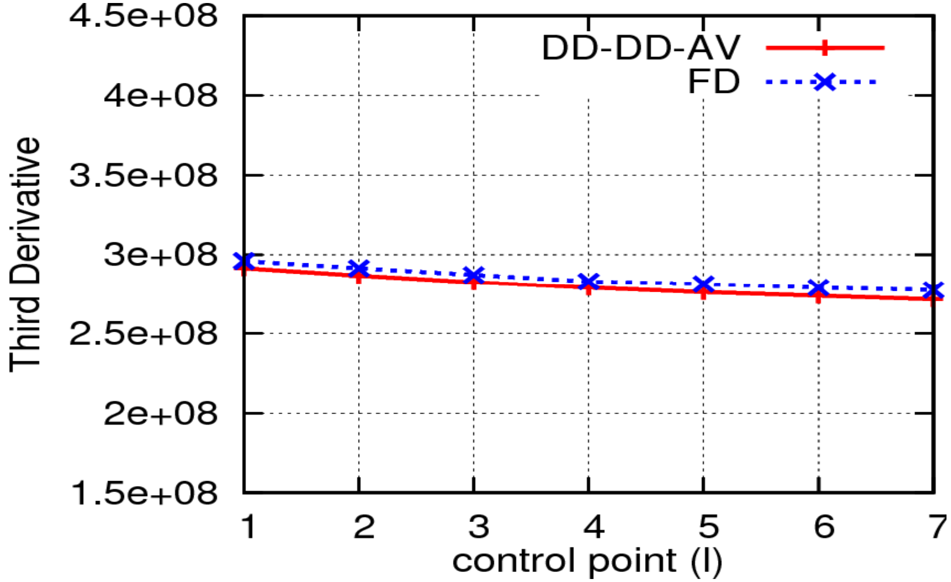
Robust Design – A Pseudo 1D Example

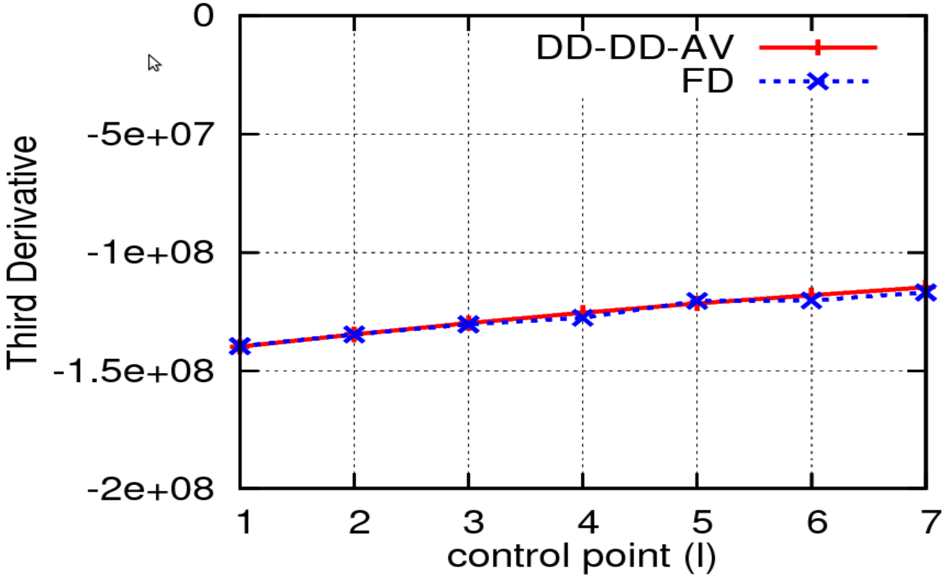


Robust Design – A Pseudo 1D Example

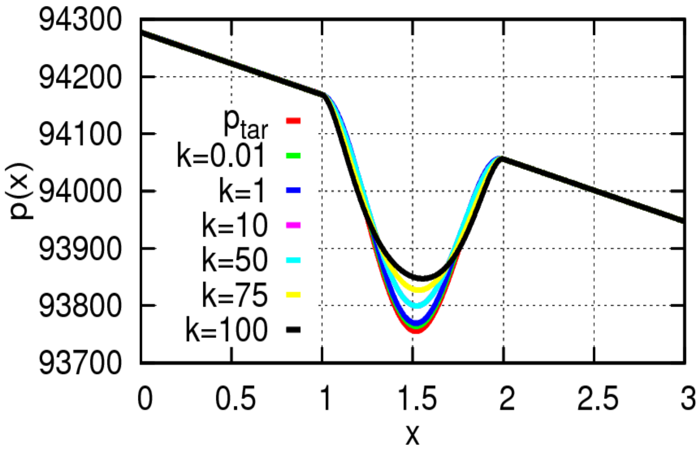


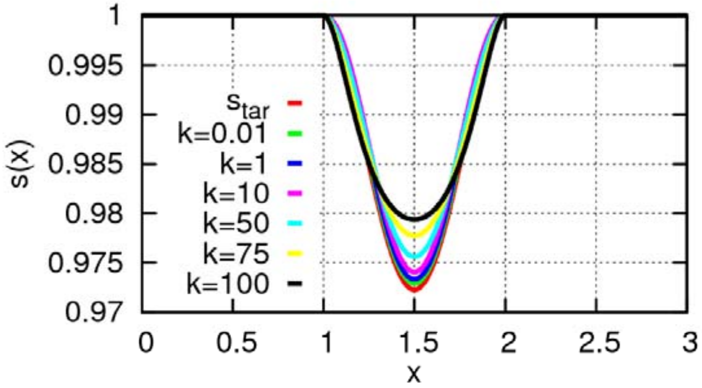
$$\frac{d^3 F}{dc_i dc_j db_l}$$

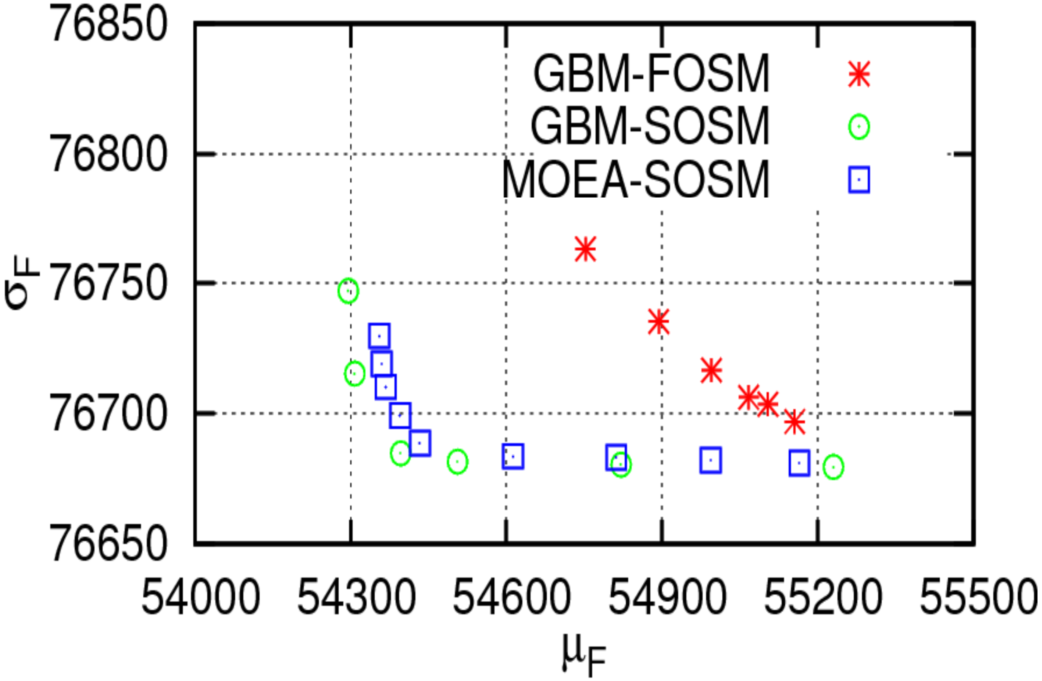




Robust Design – A Pseudo 1D Example

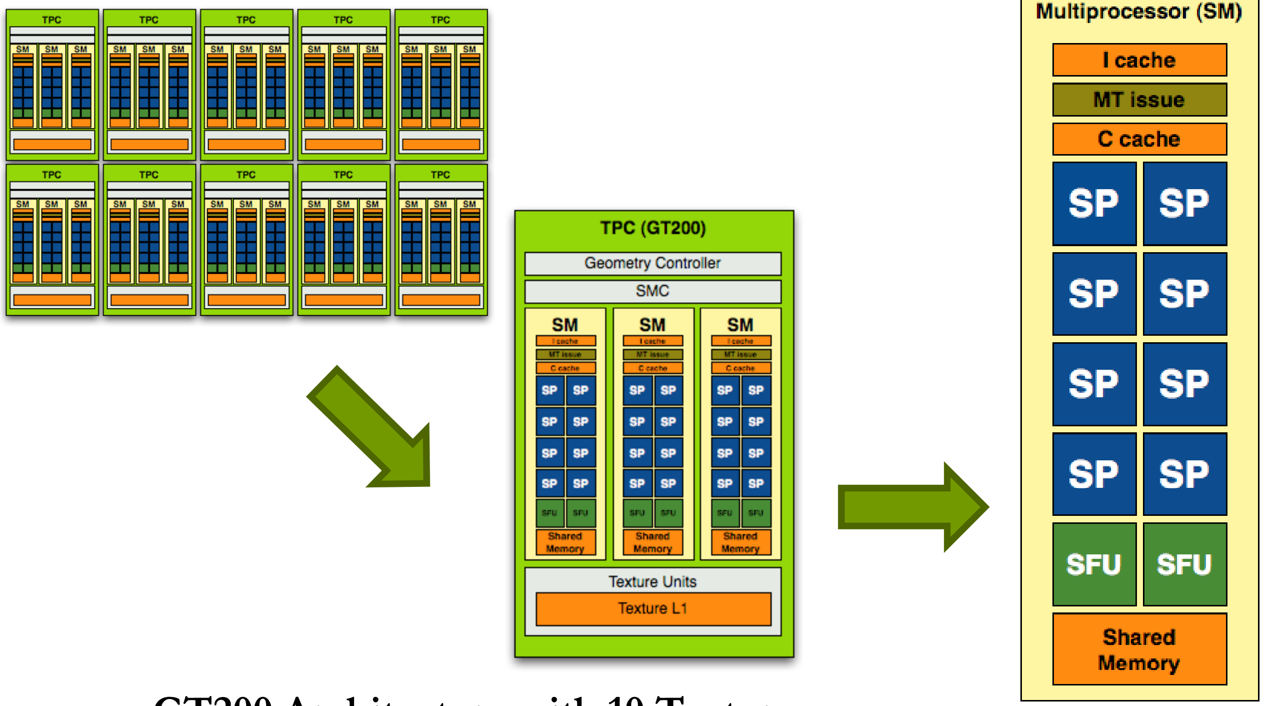




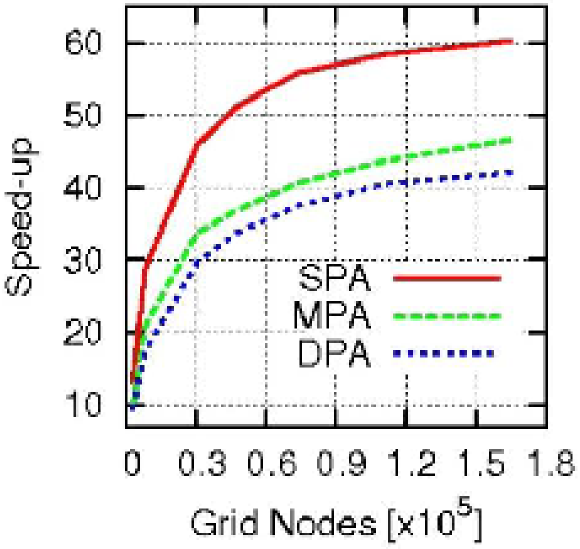


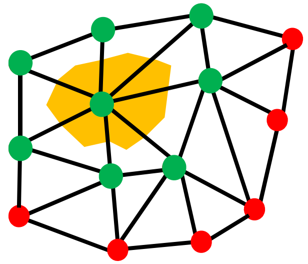
Hessian Computation & Parallelization on GPUs

NVIDIA GTX 285



GT200 Architecture with 10 Texture Processor Clusters (TPCs)





I.C. KAMPOLIS, X.S. TROMPOUKIS, V.G. ASOUTI, K.C. GIANNAKOGLOU: 'CFD-based Analysis and Two-level Aerodynamic Optimization on Graphics Processing Units', Computer Methods in Applied Mechanics and Engineering, Vol. 199, No. 9-12, pp. 712-722, 2010.

V.G. ASOUTI, X.S. TROMPOUKIS, I.C. KAMPOLIS, K.C. GIANNAKOGLOU: 'Unsteady CFD Computations Using Vertex-Centered Finite Volumes for Unstructured Grids on Graphics Processing Units', International Journal for Numerical Methods in Fluids, to appear 2010.

Summary/Conclusions

► In aerodynamic optimization, the use of second-order sensitivities dramatically reduces the number of required optimization cycles. However, with many design variables, the CPU cost per cycle of exact Newton methods becomes “prohibitive”.

► This problem can be overcome by means of the exactly-initialized quasi-Newton algorithm. This approach outperforms both exact and quasi-Newton methods, irrespective of the number of design variables.

► A further noticeable improvement is achieved by applying a one-shot algorithm in which the state and optimization equations are solved simultaneously. In all cases, the one-shot method outperforms its conventional counterpart. The use of the one-shot, exactly initialized, quasi-Newton algorithm is highly recommended.

► With problems with many design variables, the use of Truncated Newton Methods (based on AV-DD) are highly recommended.

► Robust design methods (SOSM approaches) may also benefit a lot from the availability of efficient methods to compute high-order derivatives of F.

Acknowledgements

Dr. Dimitris PAPADIMITRIOU

Dr. Alexander ZYMARIS

Vaggelis PAPOUTSIS

Thomas ZERVOGIANNIS

Forthcoming Event

ECCOMAS
Thematic Conference

CFD & OPTIMIZATION

16-18 May 2011, Antalya-Turkey

http://eccomas.ae.metu.edu.tr

CALL FOR PAPERS

The thematic conference on CFD & OPTIMIZATION is devoted to all aspects of optimization studies using CFD tools. The conference topics include but are not limited to :

- Gradient based methods
- Adjoint methods
- Evolutionary algorithms
- Genetic algorithms
- Response Surface methods
- ANN based methods
- POD based methods
- Hybrid methods
- Multi-objective optimization
- Parallel algorithms

- Aerodynamic shape optimization
- Aeroelastic optimization
- Aeroacoustic optimization
- Drag minimization
- Robust control
- Turbomachinery applications
- Wind energy applications
- Multi-disciplinary Design Optimization

Abstract submission deadline : January 10, 2011

http://eccomas.ae.metu.edu.tr