NATIONAL TECHNICAL UNIVERSITY OF ATHENS (NTUA)
SCHOOL OF MECHANICAL ENGINEERING
PARALLEL CFD & OPTIMIZATION UNIT (PCOpt/NTUA)

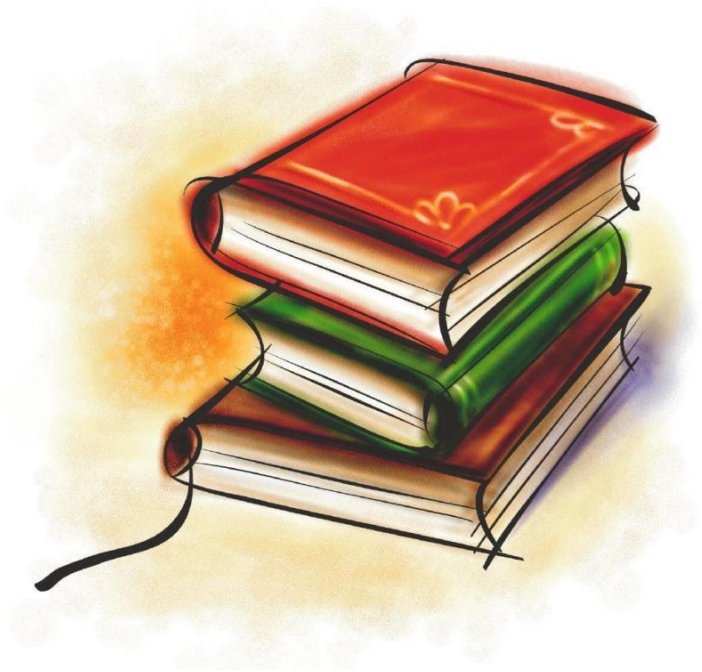
The Adjoint Method in PDE-based Shape Optimisation (ShpO)

Dr. Kyriakos C. Giannakoglou, Professor NTUA

Dr. Varvara Asouti, Adjunct Lecturer NTUA

Contributors:

Dr. E. Papoutsis-Kiachagias
Dr. X. Trompoukis
Dr. T. Tsiakas
Dr. A. Margetis
Dr. K. Gkaragkounis
Dr. M. Kontou
Dr. T. Skamagkis
Dr. I. Kavvadias
Dr. A. Zymaris
F. Libretti
S. Stalikas
D. Katmeridis



Introduction

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

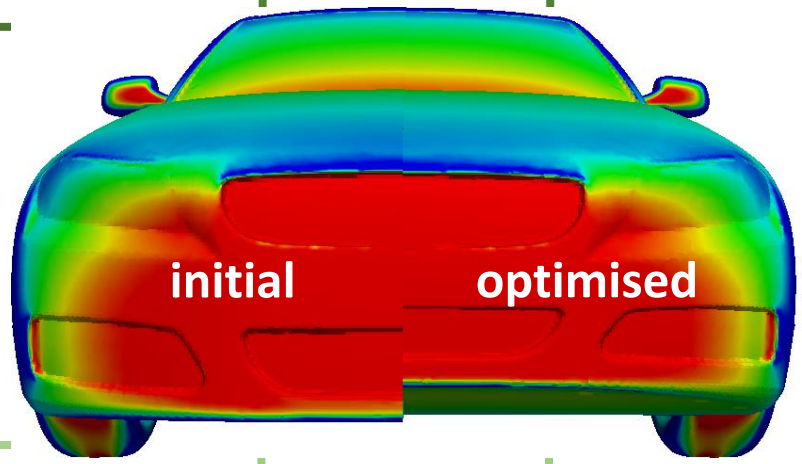
Prerequisites for running ShpO in CFD

PR1

Flow (CFD) Model (& s/w) incl. **grid generation & grid displacement**.

PR3

Objective & Constraint Function(s); gradient-based optimisation requires their derivatives w.r.t. b_n, n=1,…,**N**.



PR2

Case Parametrisation, defining the design variables (Degrees of Freedom, **DoFs**) b_n, n=1,…,**N** of the case.

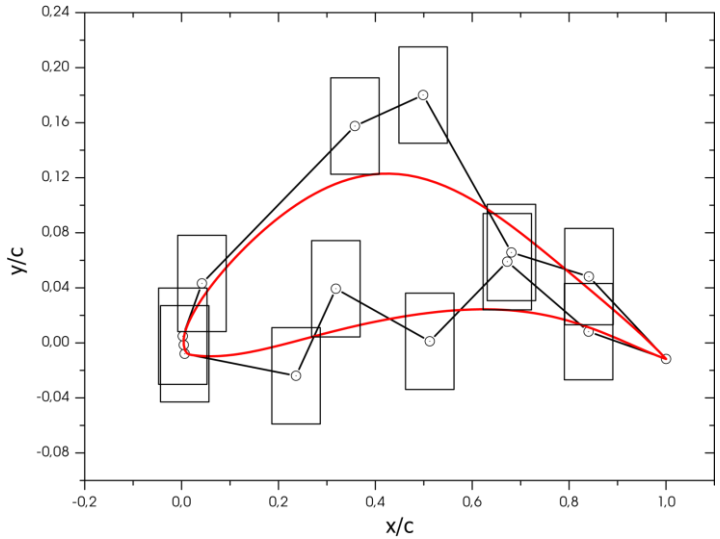
Optimisation Method, incl. the computation of **grad(…)** if needed.

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

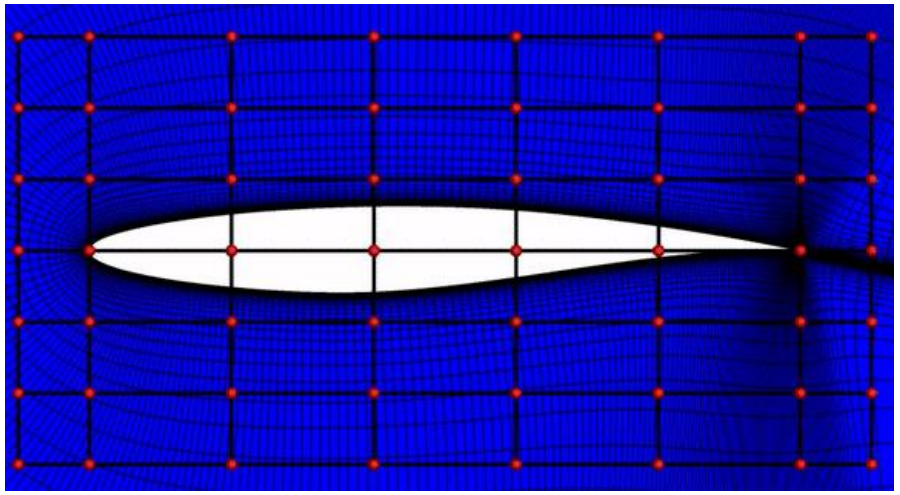
Shape Parametrisation

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

The selection of a CAD-based or CAD-free parametrisation affects the quality of shapes generated in the course of the optimisation & its convergence rate. Different parametrisations, with different sets of design variables (Degrees of Freedom, DoFs) b_n, n=1,…,N, may lead to different optimal solutions.

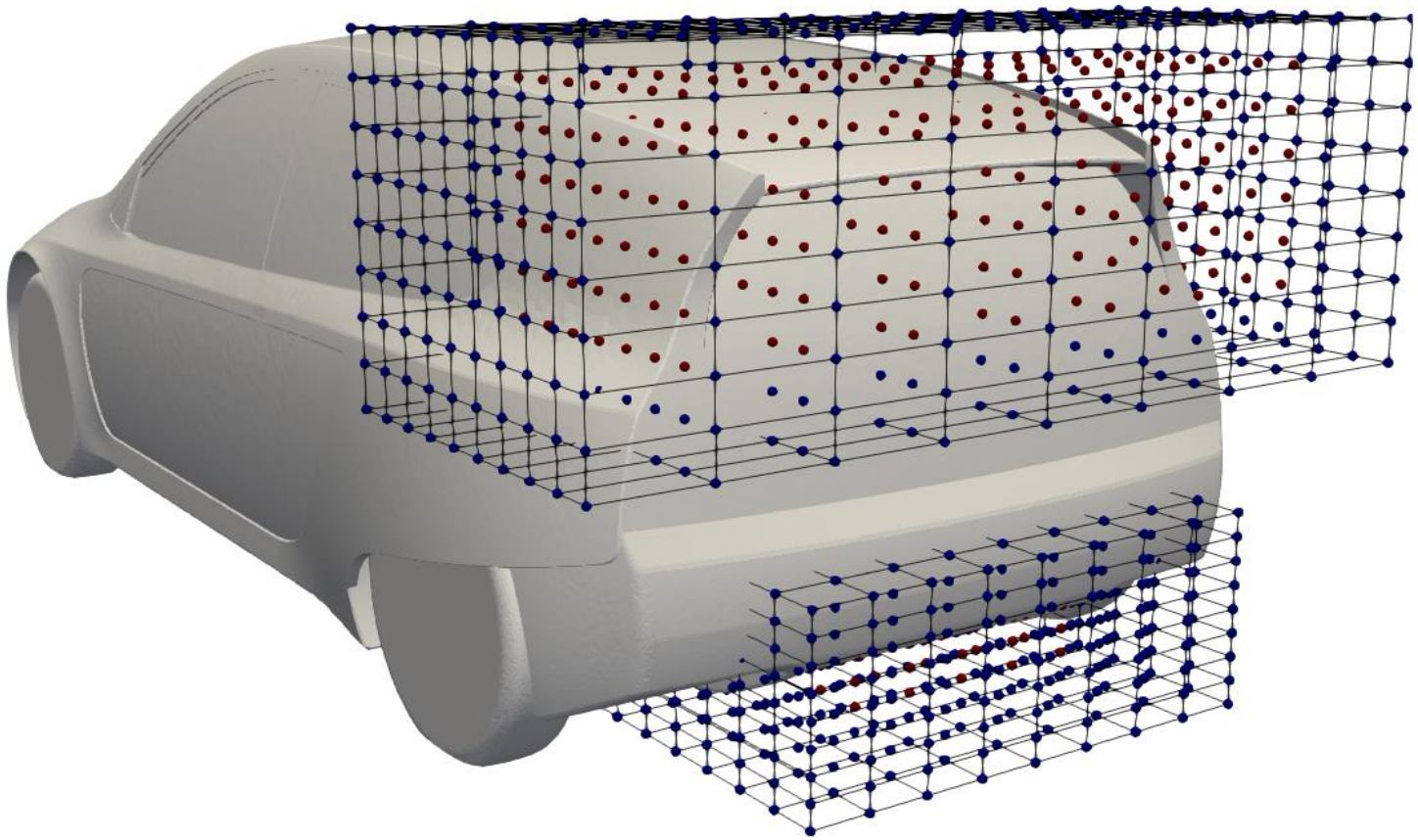


CAD-based Parametrisation: Using Bezier, NURBS, Splines etc patches and native CAD parameters. The coordinates of displaceable control points stand for the N design variables.



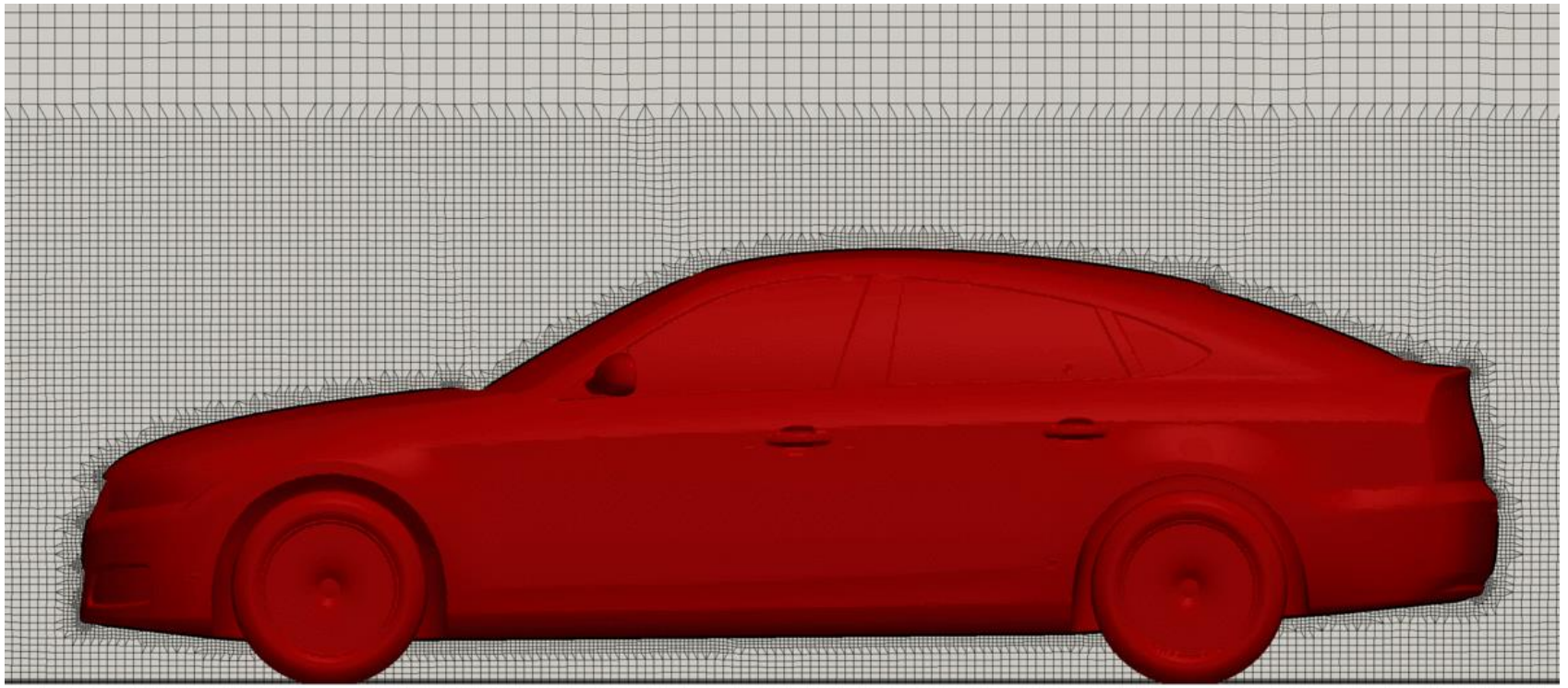
CAD-free Parametrisation: Using morphing techniques based, for instance, on volumetric B-Splines. Shape & CFD grid are simultaneously adapted. An alternative way is node-based parametrisation. Returning to CAD is difficult, possibly impairing the quality of the designed shapes.

Shape Parameterisation (CAD-Free, for a Car)



Symmetry?
Other constraints?

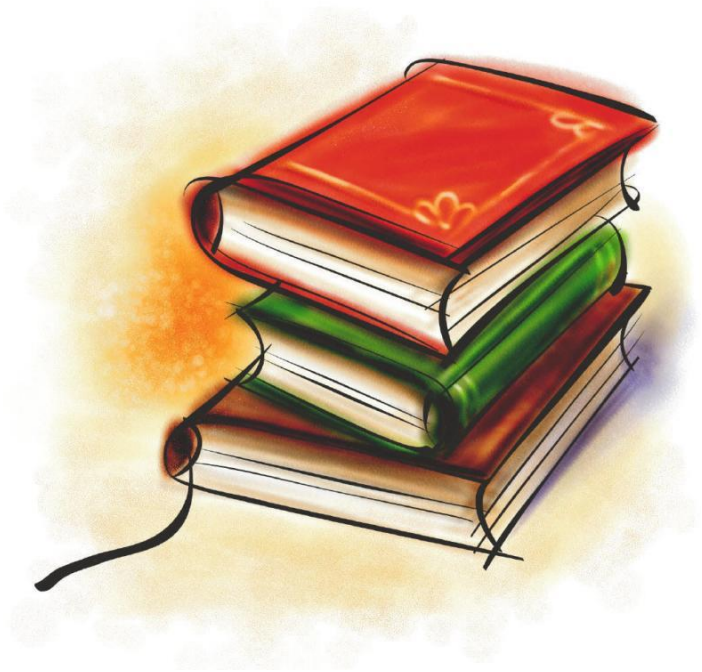
Grid Displacement Tools to Support a Shape Optimisation (ShpO) Loop



Fast grid displacement tools to support ShpO. **Internal grid is adapted to the displaced boundaries, in each optimisation cycle. Very efficient. This is a good reason to avoid remeshing; other reasons?**

Also used in: Flows in domains with moving boundaries, aeroelasticity, etc

Some relevant Grid Displacement methods: Algebraic models (Inverse Distance Weighted, IDW), Laplacian Models, Volumetric B-Splines, Linear/Torsional Spring Analogies, Harmonic Coordinates, Delaunay Graph, etc



Continuous Adjoint Extended to Multi-Dimensional / Turbulent Flows, for ShpO

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

Continuous Adjoint in 2D/3D Laminar Flows- ShpO – Primal Eqs.

Laminar flow of an incompressible fluid :

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

$$R_i^v = \frac{\partial (v_j v_i)}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0 \ , \qquad i = 1, 2(, 3)$$

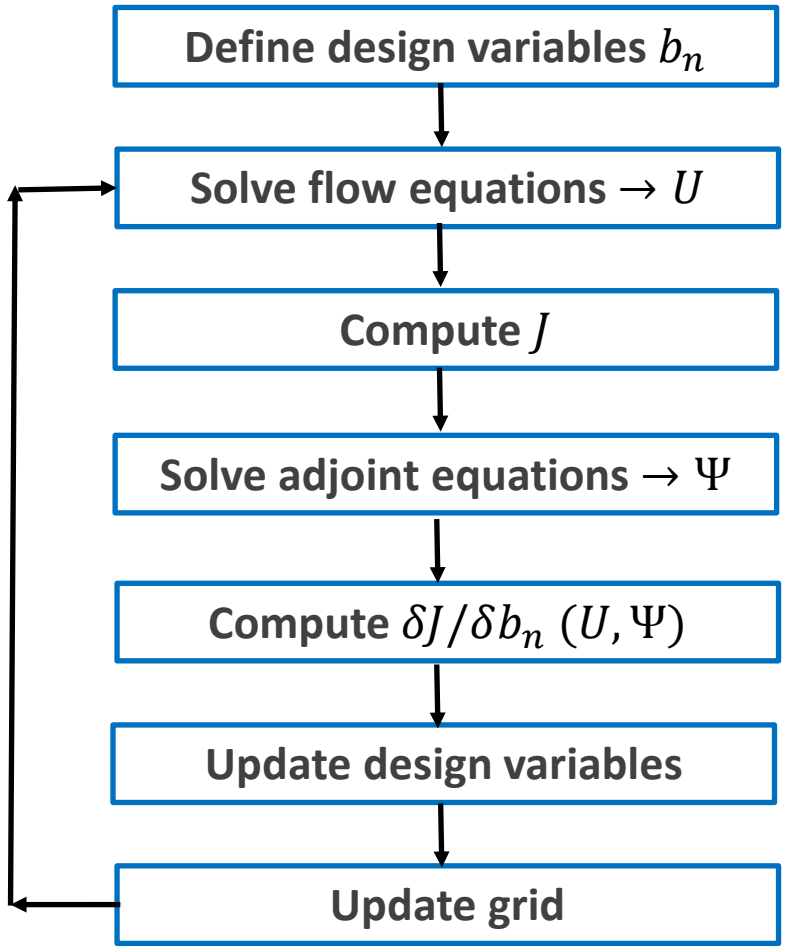
where $\tau_{ij} = \nu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$.

For turbulent flows, see next slides.

► **Steps in the development of the Continuous Adjoint Method to this flow model:**

1. **Differentiate the primal equations (PDEs) and the objective function J w.r.t. b_n.**
2. **Derive adjoint equations (field eqs. in the form of PDEs) as well as the adjoint BCs.**
3. **Discretize & solve the (primal &) adjoint PDEs.**
4. **Compute grad(J), a.k.a. Sensitivity Derivatives (SDs).**
5. **Update b_n!**

Adjoint-based Optimisation (adjointOptimisationFoam)



⇨ Depends on the shape parametrisation (in ShpO)

⇨ Solves the primal (flow) equations & computes the primal solution U

⇨ J may stand for forces, moments, losses, etc

⇨ Solves the adjoint equations & computes the adjoint field Ψ

⇨ Computes Grad(J) w.r.t. the design variables

⇨ Using steepest descent, CG, Quasi-Newton methods (BFGS), etc.

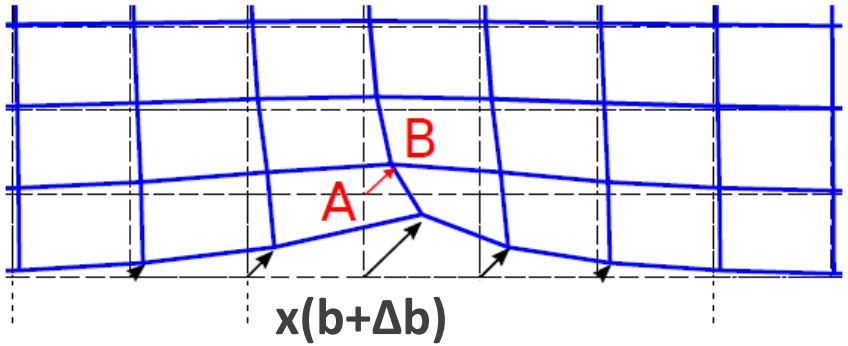
⇨ Using PDE-based, Algebraic models, etc.

Everything is implemented within OpenFOAM v2206: No need for third-party software, external scripts, etc

Continuous Adjoint in 2D/3D Flows- ShpO - Total/Partial Derivatives

$$\frac{\delta \Phi}{\delta b_n} = \frac{\partial \Phi}{\partial b_n} + \frac{\partial \Phi}{\partial x_k}\frac{\delta x_k}{\delta b_n}$$

Grid Sensitivities



Partial (b_n) & spatial (x_k) derivatives permute!

$$\frac{\partial}{\partial b_n}\left(\frac{\partial \Phi}{\partial x_j}\right) = \frac{\partial}{\partial x_j}\left(\frac{\partial \Phi}{\partial b_n}\right)$$

Total (b_n) & spatial (x_k) derivatives don't!

$$\frac{\delta}{\delta b_n}\left(\frac{\partial \Phi}{\partial x_j}\right) = \frac{\partial}{\partial x_j}\left(\frac{\delta \Phi}{\delta b_n}\right) - \frac{\partial \Phi}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)$$

The math. development of the adjoint method & the handling of Grid Sensitivities gives rise to 3 continuous adjoint formulations:

- **Field Integral (FI) Adjoint**
- **Enhanced-Surface Integral (E-SI) Adjoint**
- ~~**Severed SI Adjoint**~~

Just: Surface Integral (SI) Adjoint

The FI Adjoint Formulation (FI: Field Integral)

- **The resulting SDs include field integrals of Grid Sensitivities $\delta x_i/\delta b_n$.**

An example of such a term in the SD expression is:

$$\frac{\delta J}{\delta b_n} = \ldots + \int_\Omega \left\{ -u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} \right\} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega$$

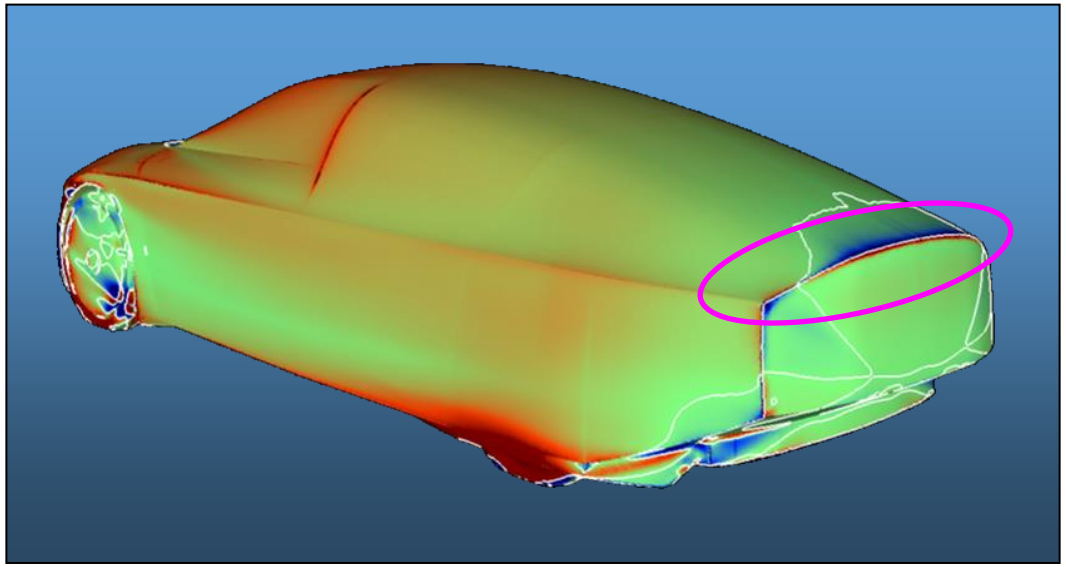
Grid Sensitivities

- **Computing Grid Sensitivities requires a Grid Displacement Model (GDM) & its differentiation (finite differences, analytical differentiation, etc).**

- **SDs computed by the FI Adjoint are accurate; however, the cost of the part of the code dealing with grid sensitivities scales with N.**

- **Computing Sensitivity Maps (see next slide) using the FI Adjoint becomes prohibitively expensive, as the number of surface nodes increases.**

Sensitivity Map



Plotted on the initial shape, the sensitivity map represents iso-areas of δF/δn, where n is the unit vector normal to the surface and indicates how much each part of the body surface, if deformed (inwards or outwards), contributes to the improvement of the objective function F value. The sensitivity map offers guidance, even to an experienced designer, regarding the direction of possible surface displacements, prior to running an optimization loop.

This case will be presented later!

The E-SI Adjoint Formulation (E-SI: Enhanced Surface Integral)

- To eliminate grid sensitivities over the domain, the **Enhanced-SI (E-SI) adjoint** assumes a **Laplacian GDM)** to be introduced into the primal equations:

$$R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0$$

- New Lagrangian & new adjoint fields (m_i^α):

$$L = J + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega m_i^a R_i^m d\Omega$$

- **Grid Sensitivities** are eliminated by satisfying the **adjoint GDM equations**:

$$R_k^{m^a} = \frac{\partial^2 m_k^a}{\partial x_j^2} + \frac{\partial}{\partial x_j}\left\{ u_i v_j \frac{\partial v_i}{\partial x_k} + u_j \frac{\partial p}{\partial x_k} + \tau_{ij}^a \frac{\partial v_i}{\partial x_k} - u_i \frac{\partial \tau_{ij}}{\partial x_k} - q \frac{\partial v_j}{\partial x_k} - \frac{\partial J_{\Omega'}}{\partial x_k} \right\} = 0$$

Journal of Computational Physics, 301:1-18, 2015.
Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.

The E-SI Adjoint Formulation (E-SI: Enhanced Surface Integral)

- **The Enhanced-Surface Integral (E-SI) continuous adjoint method computes accurate sensitivity derivatives, with the additional advantage that the gradient of J is expressed in terms of surface integrals only ("reduced adjoint").**

- **Practically, the E-SI adjoint is as accurate as the FI adjoint, though much cheaper!**

- **What if a different (than the Laplacian) GDM is used to adapt the CFD grid to the updated geometries, during the optimisation? The adjoint Laplace GDM can safely be used even in this case.**

- **The Severed-SI adjoint (by arbitrarily eliminating the effect of Grid Sensitivities) is NOT recommended. Examples follow!**

Journal of Computational Physics, 301:1-18, 2015.
Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.

Continuous Adjoint Formulations – At a Glance

FI Adjoint

$$\frac{\delta J}{\delta b_n} = \int_\Omega A_{jk}\left(U,\Psi\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega$$

$$= \int_S A_{jk}n_j\frac{\delta x_k}{\delta b_n}dS - \int_\Omega \frac{\partial A_{jk}}{\partial x_j}\frac{\delta x_k}{\delta b_n}d\Omega$$

Includes Field Integrals of Grid Sensitivities →
SDs: accurate but can be expensive to compute (cost ∝ N)

Enhanced-SI (E-SI) Adjoint

$$\frac{\partial^2 m_k^a}{\partial x_j^2} - \frac{\partial A_{jk}}{\partial x_j} = 0$$

$$\frac{\delta J}{\delta b_n} = \int_S A_{jk}n_j\frac{\delta x_k}{\delta b_n}dS - \int_S \frac{\partial m_k^a}{\partial x_j}n_j\frac{\delta x_k}{\delta b_n}dS$$

Enhanced Surface Integrals
To eliminate field integrals of Grid Sensitivities, the Adjoint to a Laplace Grid Displacement model must be included (and solved).
SDs: cheap and accurate to evaluate!

(Severed) SI Adjoint

$$\frac{\delta J}{\delta b_n} = \int_S A_{jk}n_j\frac{\delta x_k}{\delta b_n}dS$$

SDs: cheap to evaluate but inaccurate!

Continuous Adjoint Formulations – Recommended Treatments:

FI Adjoint

$$\frac{\delta J}{\delta b_n} = \int_\Omega A_{jk}\left(U,\Psi\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega$$

$$= \int_S A_{jk}n_j\frac{\delta x_k}{\delta b_n}dS - \int_\Omega \frac{\partial A_{jk}}{\partial x_j}\frac{\delta x_k}{\delta b_n}d\Omega$$

SI Adjoint

$$\frac{\partial^2 m_k^a}{\partial x_j^2} - \frac{\partial A_{jk}}{\partial x_j} = 0$$

$$\frac{\delta J}{\delta b_n} = \int_S A_{jk}n_j\frac{\delta x_k}{\delta b_n}dS - \int_S \frac{\partial m_k^a}{\partial x_j}n_j\frac{\delta x_k}{\delta b_n}dS$$

Includes Field Integrals of Grid Sensitivities →
SDs: accurate but can be expensive to compute (cost ∝ N)

Includes only Surface Integrals
To eliminate field integrals of Grid Sensitivities, the Adjoint to a Laplace Grid Displacement model must also be included (and solved).
SDs: cheap and accurate to evaluate!
Appropriate for sensitivity maps.

Turbulent Flows: Continuous Adjoint & Turbulence Models

- Enrich the Lagrangian by including the turbulence model eq(s). and the Hamilton-Jacobi eq. for computing distances from solid walls (if needed). Example: **Spalart-Allmaras** model:

$$L = J + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu_a} R^{\tilde{\nu}} d\Omega + \int_\Omega \Delta_a R^\Delta d\Omega$$

- Extra adjoint equations (adjoint to the turbulence model PDEs), new terms in the adjoint mean flow eqs. and BCs.

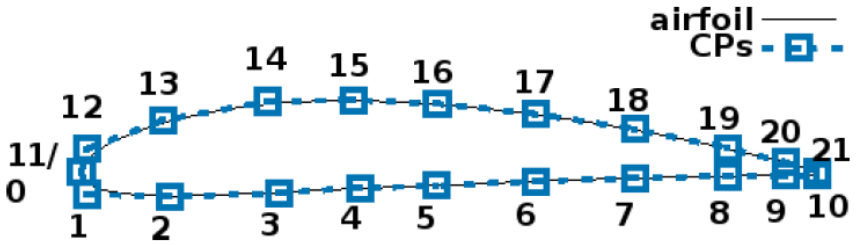
- Extended to models using **Wall Functions** by introducing the **Adjoint Wall Function**.

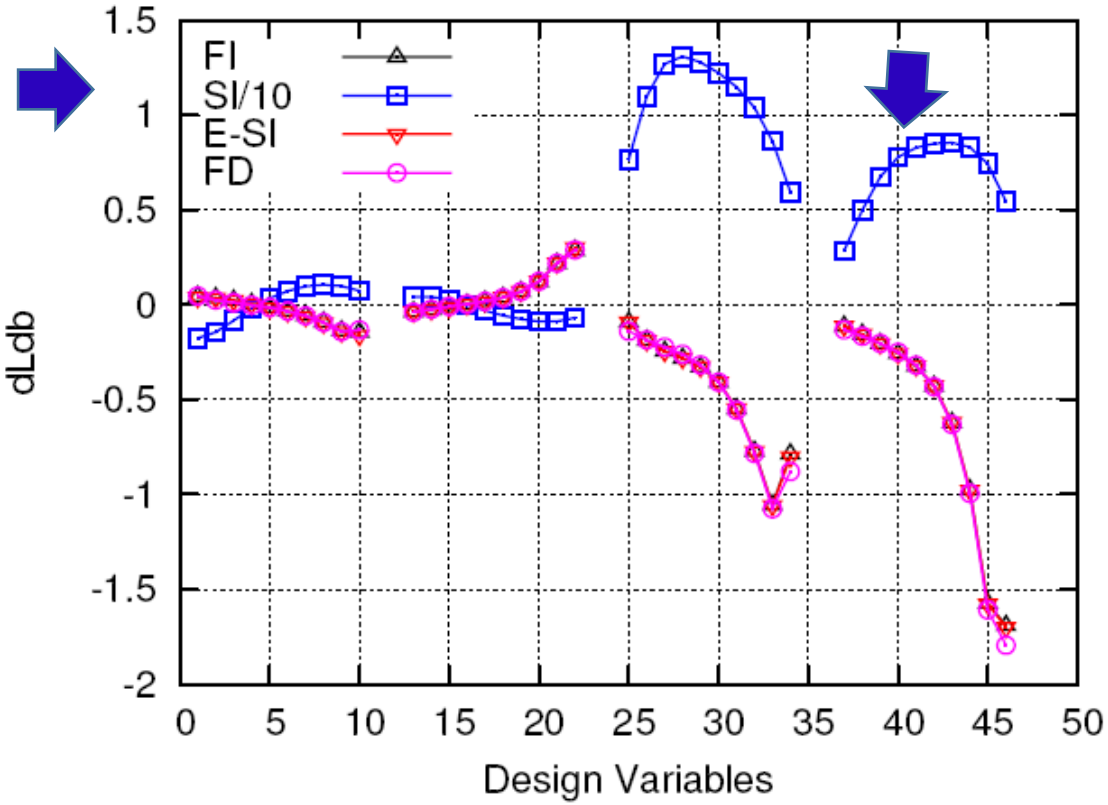
- Similar developments for the adjoint to the **k-ε** and **k-ω SST** turbulence models.

Computers & Fluids, 38:528-1538, 2009.
Journal of Computational Physics, 229(13): 5228-5245, 2010.
Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.
Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.

Comparison of the FI, E-SI, Severed-SI Continuous Adjoint & FD

Severed-SI Adjoint





Turbulent flow around the NACA0012 airfoil ($Re=10^6$, $a_{inf}=3^o$) : Lift SDs computed by the FI, Severed-SI (scaled by 10 to fit into the same diagram), E-SI adjoint and FD. SDs computed w.r.t. the x (ID<23) and y (ID>23) coordinates of 24 NURBS CPs parametrising the two airfoil sides. The adjoint to the Spalart-Allmaras turbulence model is used in all cases.

Continuous Adjoint to the Spalart-Allmaras Turbulence Model (1/2)

Example: Low-Re Spalart-Allmaras Turbulence Model:

An additional primal equation:

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

where $\nu_t = \tilde{\nu} f_{v_1}$.

Models depending on distances from the walls – The Eikonal equation:

$$R^{\Delta} = \frac{\partial (c_j \Delta)}{\partial x_j} - \Delta \frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0 \quad , \quad c_j = \frac{\partial \Delta}{\partial x_j}$$

should be included in the augmented objective function:

$$L = J + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu_a} R^{\tilde{\nu}} d\Omega + \int_\Omega \Delta_a R^{\Delta} d\Omega$$

Continuous Adjoint to the Spalart-Allmaras Turbulence Model (2/2)

Example: Low-Re Spalart-Allmaras Turbulence Model:

New field adjoint equations:

$$R^{\widetilde{\nu_a}} = -\frac{\partial(v_j\widetilde{\nu_a})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu_a}}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial\widetilde{\nu_a}}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_j} + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu_a}\frac{\partial\widetilde{\nu}}{\partial x_j}\right) + \widetilde{\nu_a}\widetilde{\nu}\mathcal{C}_{\widetilde{\nu}}$$
$$+\frac{\partial\nu_t}{\partial\widetilde{\nu}}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right) + (-P+D)\,\widetilde{\nu_a} + \frac{\partial J_{\Omega'}}{\partial\widetilde{\nu}} = 0$$

$$R^{\Delta_a} = -2\frac{\partial}{\partial x_j}\left(\Delta_a\frac{\partial\Delta}{\partial x_j}\right) + \widetilde{\nu}\widetilde{\nu_a}\mathcal{C}_{\Delta} = 0$$

Computers & Fluids, 38:528-1538, 2009.
Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.

(Continuous) Adjoint to other Turbulence Models

For Other Low-Re Turbulence Models:

k-ε Model:

$$L = J + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega k^a R^k d\Omega + \int_\Omega \epsilon^a R^\epsilon d\Omega$$

Engineering Optimization, 47(3):370-389, 2015.

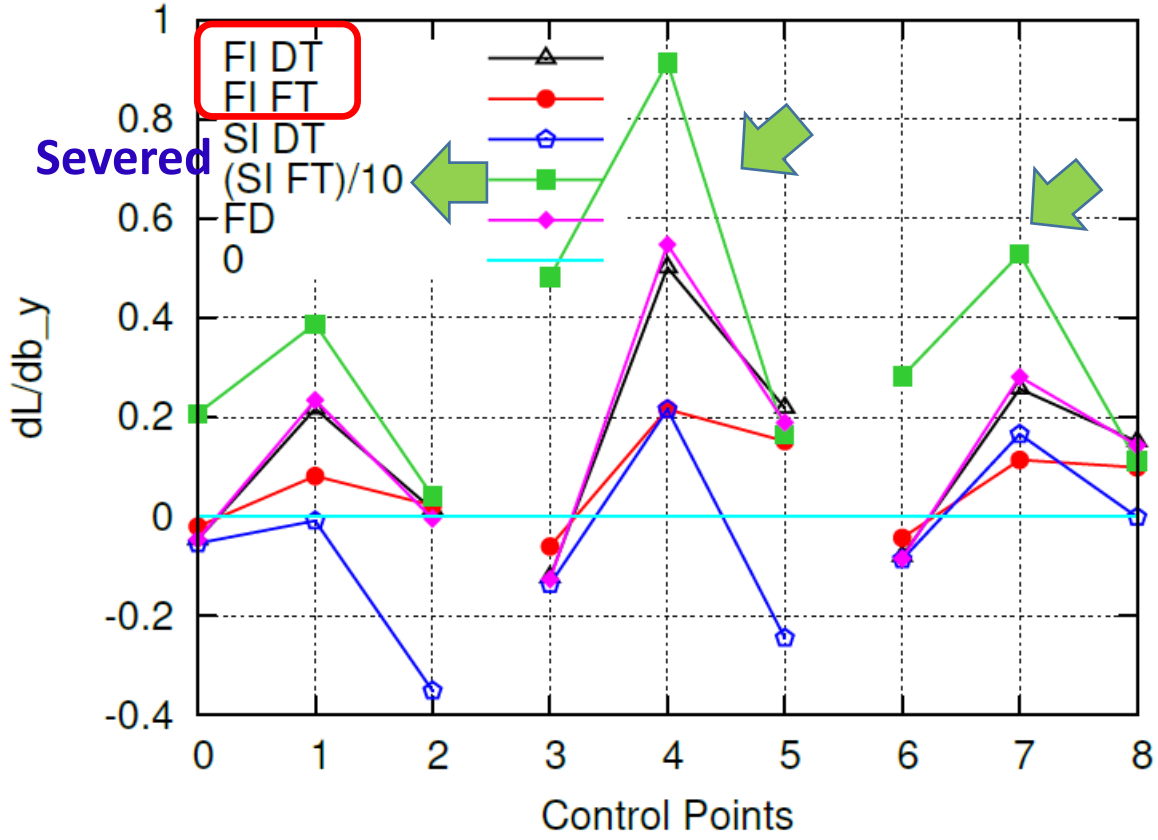
k-ω SST Model:

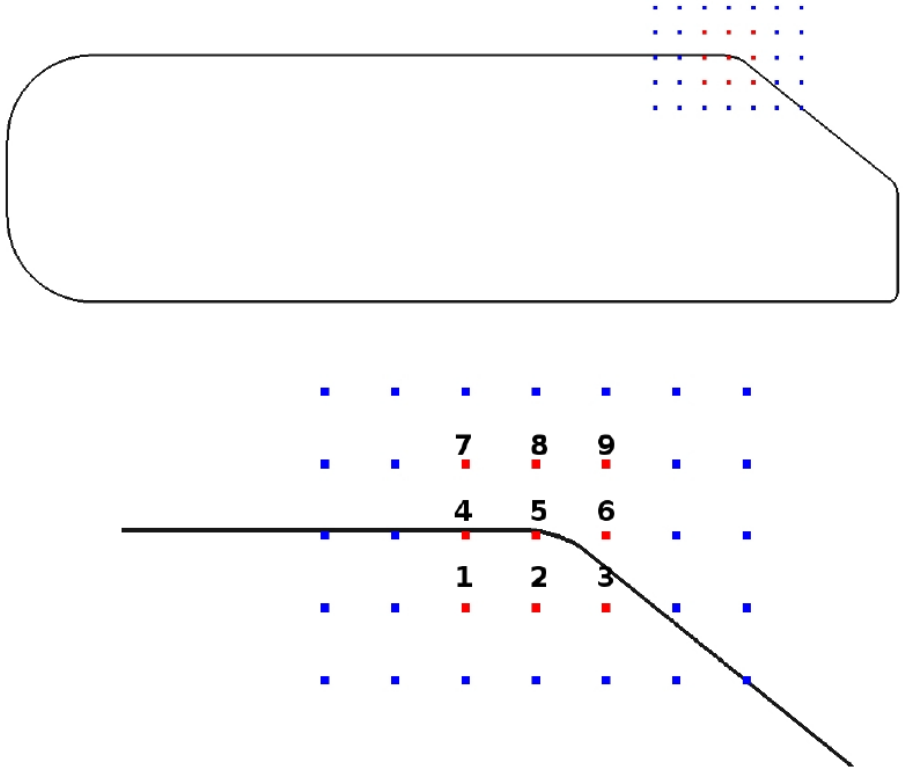
$$L = J + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega k^a R^k d\Omega + \int_\Omega \omega^a R^\omega d\Omega$$

Engineering Optimization, 47(11):1523-1542, 2015.

See more in the cited papers…

Impact of the "Frozen Turbulence Assumption" & Severed-SI Adjoint





Drag SDs for the 2D Ahmed body (Re=2.89x10^6) computed using the adjoint to the Low-Re Spalart-Allmaras model. Comparison with FD, with Differentiated Turbulence model (DT) and Frozen Turbulence model (FT). SDs (w.r.t. the y coordinates of the CPs) computed by making the "frozen turbulence" assumption and, especially, those computed by the Severed-SI adjoint are inaccurate (one order of magnitude off) even if the adjoint turbulence model equations are solved.

(Continuous) Adjoint to Turbulence Models

Hi-Re Spalart-Allmaras Turbulence Model:

Bridge the gap between the solid wall and the first node (P) or cell-center off the wall:

$$f_{WF} = y^+ - v^+ - e^{-\kappa B}\left[e^{\kappa v^+} - 1 - \kappa v^+ - \frac{(\kappa v^+)^2}{2} - \frac{(\kappa v^+)^3}{6}\right] = 0$$

where everything depends on the friction velocity

$$y_P^+ = \frac{\Delta^P v_\tau}{\nu}, \quad v_P^+ = \frac{|v_i|^P}{v_\tau}, \quad v_\tau^2 = -\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right]^f n_j t_i^I$$

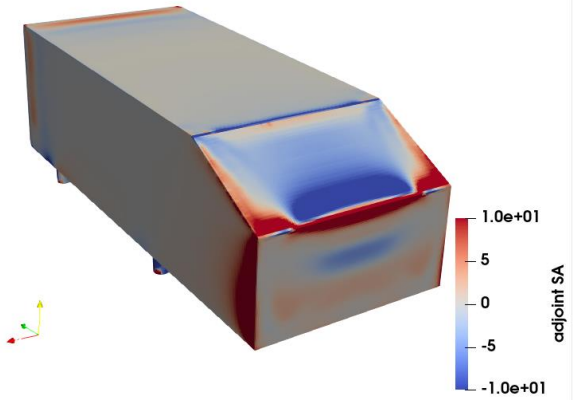
The adjoint method derives the adjoint law of the wall and the adjoint friction velocity:

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

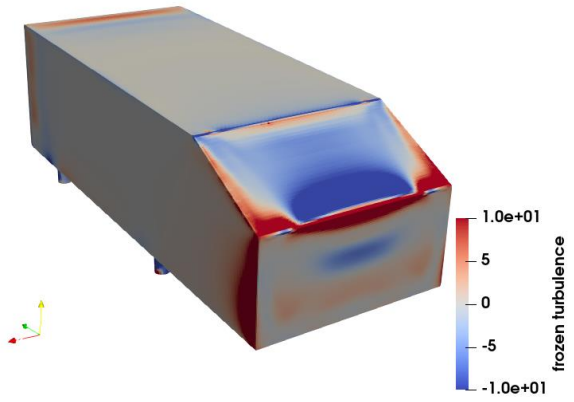
A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou and C. Othmer: 'Adjoint Wall Functions: A New Concept for use in Aerodynamic Shape Optimization', Journal of Computational Physics, 229(13): 5228-5245, 2010.

Why insisting on (Continuous) Adjoint to Turbulence Models?

Spalart-Allmaras Model

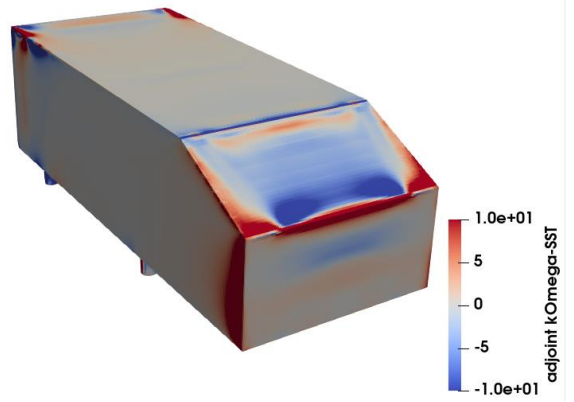


Turbulent Adjoint

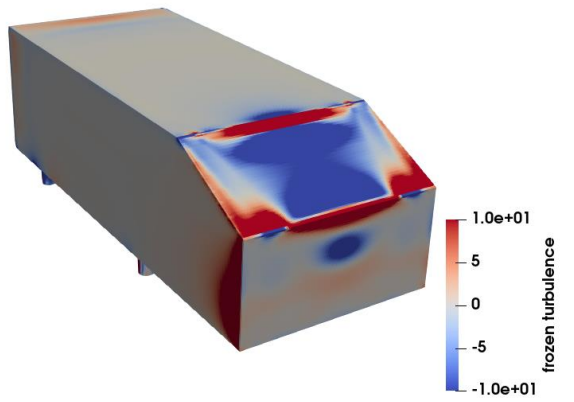


Frozen Turbulence

k-ω SST



Turbulent Adjoint

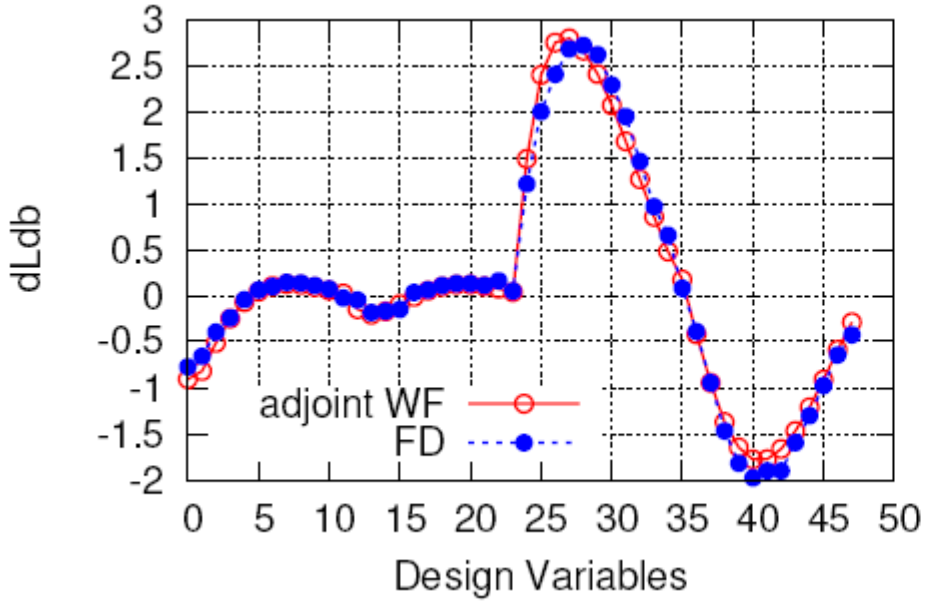


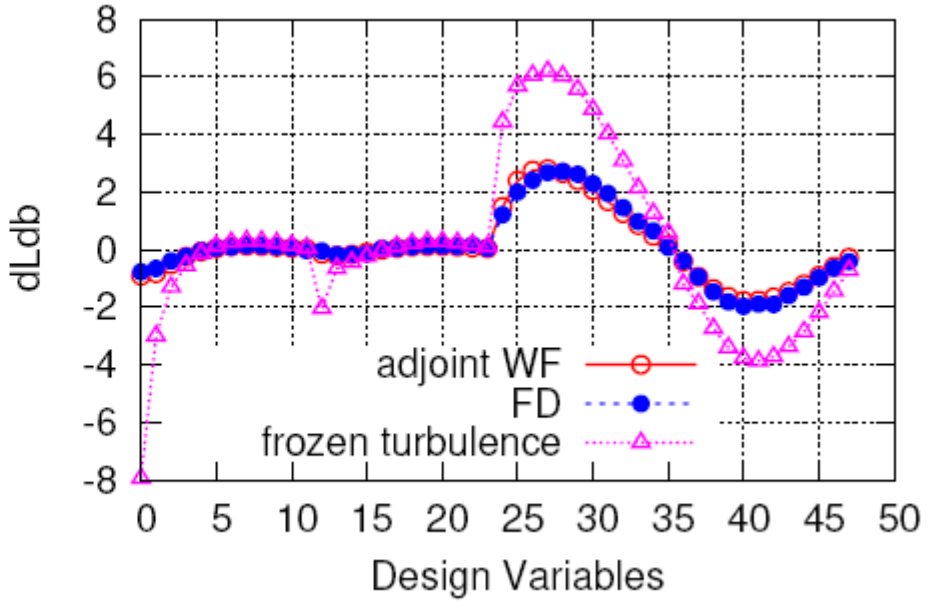
Frozen Turbulence

Drag SDs for the Ahmed body (Re=2.89x10^6) computed using the adjoint to the Low-Re Spalart-Allmaras model.

The computed sensitivity maps for drag, with or without making the frozen turbulence assumption, are quite similar if the Spalart-Allmaras model is used, but noticeably differ for the k-ω STT model. In *adjointOptimisationFoam v2206* both are available. Findings confirmed with Finite Differences, not shown here.

(Continuous) Adjoint to Turbulence Models with Wall Functions





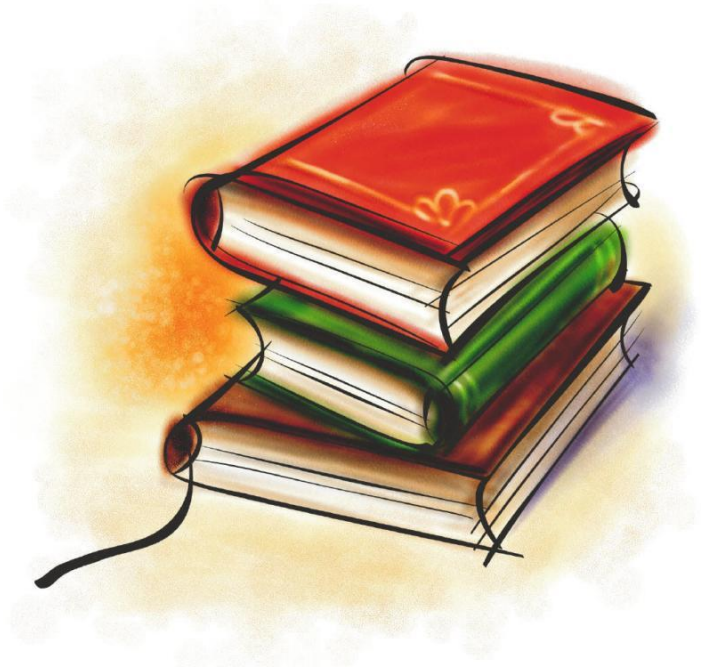
Left: Drag sensitivities (NACA0012 airfoil, Re=6x10^6, a_{inf}=2^o) computed using adjoint wall functions and FD. The first 24 points correspond to the derivatives w.r.t. the x coordinates of the suction and pressure side control points. Right: Same curves compared with those resulting from the adjoint method making the frozen turbulence assumption.

Archives of Computational Methods in Engineering, 23(2): 255-299, 2016.

PCOpt

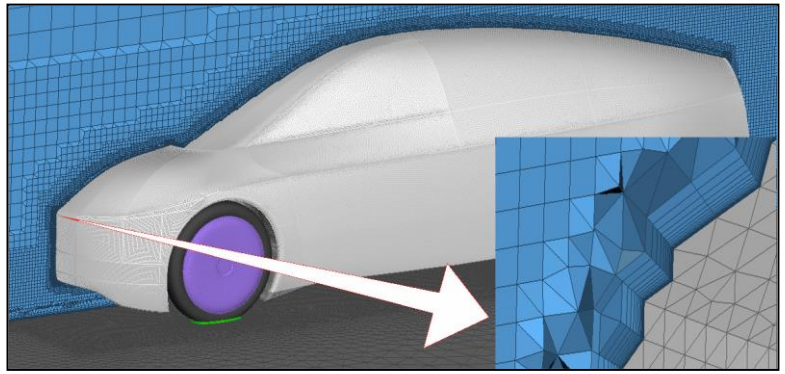
Lessons Learned

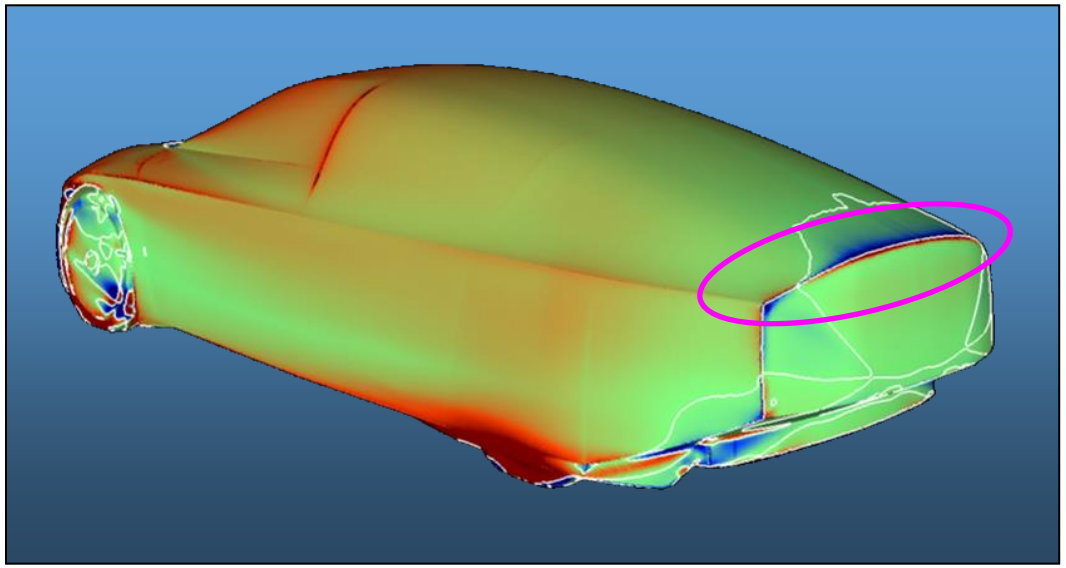
- **Don't make the frozen turbulence assumption! The adjoint to the turbulence model PDE(s) must be solved.**
- **Don't make the frozen distance assumption! The adjoint to the Eikonal eq. computing distances from the walls (if needed by the turbulence model) must be solved.**
- **Use adjoint law of the wall, if wall functions are used in the primal flow model!**
- **Don't use the Severed-SI adjoint! Use the FI or E-SI adjoint instead.**
- **In the E-SI adjoint, the adjoint Laplace GDM can safely be used in an optimization loop even if a different GDM is used to adapt the CFD grid to the updated geometries.**



Automotive Applications

ShpO of the XL1 model by VW





Plotted on the initial shape, the sensitivity map represents iso-areas of δF/δn, where n is the unit vector normal to the surface and indicates how much each part of the body surface, if deformed (inwards or outwards), contributes to the improvement of the objective function F value. The sensitivity map offers guidance, even to an experienced designer, regarding the direction of possible surface displacements, prior to running an optimisation loop.

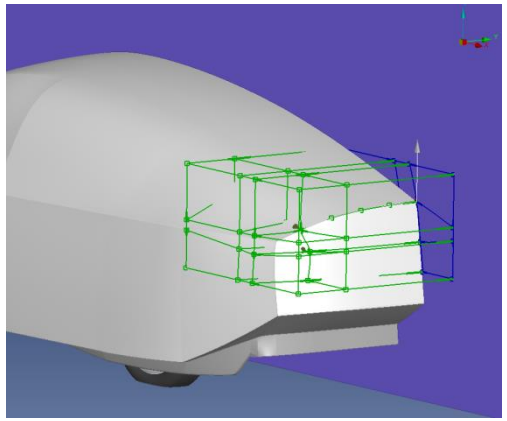
Funded by

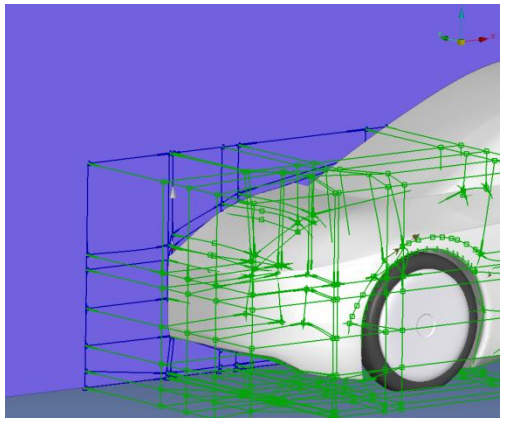
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

ShpO of the XL1 model by VW

Rear Spoiler

1



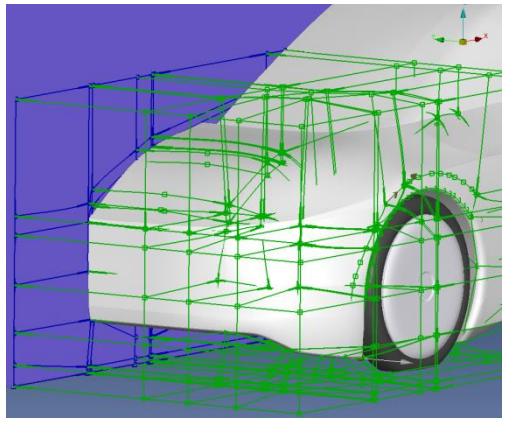


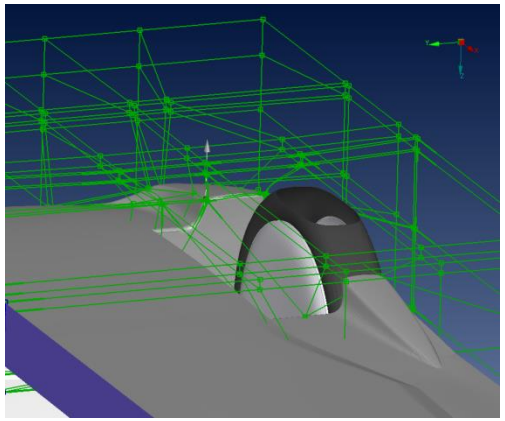
Front-end of the Bonnet

2

Front Wing

3





Front Wheel Spoiler

4

Independent parametrisation & simultaneous optimisation of 4 different parts of the car.

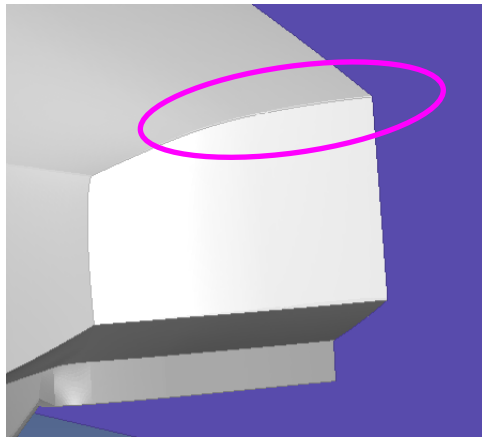
Funded by

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

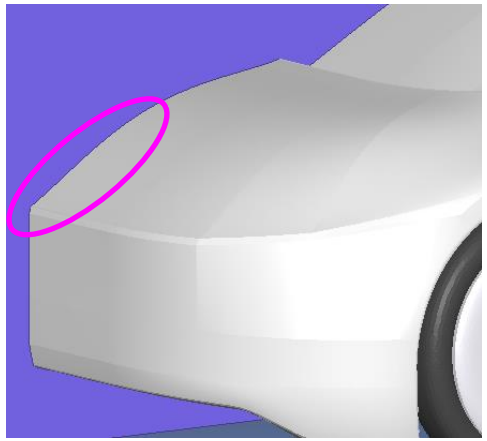
ShpO of the XL1 model by VW – Results (1/4)

Starting / Baseline Shapes:

Rear Spoiler



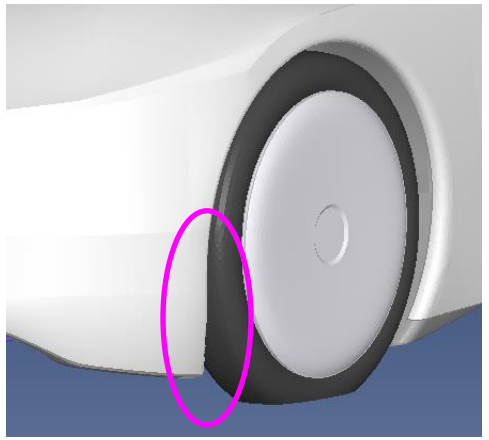
1



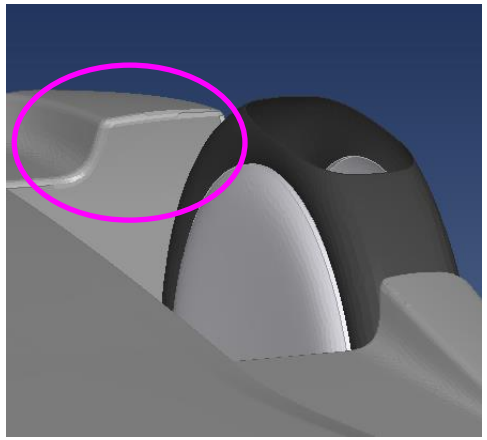
2

Front-end of the Bonnet

Front Wing



3



4

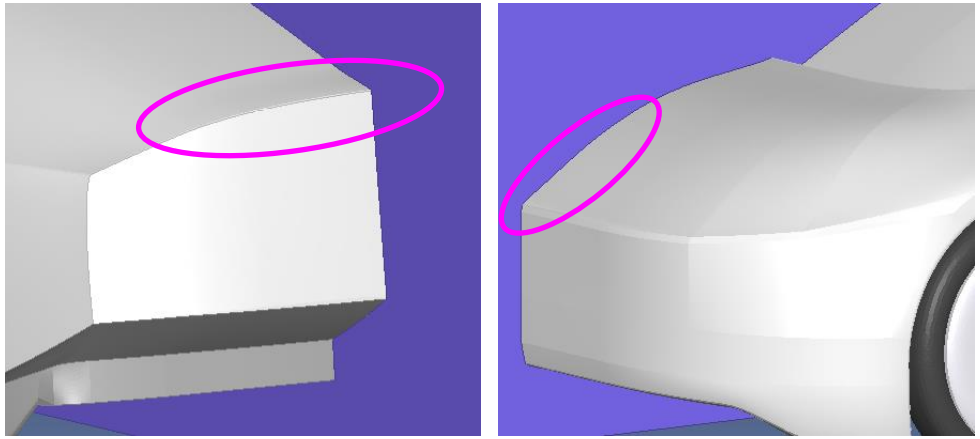
Front Wheel Spoiler

Funded by

ShpO of the XL1 model by VW – Results (2/4)

Shape after a single optimisation cycle (towards min. Drag):

Rear Spoiler

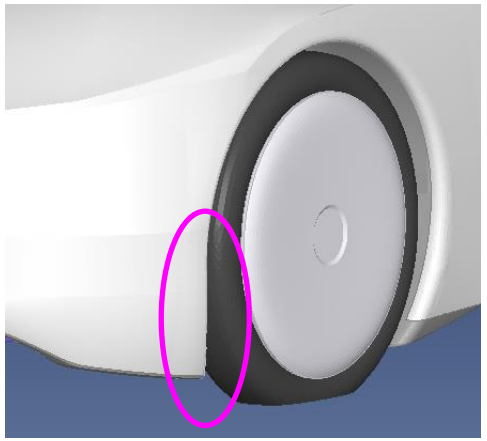


1

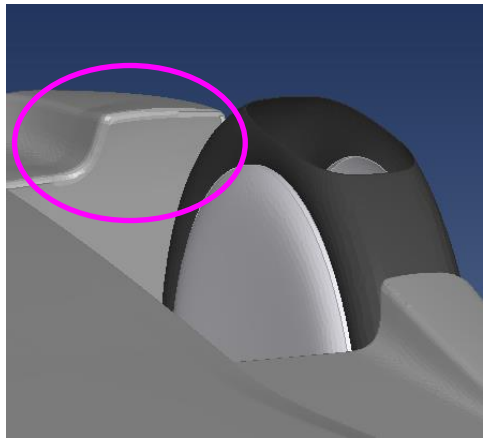
2

Front-end of the Bonnet

Front Wing



3



4

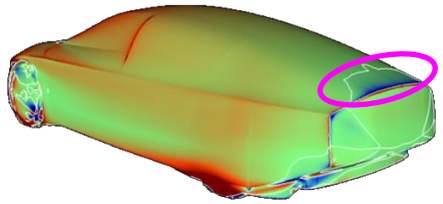
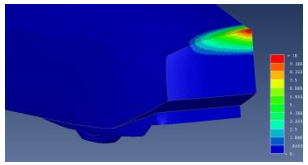
Front Wheel Spoiler

Funded by

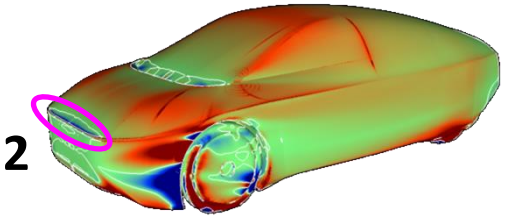
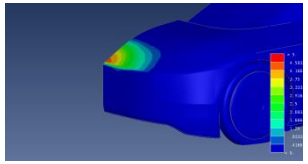
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

ShpO of the XL1 model by VW – Results (3/4)

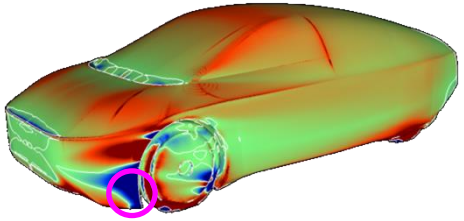
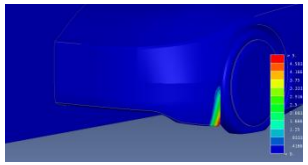
Drag Reduction (after ONLY one optimisation cycle)

1  

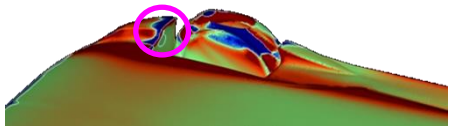
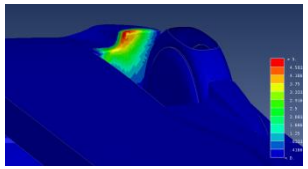
Rear Spoiler: -0,9% for 10mm displacement

2  

Front-end of the Bonnet: -0,13% for 5mm displacement

3  

Front Wing: -0,6% for 5mm displacement

4  

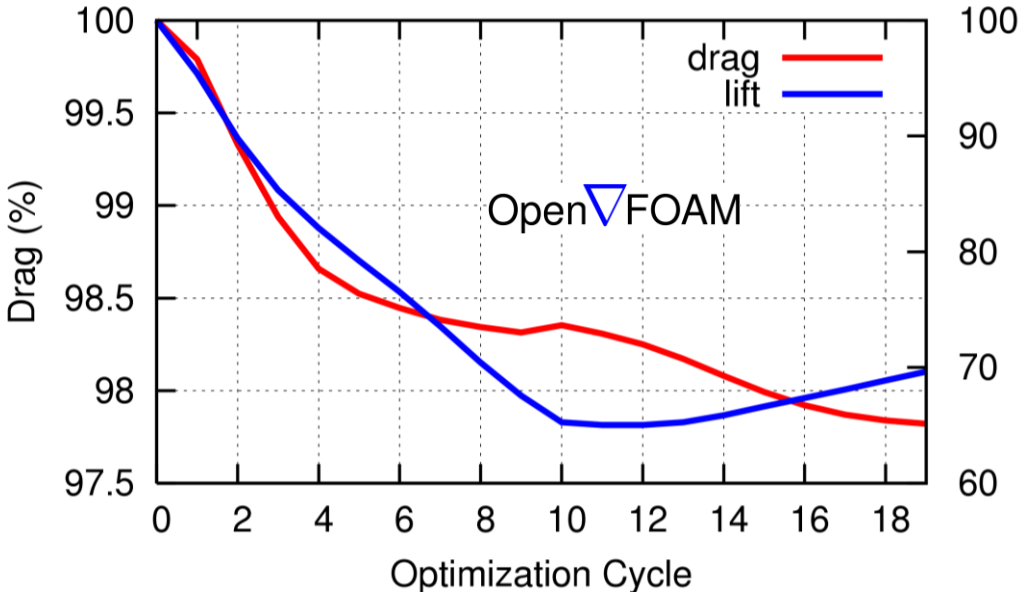
Front Wheel Spoiler: -0,8% for 10mm displacement

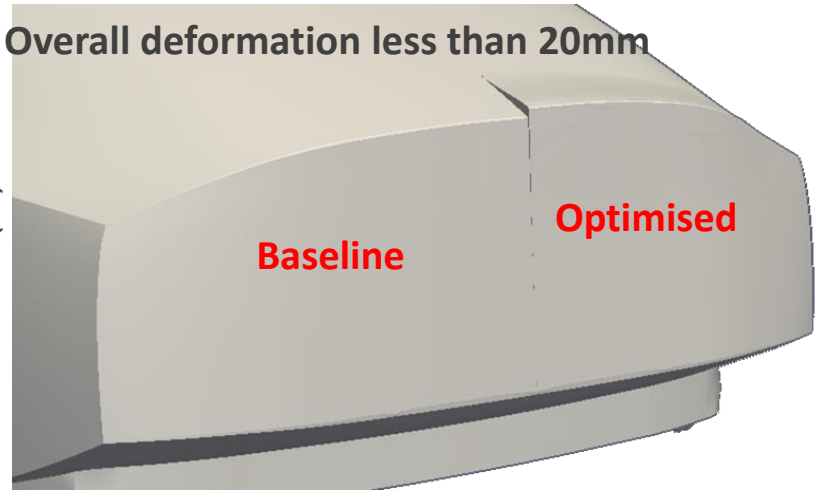
Funded by

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

ShpO of the XL1 model by VW – Results (4/4)

Running an optimisation loop, for min. Drag





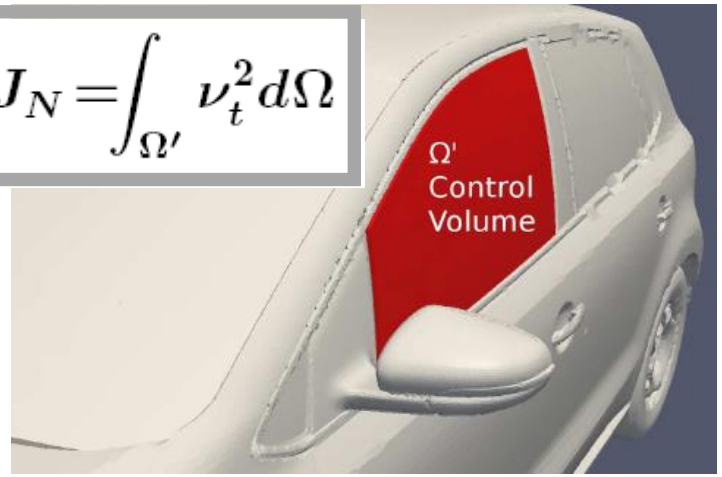
- *>2% drag reduction (great for an aerodynamically already nearly perfect car)*
- *30% improved down-force (not included in target)*
- *Overall cost: <5 flow (CFD) solutions*

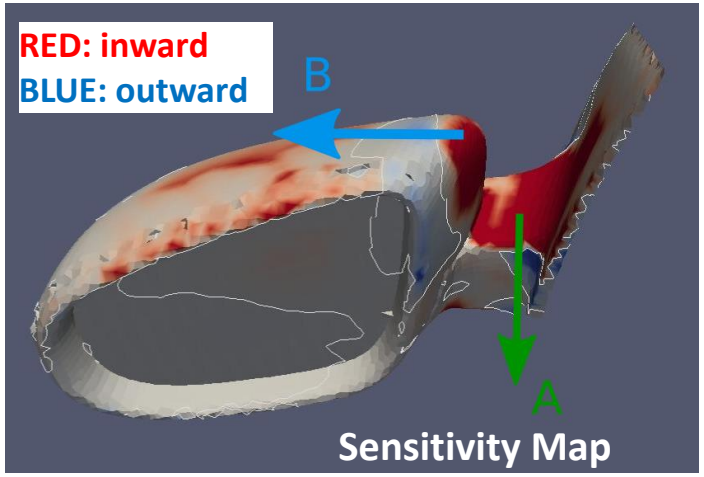
Funded by

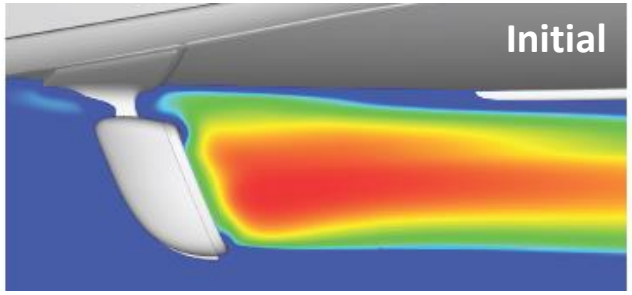
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

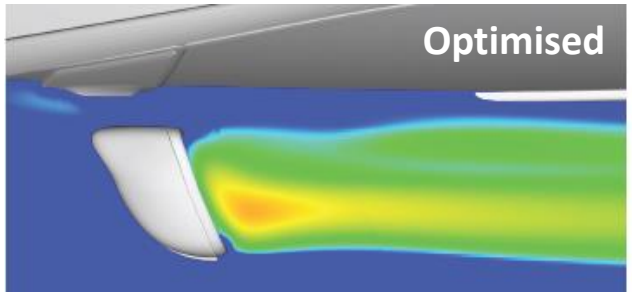
ShpO of the Side Mirror of a Car for Noise Reduction

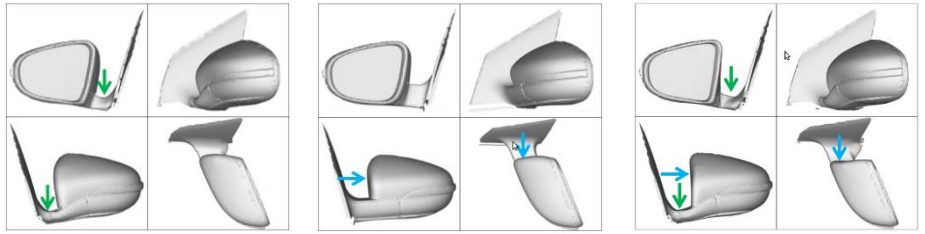
$$J_N = \int_{\Omega'} \nu_t^2 d\Omega$$



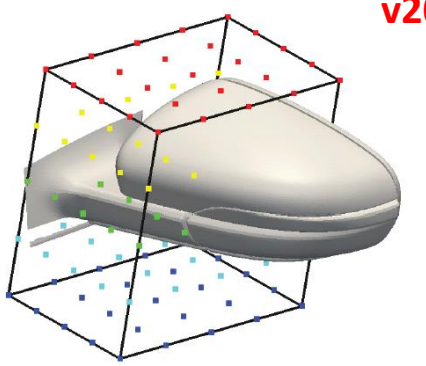


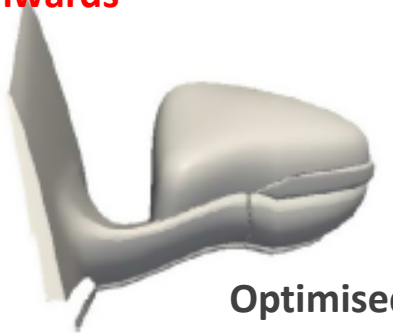






Code available in OpenFOAM v2006 onwards



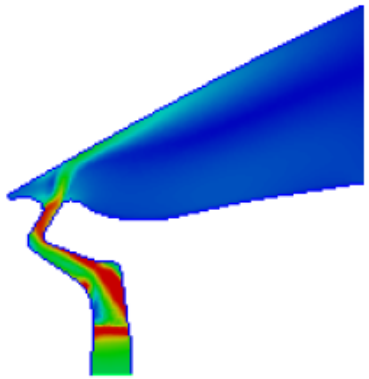


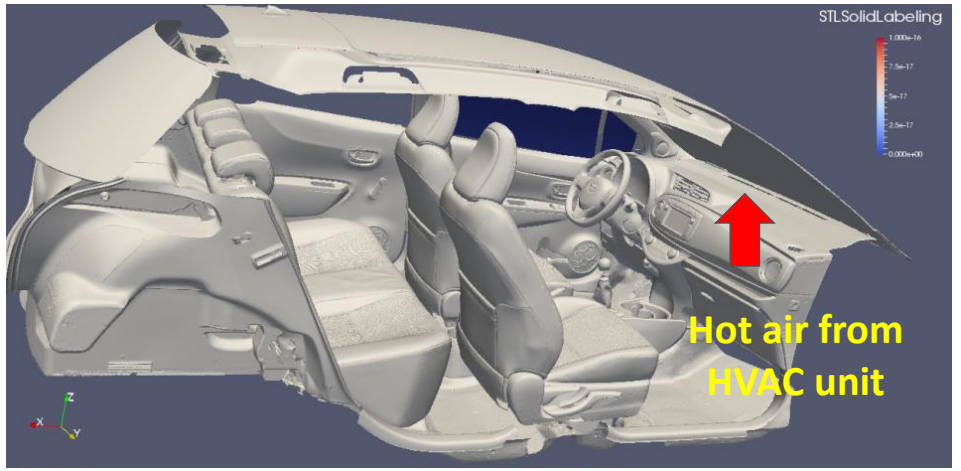
Minimisation of the noise perceived by the driver. A turbulence-based objective function is used, so this problem cannot be solved without the adjoint to the turbulence model equations. A relatively coarse grid (2.4 Mi cells) & the Volumetric B-Splines morpher (81 DoFs) were used in the adjoint optimisation. A re-evaluation of the optimal solution on a fine grid (31 Mi cells) confirmed a reduction in J_N by 25%.

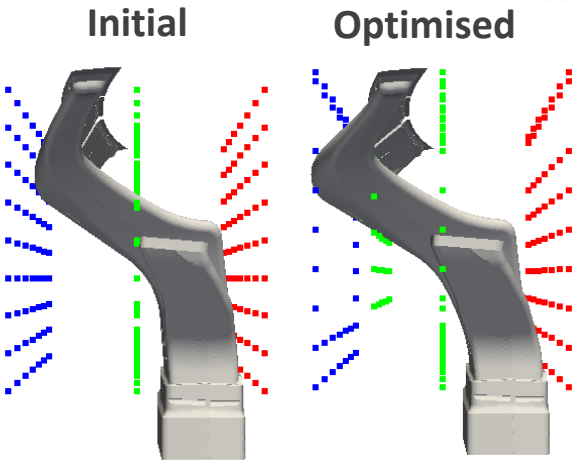
Computers & Fluids, 122:223-232, 2015.

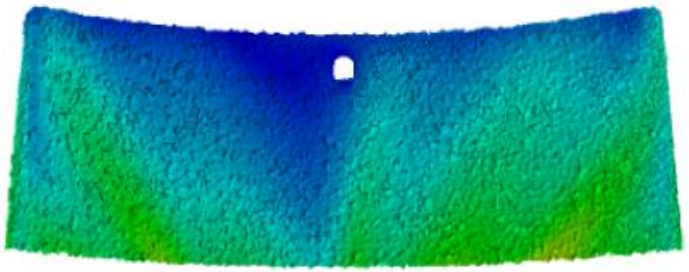
Application funded by

ShpO of the Defroster Nozzle of the HVAC unit of a Car

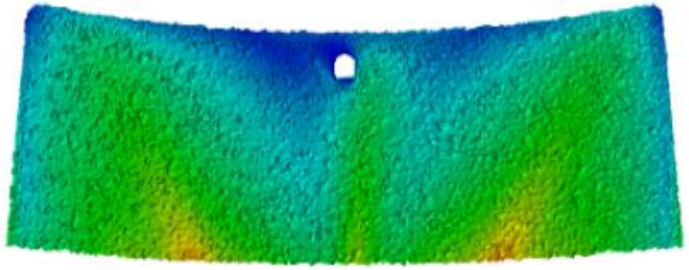








Initial



Optimised

$$J_V = \frac{1}{2}\int_{\Omega'} \left(v - v^{target}\right)^2 d\Omega$$

ShpO of the defroster nozzle of the HVAC unit of a TOYOTA passenger car, for improved defrosting/ demisting performance of the windshield. The objective was to shorten the time for dispelling condensation or frost on the windshield in the most uniform way by reaching a certain air velocity close to the windshield. The optimised geometry was manufactured (3D printing) and submitted to a defrost test in the TME's climate chamber (@ -20°), leading to 15% less windshield defrost time.

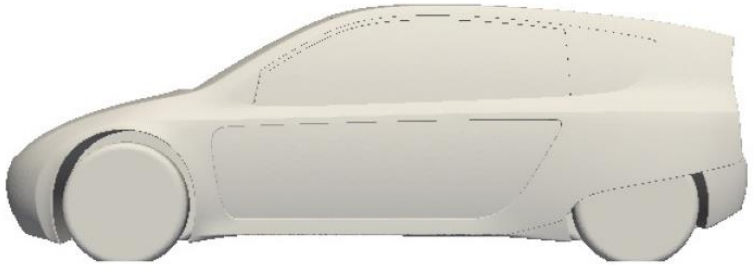
Green areas in the velocity isolines' plot on the windshield correspond to v^{target}.

Application funded by TOYOTA

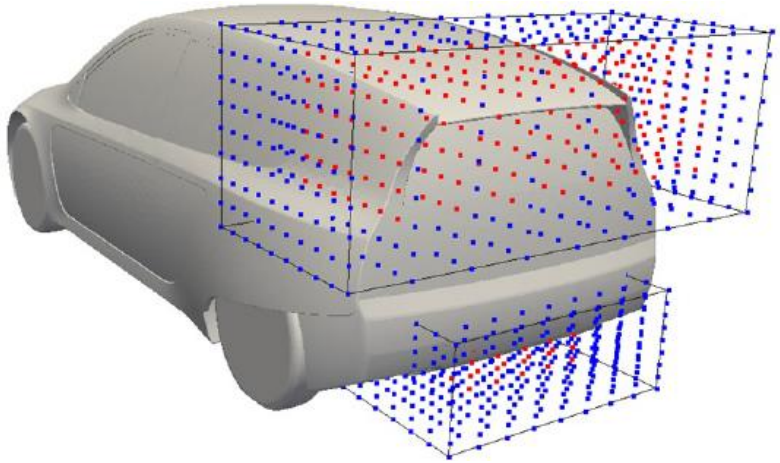
Multi-Point Aerodynamic ShpO of a Concept Car

- **ShpO of an ultra-lightweight vehicle, designed by the Toyota Aerodynamic Dept., for making it less sensitive to side-wind (30° from the port side; min. yaw moment), while maintaining a very low drag at 0° (longitudinal wind).**
- **Objective function:**

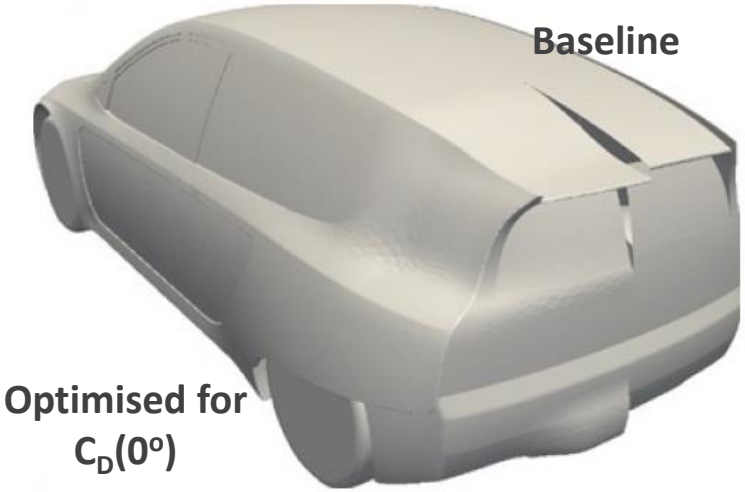
$$J = \omega_D C_D^{0^o} + \omega_M C_M^{30^o}$$



- **Pareto front computed by optimizing with different values of (ω_D,ω_M).**
- **Two simultaneously acting morphing boxes at the spoiler and diffuser areas.**
- **RANS-based adjoint optimisation using the adjoint to the Spalart-Allmaras model (with wall functions) & a coarse polyhedral grid with ~1.6Mi cells (average y^+=32 at the first cell-centers off the walls).**

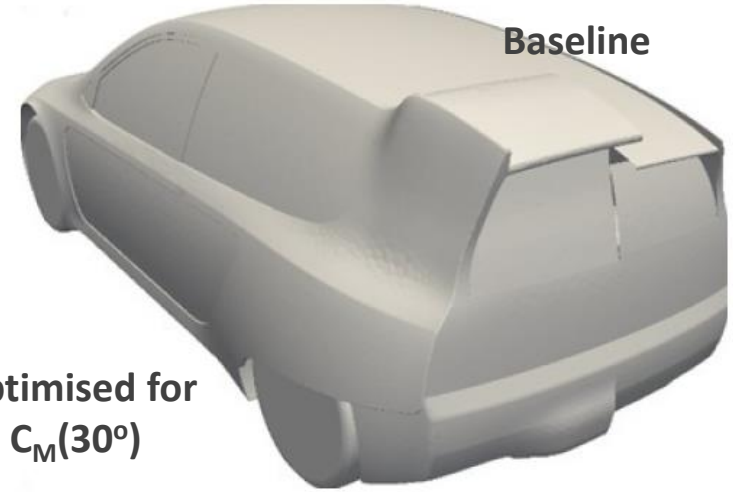


Baseline



Optimised for $C_D(0^o)$

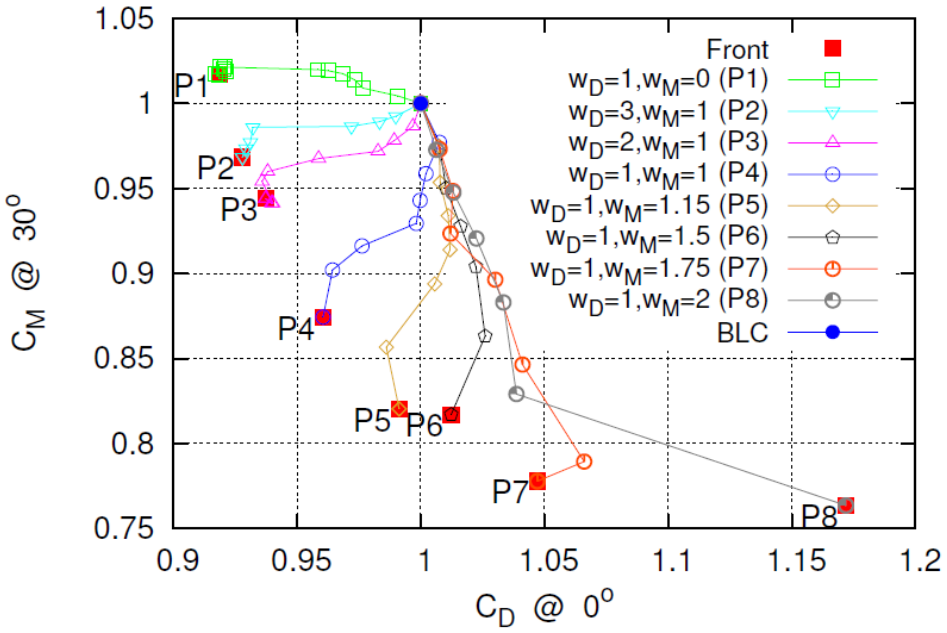
Baseline

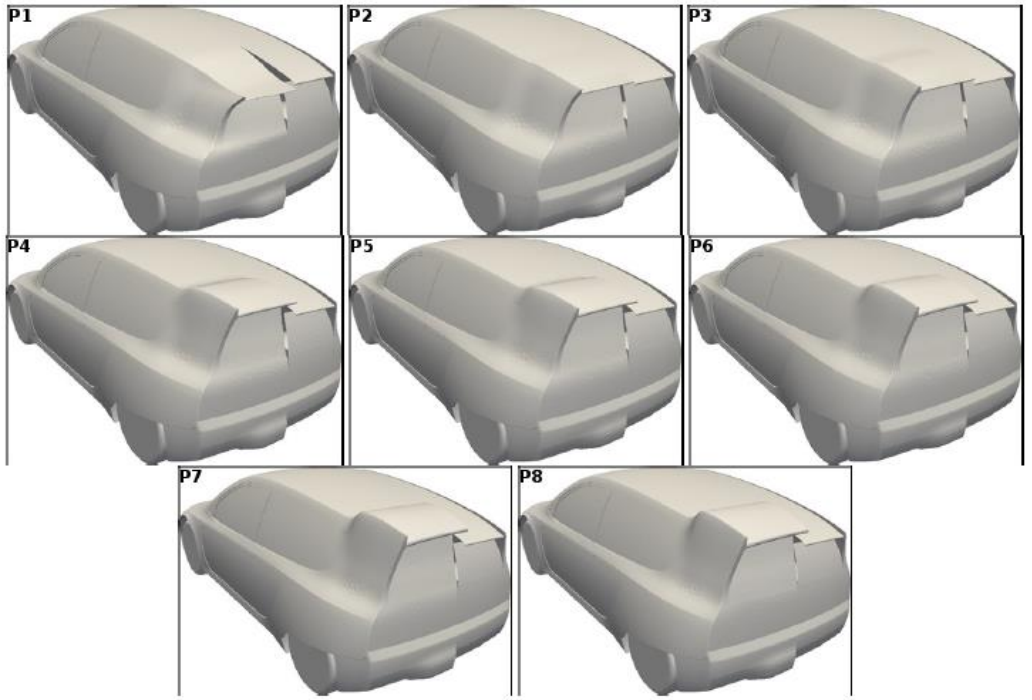


Optimised for $C_M(30^o)$

Application funded by TOYOTA

Multi-Point Aerodynamic ShpO of a Concept Car





Optimised geometries (port side) compared to the BLC (starboard side). Drag reduction at 0° results from a lowered spoiler, boat-tailing and a prolonged and widened diffuser. In contrast, yaw moment reduction at 30° comes mainly from the increased spoiler height and the slight widening of the car; these increase pressure on the port side and decrease it on the starboard side to counter-balance the yaw moment of the side-wind.

Structural and Multidisciplinary Optimization, 59(2): 675–694, 2019.

Application funded by TOYOTA

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

What is a Detached Eddy Simulation Model in Car Applications?

Detached Eddy Simulation (DES) appears as an improved alternative to RANS models. The DES model switches to a subgrid scale (SGS) formulation in regions of the flow domain that are fine enough to allow large eddy simulation (LES) calculations.

Near solid boundaries, the turbulent length scale is less than the cell size, so the flow is modeled using RANS. In areas where the turbulent length scale exceeds the cell size, the regions are solved using the LES mode. This "hybrid model" reduces grid finesse requirements and, thus, cuts down the computational cost.

Originally, the DES concept was built around the widely used Spalart-Allmaras one-equation turbulence model, where the transition to an SGS closure was merely done by altering the length scale that appears in the turbulent transport equation of the model variable.

The Delayed DES (DDES) additionally makes use of a delay function to prevent premature switching to LES within the boundary layer.

Spalart. Detached-Eddy Simulation. Annual Review of Fluid Mechanics, 41(1):181–202, 2009.

URANS – DES – DDES Variants of the Spalart-Allmaras Turbulence Model

URANS-SA	**DES-SA**	**DDES-SA**
$\mathcal{P}=c_{b_1}\widetilde{S}$	$\mathcal{P}=c_{b_1}\widetilde{S}\,(1-f_{t_2})$	$\mathcal{P}=c_{b_1}\widetilde{S}\,(1-f_{t_2})$
$\mathcal{D}=\dfrac{c_{w_1}f_w\widetilde{\nu}}{\widetilde{d}^2}$	$\mathcal{D}=\left(c_{w_1}f_w-\dfrac{c_{b_1}}{\kappa^2}f_{t_2}\right)\dfrac{\widetilde{\nu}}{\widetilde{d}^2}$	$\mathcal{D}=\left(c_{w_1}f_w-\dfrac{c_{b_1}}{\kappa^2}f_{t_2}\right)\dfrac{\widetilde{\nu}}{\widetilde{d}^2}$
$\widetilde{d}=y$	$\widetilde{d}=y-\max(0,y-\Psi c_{DES}\Delta)$	$\widetilde{d}=y-f_d\max(0,y-\Psi c_{DES}\Delta)$
	$f_{t_2}=c_{t_3}e^{-c_{t_4}\chi^2}$	$f_{t_2}=c_{t_3}e^{-c_{t_4}\chi^2}$
		$f_d=1-\tanh[(c_{d_1}R_d)^{c_{d_2}}]$

All of them require demanding unsteady computations.

Multi-Point Aerodynamic ShpO of a Concept Car

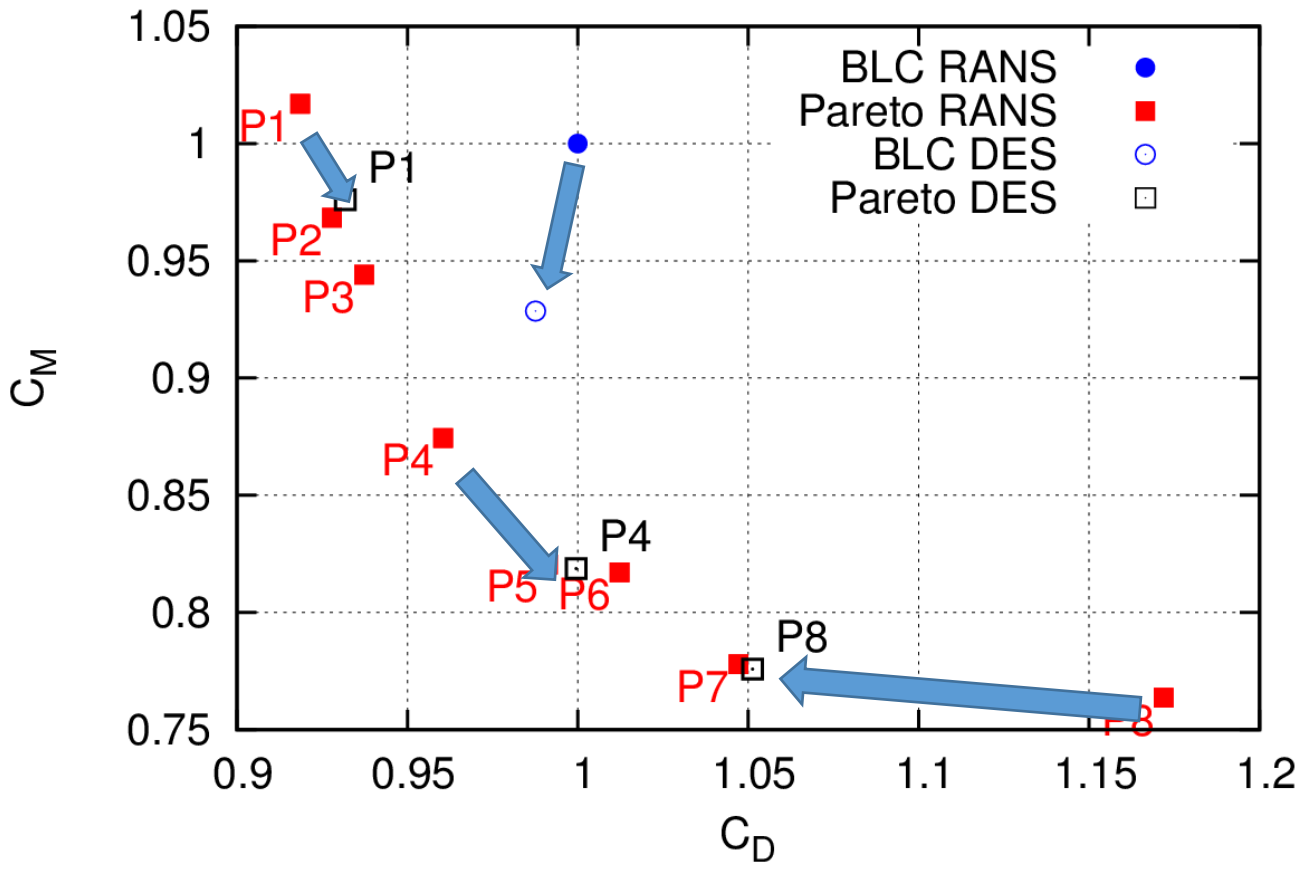
- **RANS-based adjoint ShpO** using the adjoint to the Spalart-Allmaras model & wall functions:

  Cost for computing 8 points on the Pareto front: 84h on 40 cores (3340 CPUh)

- **DDES-based re-evaluation**, on a polyhedral grid with ~30Mi cells. Integrated over 4sec., with $\Delta t=10^{-4}$sec., initialised from the RANS solution. C_D, C_M averaged over the last 3sec:

  Cost of a single DDES simulation: 50h on 200 cores (10000 CPUh)

Hardware: Intel Xeon E5-2680v2 @2.8 GHz



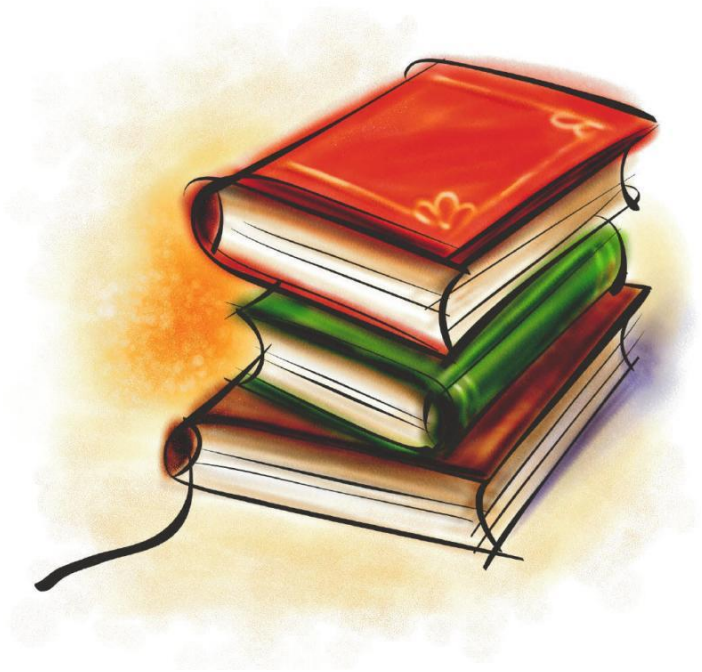
Structural and Multidisciplinary Optimization, 59(2): 675–694, 2019.

Application funded by TOYOTA

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

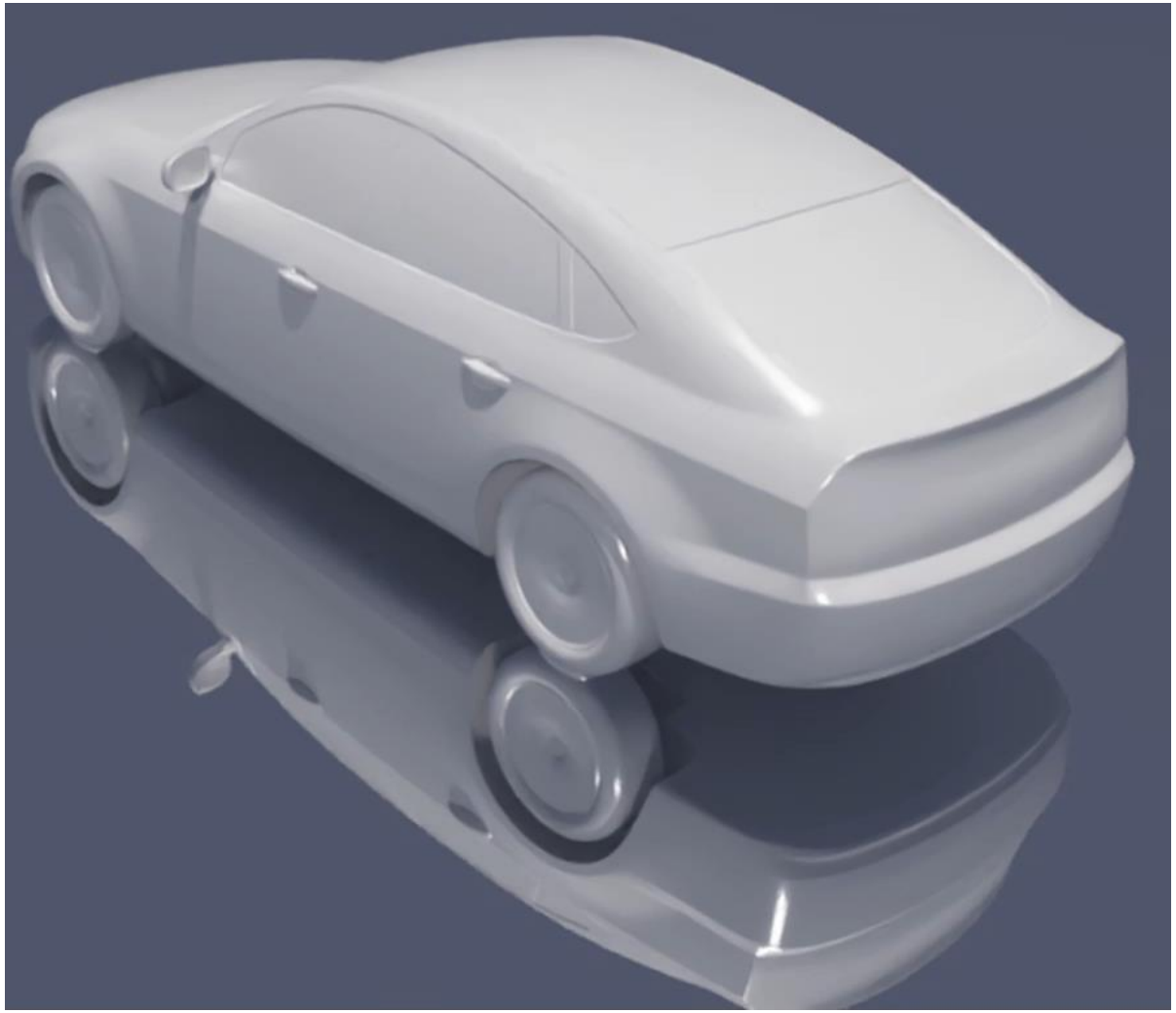
Current status of adjointOptimisationFoam

OpenFOAM version	Features
v1906	• Adjoint to incompressible, steady-state flows • Differentiation of the Spalart-Allmaras turbulence model • Computation of sensitivity maps with the E-SI approach
v1912	• Surface and volume parametrisation using volumetric B-Splines • Automated shape optimisation loops • Computation of sensitivity derivatives using the FI approach
v2006	• New objective function related to the qualitative evaluation and minimisation of noise • Sensitivity contributions from rotating boundaries
v2112	• Smoothing of sensitivity maps
v2206	• Adjoint to the k-ω SST turbulence model
v2212	• Objective functions for internal aerodynamics (flow rate, flow rate distribution, uniformity, power losses)
v2312	• Density-based and level-set topology optimisation, support for inequality constraints



Adjoint with Control on the Body Skin Itself (Nodal Parametrisation)

Optimisation based on Nodal Parametrisation



The "direct" control of surface nodes, based on sensitivity maps, will likely lead to jagged shapes.

Instead, nodal parametrisation assisted by regularisation schemes, such as the Laplace-Beltrami and p-Laplacian PDEs, can be used, leading to smooth geometries.

ShpO based on Nodal Parametrisation

- **ShpO by displacing the surface nodes leads to the richest possible design space!**
- **To generate smooth/manufacturable optimised shapes, regularization is needed.**
- **A well-performing scheme is to embed regularisation into the parametrisation itself, rather than just applying it to the computed field of sensitivities; constraint handling becomes easier this way!**
- **The regularisation scheme should be differentiated in adjoint-driven ShpO.**

E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

ShpO based on Nodal Parametrisation

- **Design variables: virtual displacement of <u>boundary faces</u>**

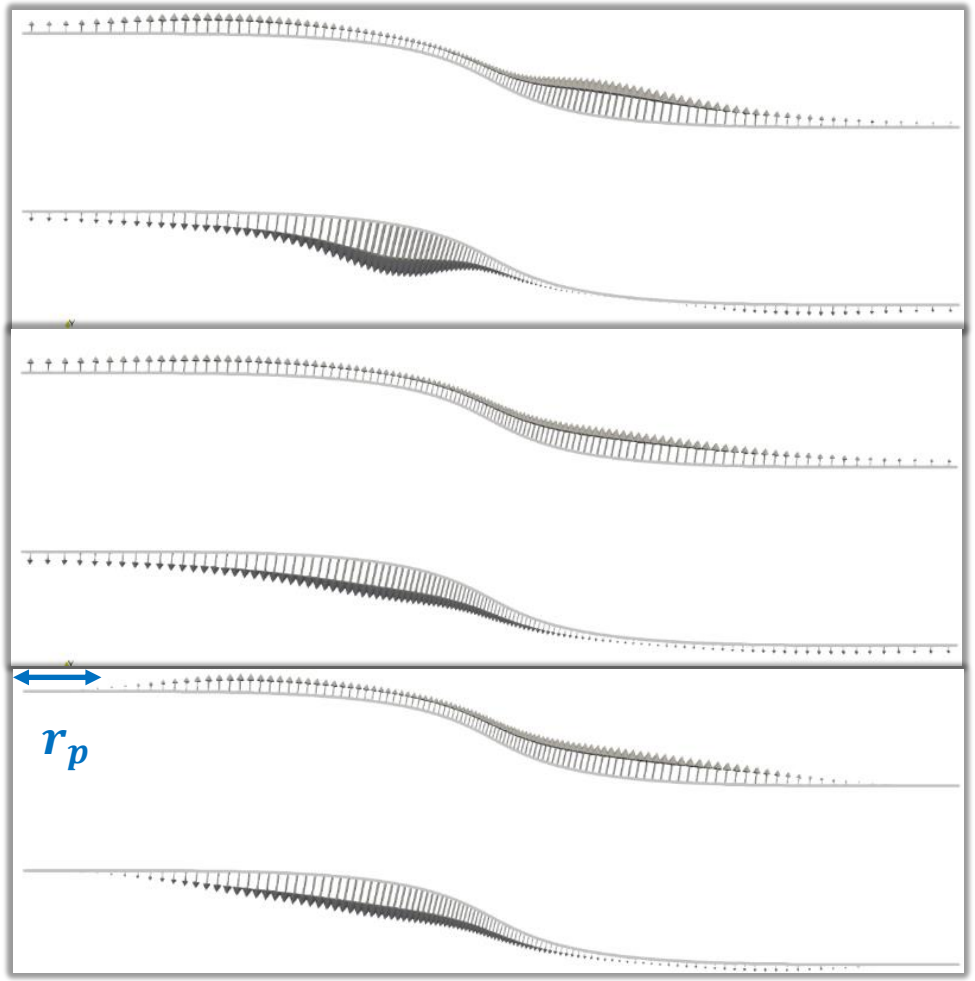
$$b_f \, , f \in [1, N_f]$$

- **Regularisation of boundary displacements**

$$m_f^k, k \in [1, 3], f \in [1, N_f]$$

- **Proximity smoothing: smoothly fade out displacements, close to "still" patches**

$$\widetilde{m_f}^k \, , k \in [1, 3], f \in [1, N_f]$$



E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

Regularisation PDEs

Laplace-Beltrami (LB)

$$R_i^{r,LB} = -r_{LB}^2 \frac{\partial^2 m_i}{\partial x_{s,j}^2} + m_i - b_i = 0$$

PDEs constrained on the active part of the boundary & solved using the Finite Area Method

p-Laplacian (pL)

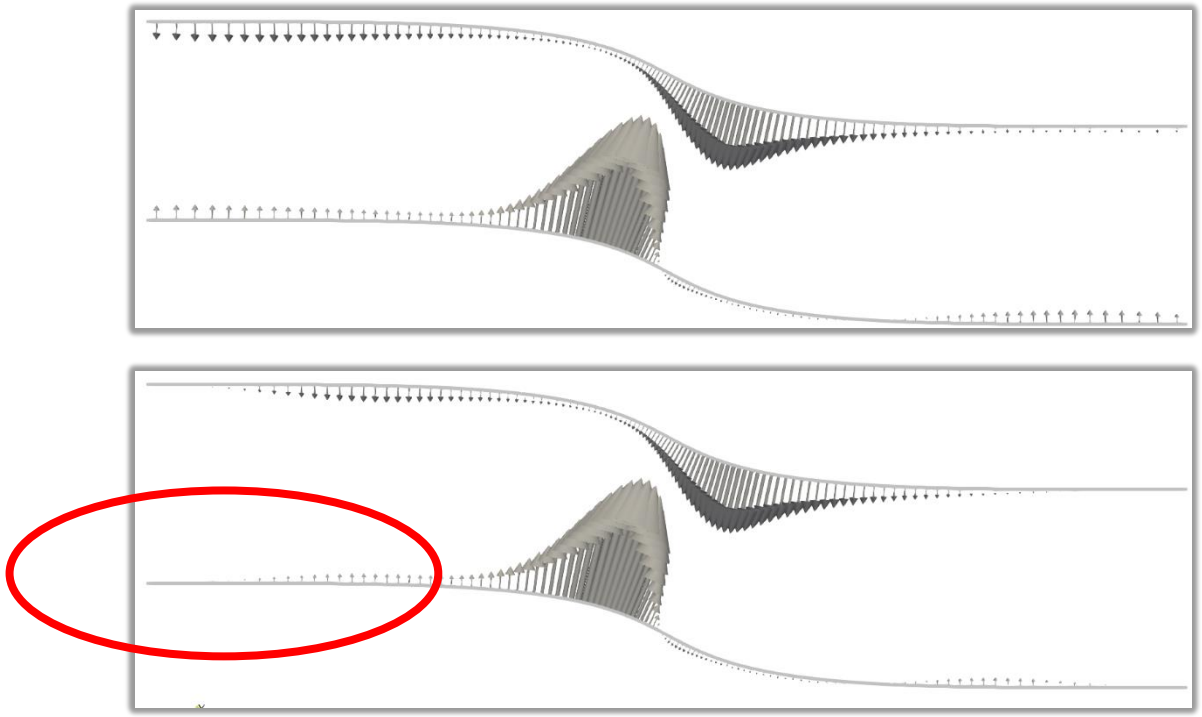
$$R_i^{r,pL} = \frac{\partial}{\partial x_j}\left[|\nabla \vec{m}|^{p-2}\frac{\partial m_i}{\partial x_j}\right] = 0,$$

$$r_{pL}\,|\nabla \vec{m}|^{p-2}\frac{\partial m_i}{\partial x_j} n_j + m_i - b_i = 0$$

PDEs solved on the entire domain using the Finite Volume Method

E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

Proximity Smoothing



Proximity smoothing is based on a function multiplying displacements, based on geodesic distances from the interface of movable and fixed parts of the geometry.

E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

Computation of Sensitivity Derivatives (SDs)

- **SDs w.r.t. the face displacement →** $\frac{\delta J}{\delta \tilde{m}_i}$

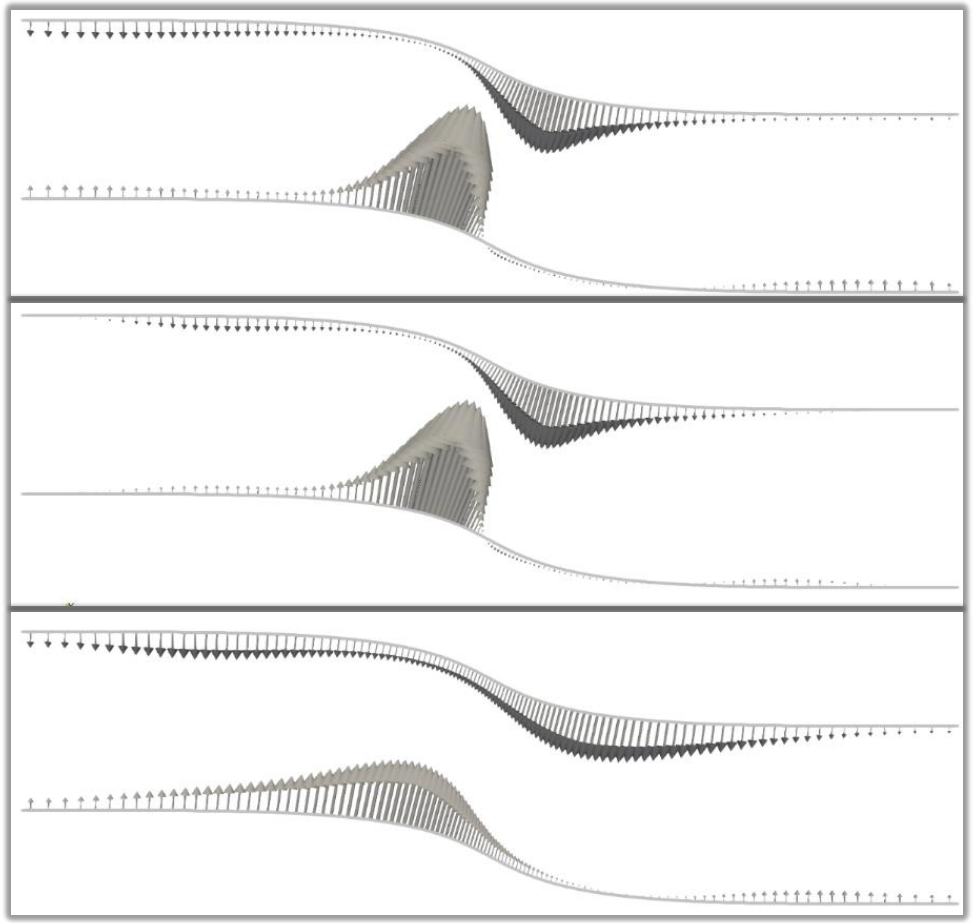
  Involves the adjoint to the flow equations

- **Differentiation of the proximity smoothing →** $\frac{\delta J}{\delta m_i}$

  Mulplication with the derivative of the proximity function

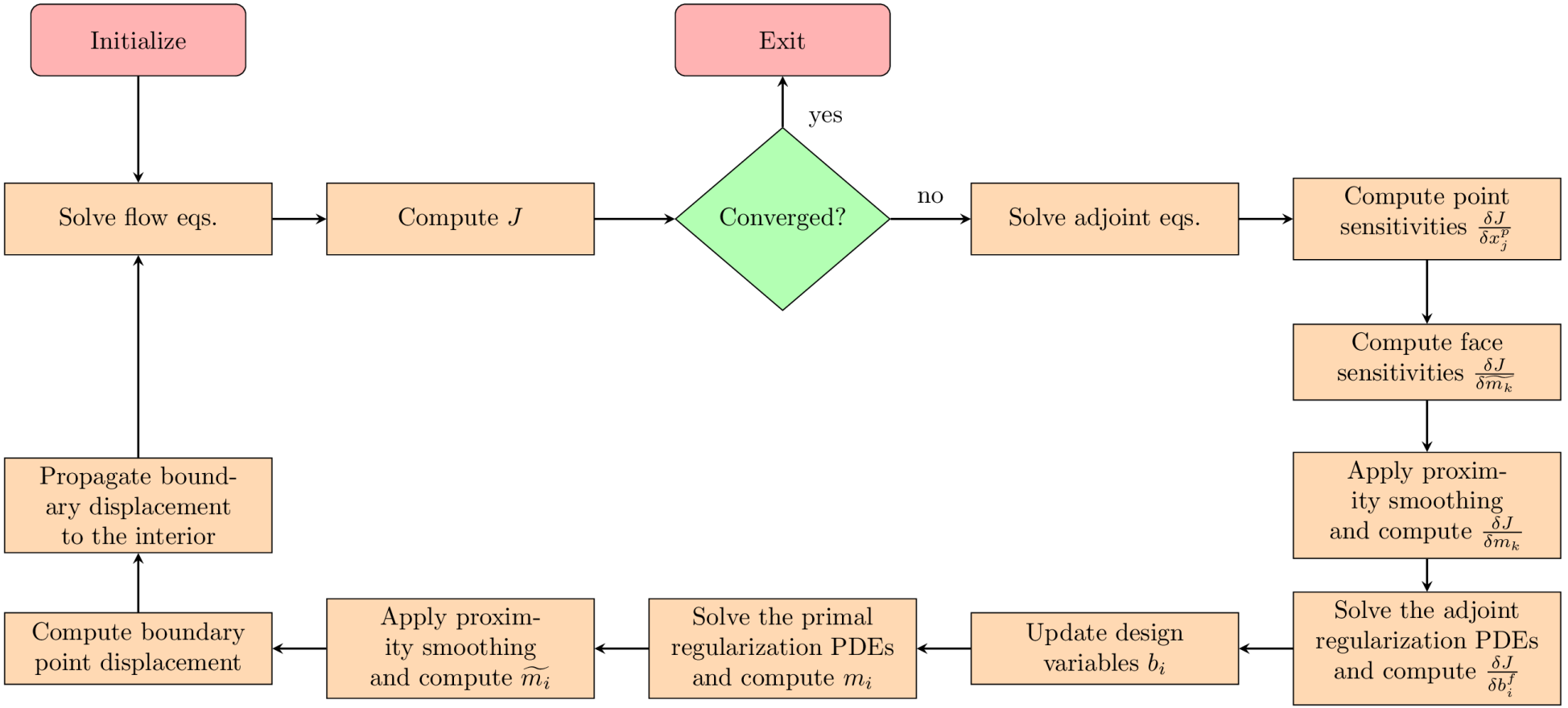
- **SDs w.r.t. the design variables →** $\frac{\delta J}{\delta b_i}$

  Solution of the adjoint to the regularization PDEs



E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

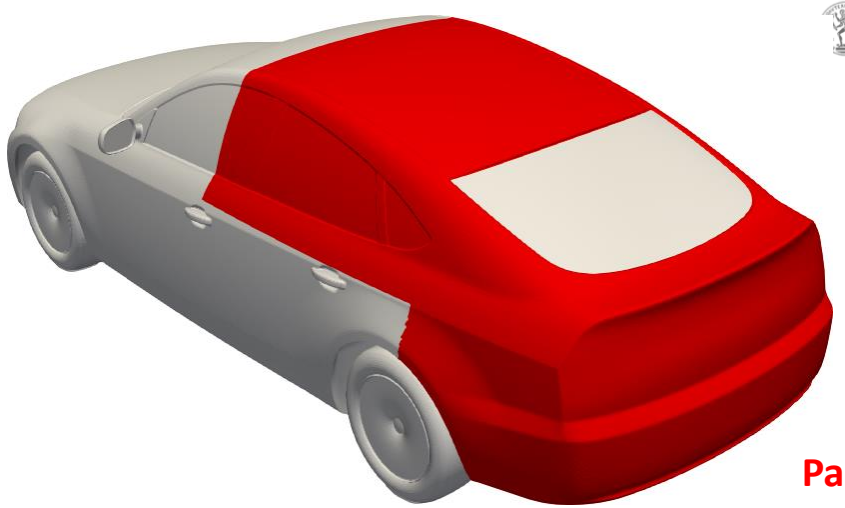
ShpO Workflow



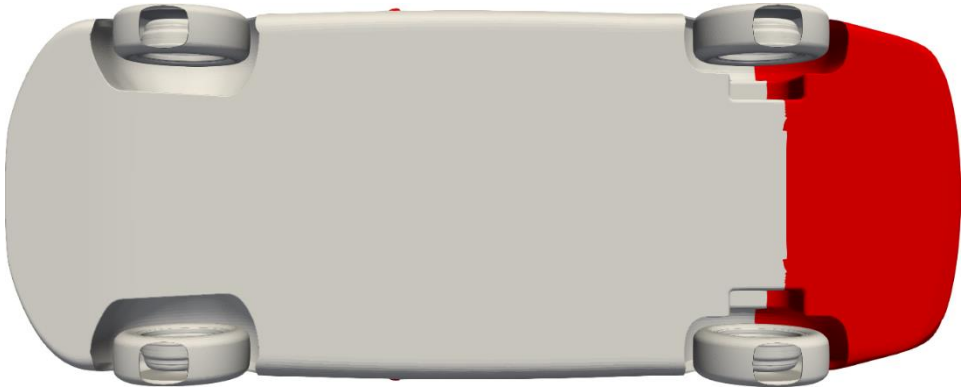
E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

ShpO of the Drivaer Car Model with Nodal Parametrisation

- **Turbulent flow,**
  RANS Spalart-Allmaras turbulence model, steady-state solver
- **V = 38.89 m/s, moving road, rotating wheels**
- **5.3M cells**
- **Min. Drag coefficient, averaged in the last 3500 iterations of each optimization cycle**
- **~40k faces on the parametrized part → 120k design variables**
- **Update computed with steepest descent**



Parametrised part



E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

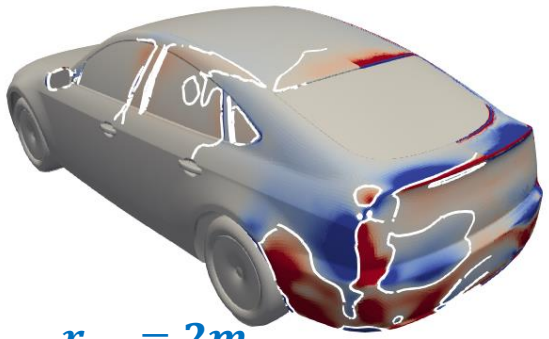
Impact of the regularization radius – Sensitivity Map

Laplace - Beltrami

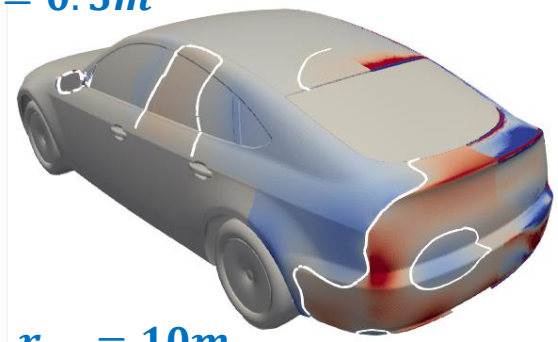
Red: move outwards
Blue: move inwards

p-Laplacian

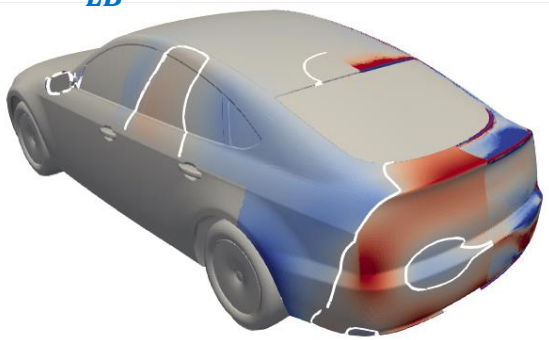
$r_{LB} = 0.1m$



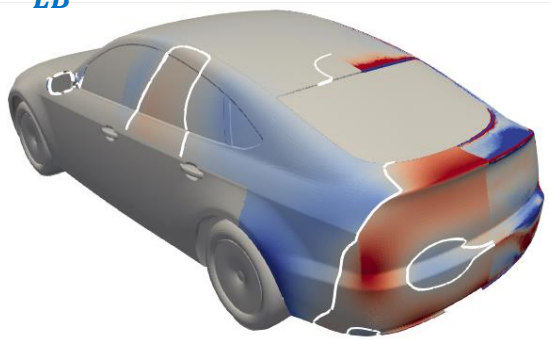
$r_{LB} = 0.5m$



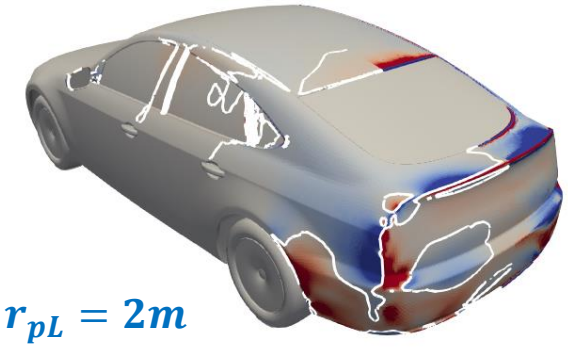
$r_{LB} = 2m$



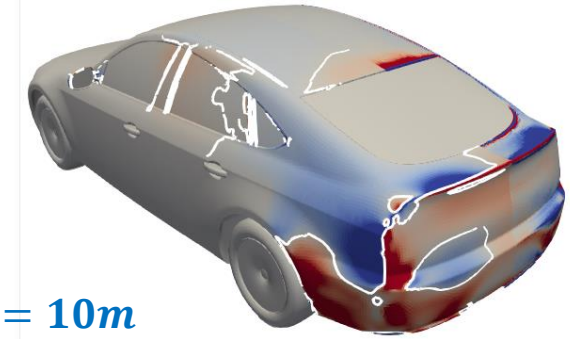
$r_{LB} = 10m$



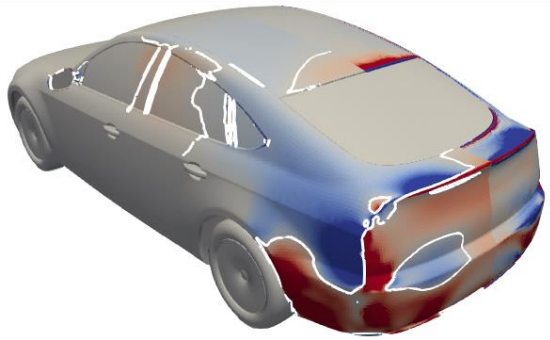
$r_{pL} = 0.1m$



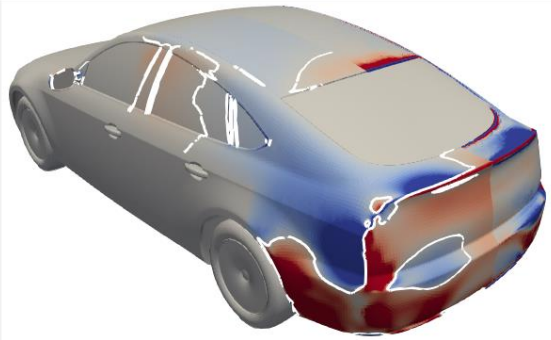
$r_{pL} = 0.5m$



$r_{pL} = 2m$



$r_{pL} = 10m$



Small wavelength features removed beyond a certain radius

Small wavelength features remain, irrespective of the radius

E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

Optimised Shapes

Laplace - Beltrami

- Smooth geometry
- J reduced by 5%

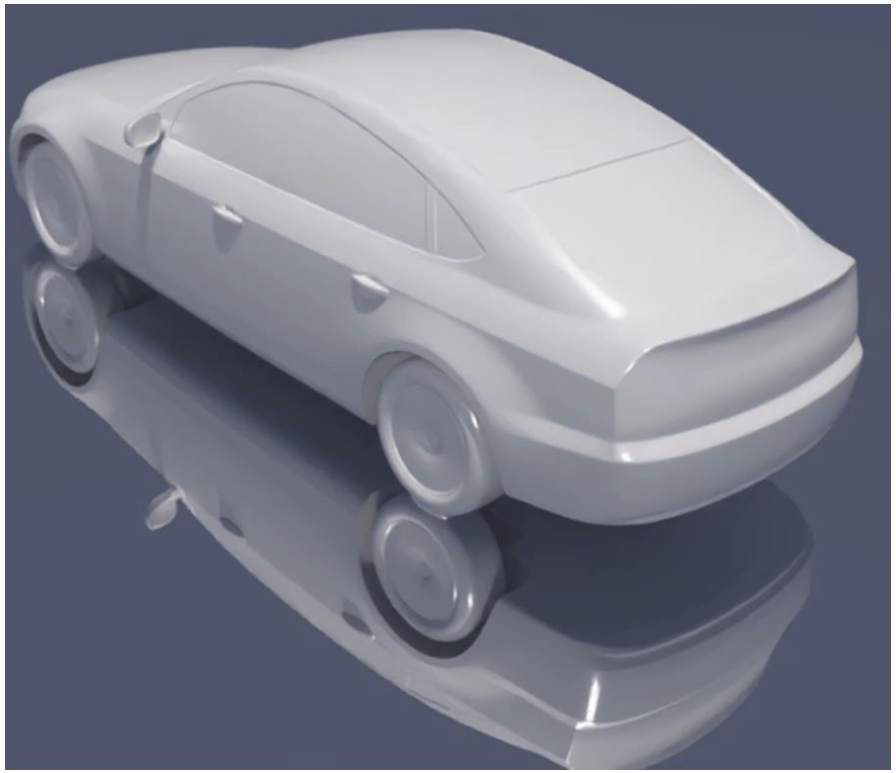
p-Laplacian

- Non-smooth geometry
- J reduced by 15%!

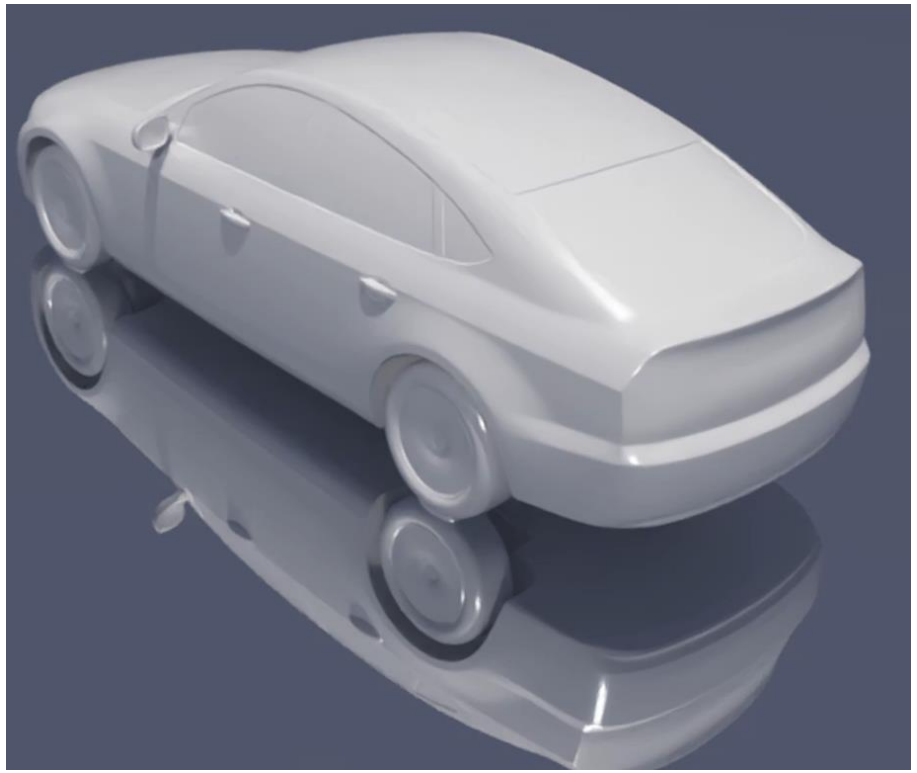
E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

Geometry Evolution

Laplace - Beltrami



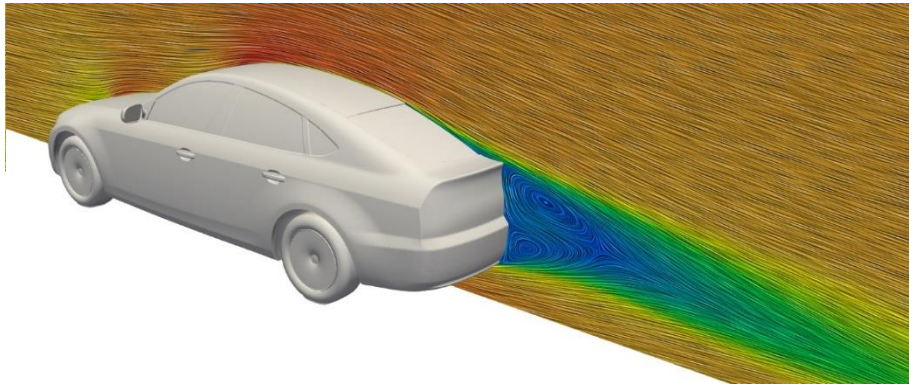
p-Laplacian



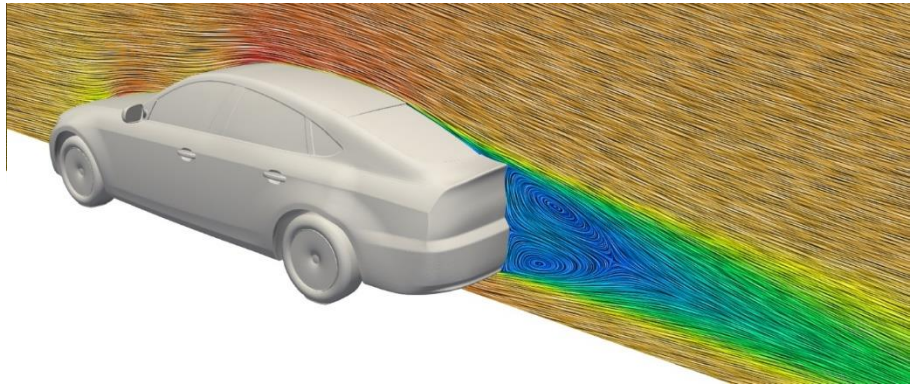
E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

ShpO of the Drivaer Car Model with Nodal Parametrisation

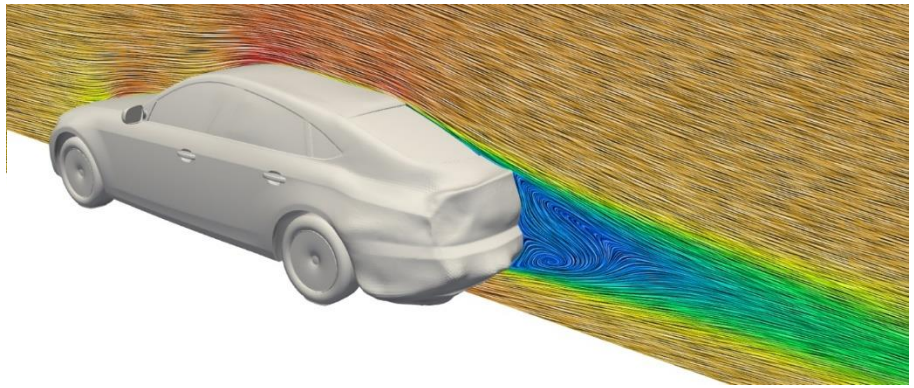
Baseline



Laplace - Beltrami



p-Laplacian



E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*

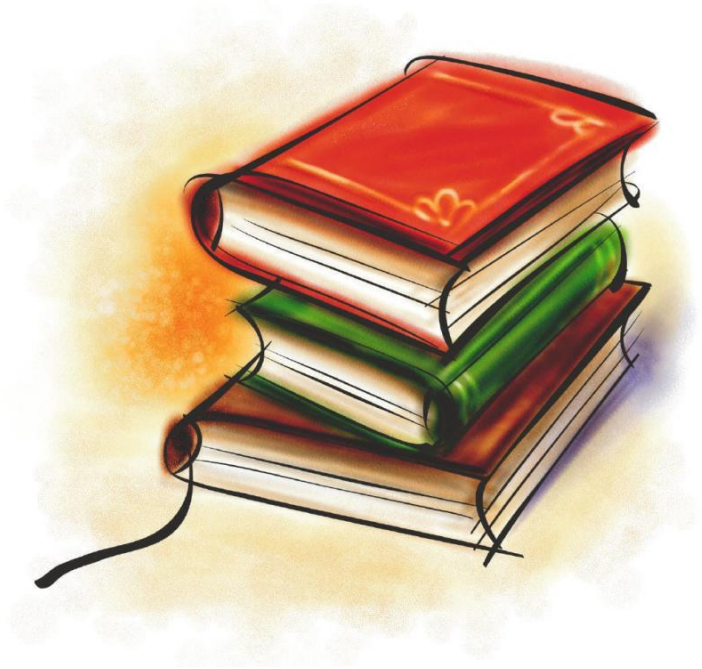
Conclusions

- **Regularisation is needed in ShpO using nodal parametrisation. The adjoint to the regularisation PDE must be solved, too.**

- **Laplace-Beltrami and p-Laplacian regularisation behave differently:**
  - **Laplace-Beltrami seems to be able to remove/smooth-out high frequency wavelengths from the sensitivity derivatives and the shape update.**
  - **There seems to be a limit on the ability of p-Laplacian to do the same (at least with p=2).**

- **Application to the Drivaer car model**
  - **p-Laplacian: high frequency wavelengths help to reduce drag considerably (-15%).**
  - **Laplace-Beltrami: smooth optimised shape, with only 5% drag reduction.**

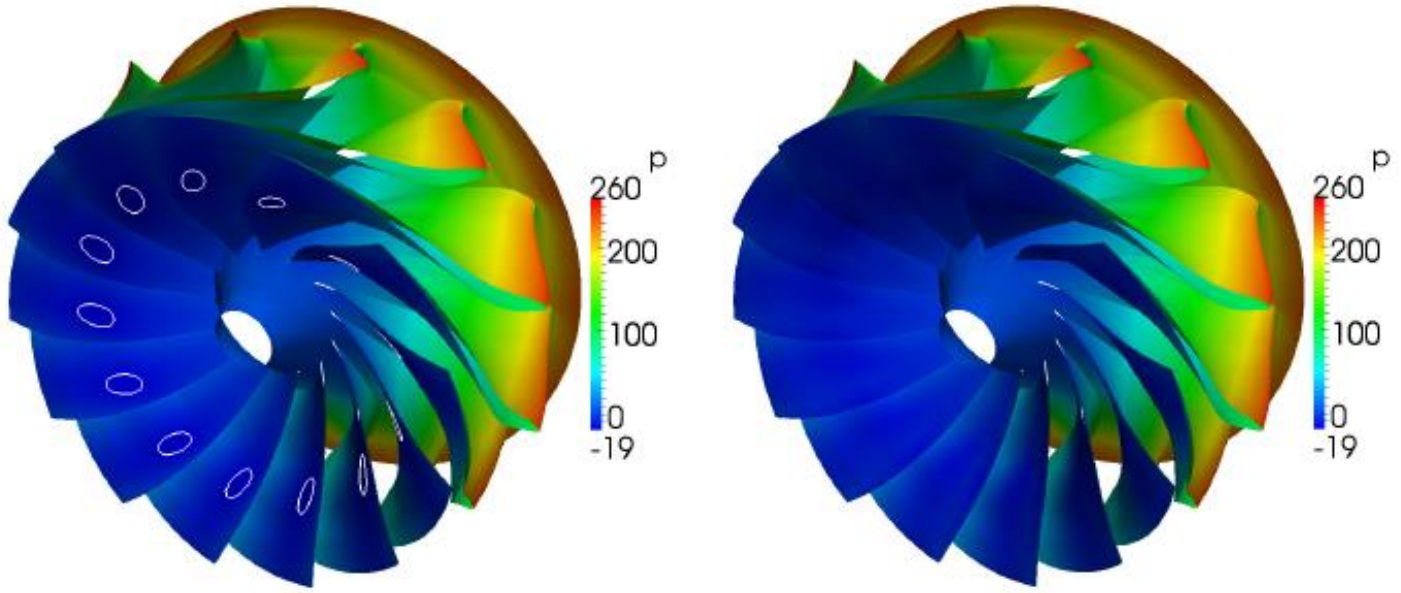
E.*M. Papoutsis-Kiachagias, K.C. Giannakoglou, An Adjoint-based, Parameterization-free Framework for Aerodynamic Shape Optimization in OpenFOAM, EUROGEN 2023, Chania, Greece, June 1-3, 2023.*



Other Applications

Hydraulic Turbine Applications - ShpO of a Francis Runner

Initial **Optimised**

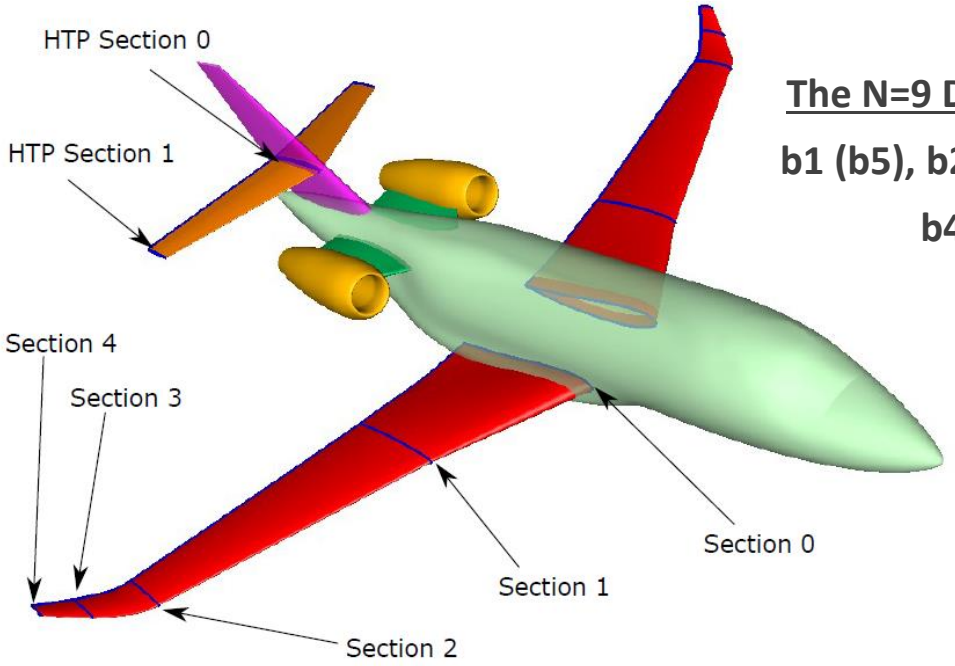


Redesigning the blades for min. cavitation. To suppress cavitation, the lowest pressure on the runner surface should become greater than the vapor pressure of the fluid. Adjoint is challenging since "min." is non-differentiable and needs to be replaced by a (differentiable) sigmoid function.

Computer Methods in Applied Mechanics and Engineering, 278:621-639, 2014.

Application funded by ANDRITZ Hydro

Compressible/Transonic Flow Cases - ShpO of a Business Jet

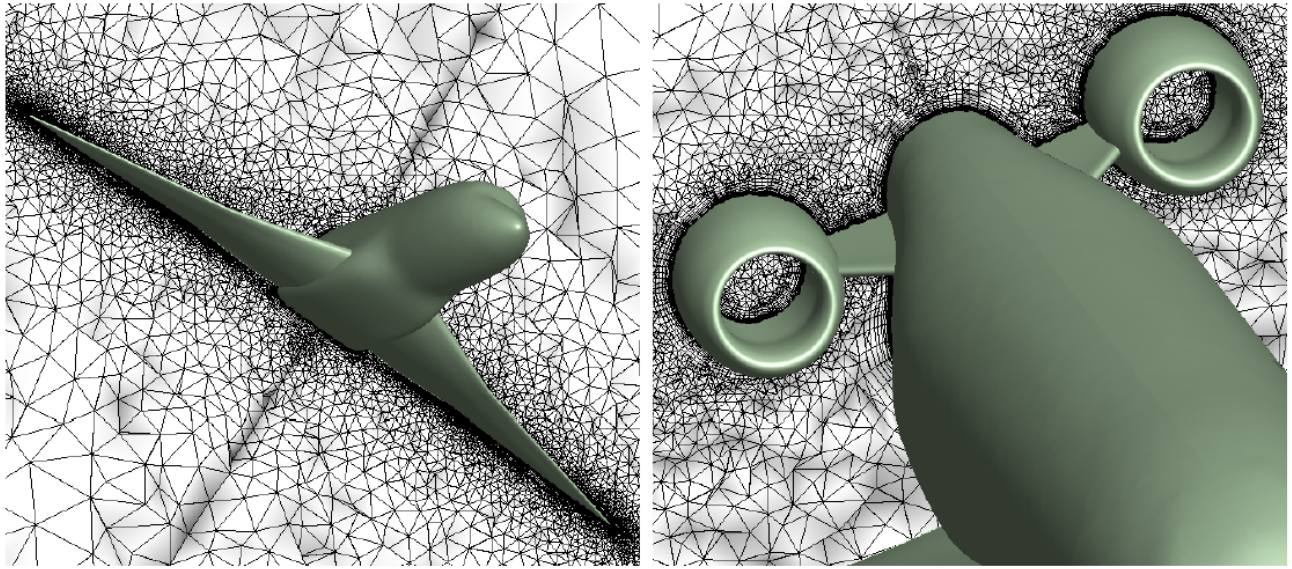


The N=9 DoFs: Wing/HTP sections directly controlled by the design variables. In specific:

b1 (b5), b2 (b6) and b3 (b7) control the twist (trailing edge camber) of sections 0, 1 and 2.

b4 (b8) controls the twist (trailing edge camber) of both sections 3 and 4.

b9 controls the rotation of both (0 and 1) HTP sections.



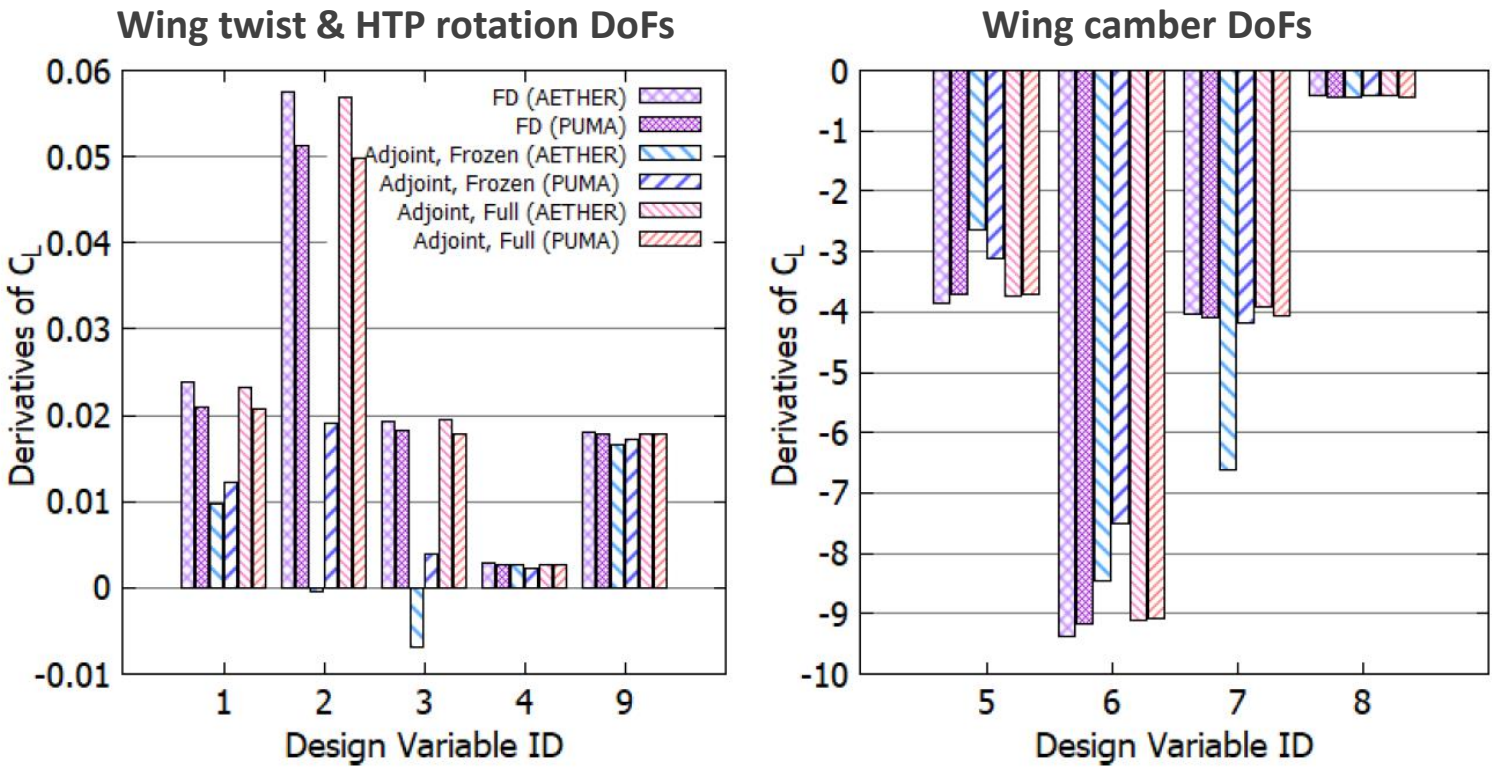
Studies performed in collaboration with (& with data provided by)

DASSAULT AVIATION

Research funded by

MADELEINE

On the Accuracy of the Computed Sensitivity Derivatives (SDs)



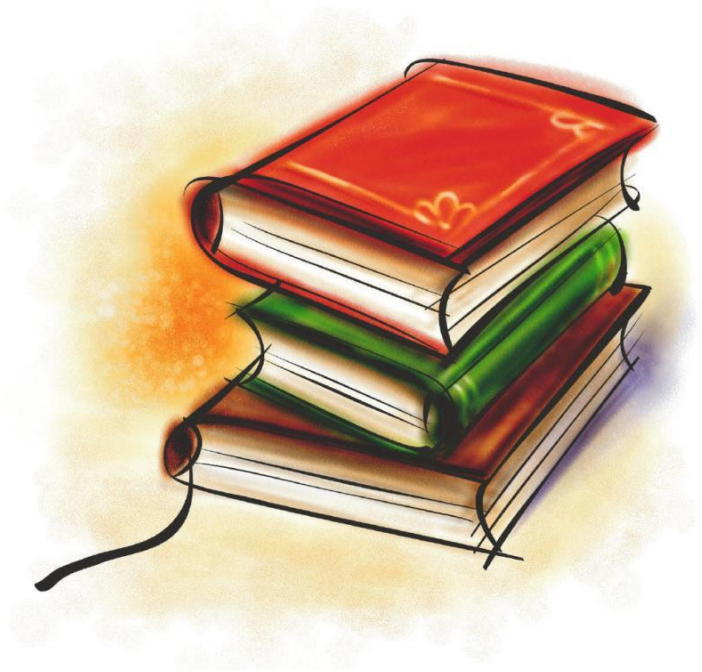
Studies performed in collaboration with (& with data provided by)

DASSAULT AVIATION

C_L sensitivities. The SDs computed by the adjoint of PUMA (continuous, finite volumes; by NTUA) and AETHER (discrete, finite elements; by DASSAULT AVIATION) are compared with FDs, at M=0.82, AoA=2.5deg.

Research funded by MADELEINE

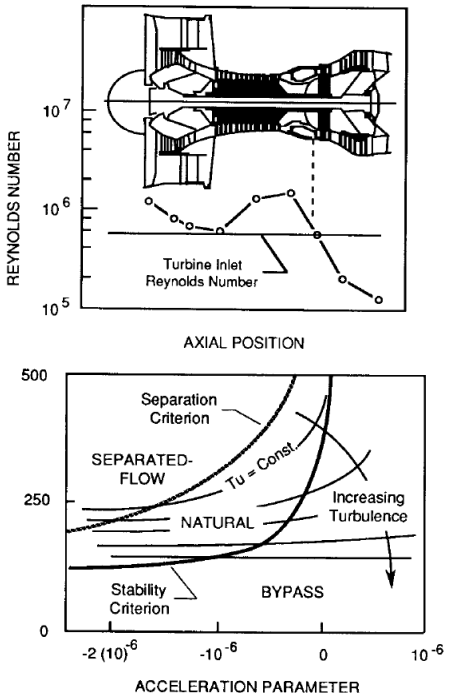
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr



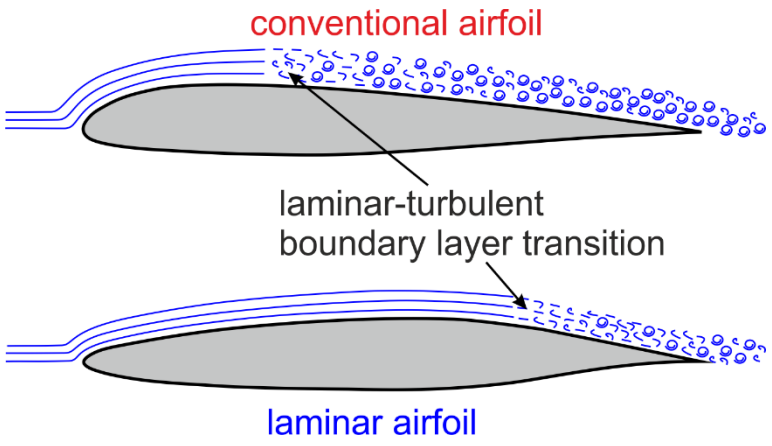
(Continuous) Adjoint for Transitional Flows

Why Transition Modeling?

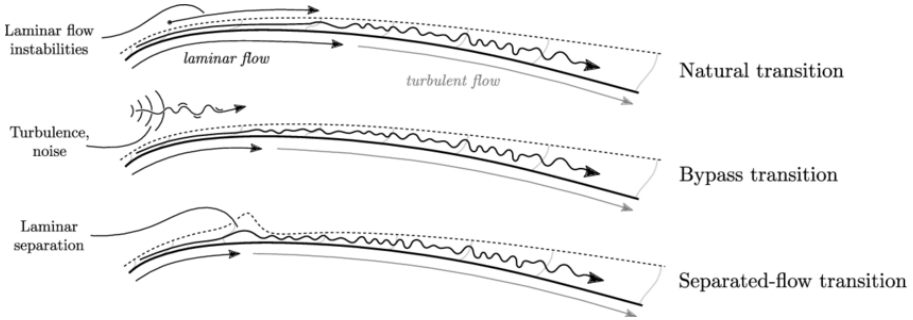
- **Effect of laminar to turbulent transition not included in the majority of CFD simulations.**
- **Boundary layer development through transition for computing losses and heat transfer on various components of the engine.**
- **Design of Natural Laminar Flow (NLF) wings for reduced energy consumption and emissions.**

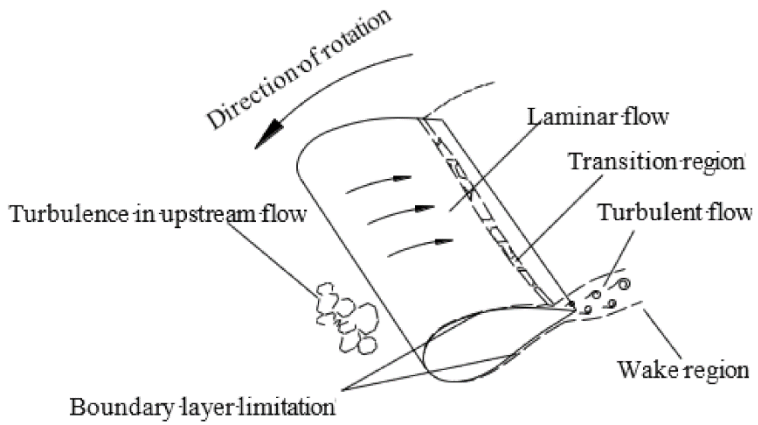


from Mayne (ASME 91-GT-261)



from DLR





from Jinlei 2021

The Primal/Flow Problem (1/2)

- **RANS Equations for Compressible Fluid Flows with the $\gamma - \tilde{R}e_{\theta t}$ Transition Model & a Distance computing eq.**

$$R_n^{\mathrm{MF}} = \frac{\partial U_n}{\partial t} + \frac{\partial f_{nk}^{\mathrm{inv}}}{\partial x_k} - \frac{\partial f_{nk}^{\mathrm{vis}}}{\partial x_k} + S_n = 0$$

Spalart-Allmaras

$$R^{\tilde{\nu}} = \frac{\partial(\rho\tilde{\nu})}{\partial t} + \frac{\partial(\rho\tilde{\nu}v_k^A)}{\partial x_k} - \frac{\rho}{\mathrm{Re}_0\,\sigma}\left\{\frac{\partial}{\partial x_k}\left[(\nu+\tilde{\nu})\frac{\partial\tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\right\} - P_{\tilde{\nu}} + D_{\tilde{\nu}} = 0$$

OR

k-ω Menter SST

$$R^{k} = \frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k v_k^A)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[(\mu+\sigma_k\mu_t)\frac{\partial k}{\partial x_k}\right] - \tilde{P}_{\kappa} + D_k = 0$$

$$R^{\omega} = \frac{\partial(\rho\omega)}{\partial t} + \frac{\partial(\rho\omega v_k^A)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[(\mu+\sigma_\omega\mu_t)\frac{\partial\omega}{\partial x_k}\right] - P_\omega + D_\omega = 0$$

4 variants of the $\gamma - \tilde{R}e_{\theta t}$ Transition Model:

- **SST-2003-LM2015**
- **SA-noft2-Gamma-Retheta**
- **SA-LM2015**
- **SA-sLM2015**

$$R^{\gamma} = \frac{\partial(\rho\gamma)}{\partial t} + \frac{\partial(\rho v_k^A\gamma)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\right] - P_\gamma + D_\gamma = 0$$

$$R^{\tilde{R}e_{\theta t}} = \frac{\partial\left(\rho\tilde{R}e_{\theta t}\right)}{\partial t} + \frac{\partial\left(\rho v_k^A\tilde{R}e_{\theta t}\right)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}(\mu+\mu_t)\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\right] - P_{\theta,t} - D_{SCF} = 0$$

Eikonal

$$R^{\Delta} = \frac{\partial}{\partial x_k}\left(\Delta\frac{\partial\Delta}{\partial x_k}\right) - \Delta\frac{\partial}{\partial x_k}\left(\frac{\partial\Delta}{\partial x_k}\right) - 1 = 0$$

where:
SST → k-ω Menter SST
SA → Spalart-Allmaras
LM → Langtry-Menter

γ Intermittency
$\tilde{R}e_{\theta t}$ Momentum-Thickness Reynolds Number

The Primal/Flow Problem (2/2)

- Why so many variants of the $\gamma - \widetilde{Re}_{\theta t}$ model?
- F_{onset} controls the transition onset location.
- Towards **differentiable** expressions, in view of the development of the adjoint method.

SST-2003-LM2015

$$F_{onset} = \max\left(F_{onset2} - F_{onset3}, 0\right), \quad F_{onset2} = \min\left[\max\left(F_{onset1}, F_{onset1}^4\right), 2\right]$$

$$F_{onset3} = \max\left[\left(1 - \left(\frac{R_T}{2.5}\right)^3\right), 0\right], \quad F_{onset1} = \frac{Re_v}{2.193 Re_{\theta c}}$$

4 variants of the $\gamma - \widetilde{Re}_{\theta t}$ Transition Model:

- SST-2003-LM2015
- SA-noft2-Gamma-Retheta
- SA-LM2015
- SA-sLM2015

(s=smooth)

SA-sLM2015

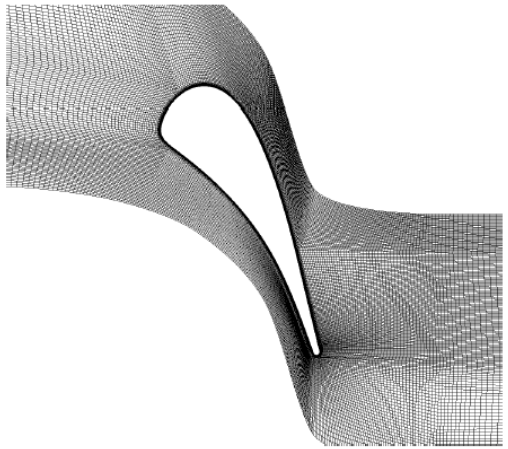
$$F_{onset} = \frac{\tanh\left[6\left(F_{onset1} - 1.35\right)\right] + 1}{2}, \quad F_{onset1} = \sqrt{\left(\frac{Re_S}{2.6 Re_{\theta c}}\right)^2 + \left(R_T\right)^2}$$

Verification / Validation of Transition Models (1/3)

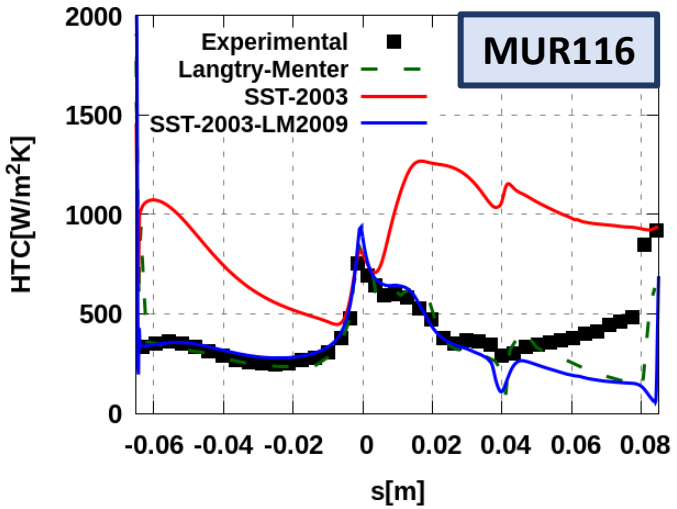
- **LS89 Transonic Turbine Blade-Airfoil with Heat Transfer.**
  - **Structured Mesh of ~50K nodes.**
  - **Transition Model: SST-2003-LM2015.**

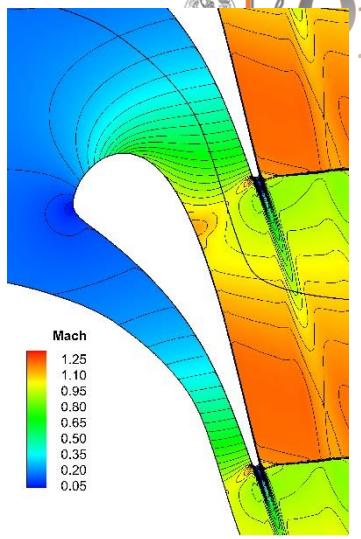
PUMA *by PCOpt/NTUA*

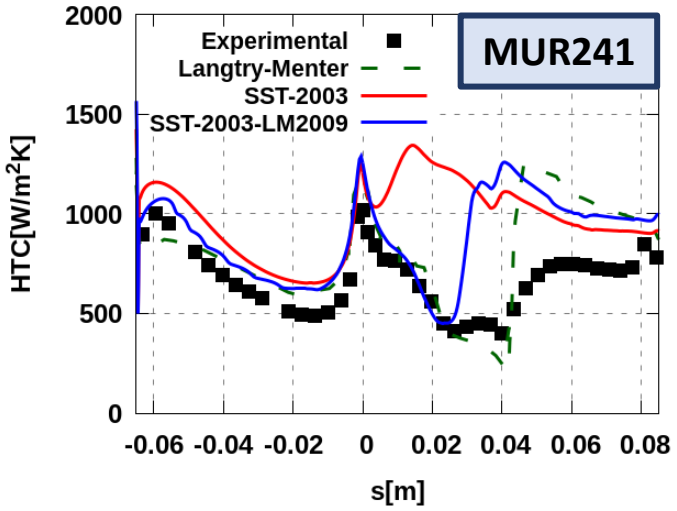
	MUR116	MUR241
P_t^{in}	3.269	3.257
T_t^{in}	418.90	416.40
P^{out}	1.550	1.574
T^w	297.55	299.75
α_1	0	0
Tu(%)	0.8	6
$(\mu_t/\mu)^{in}$	11	1000

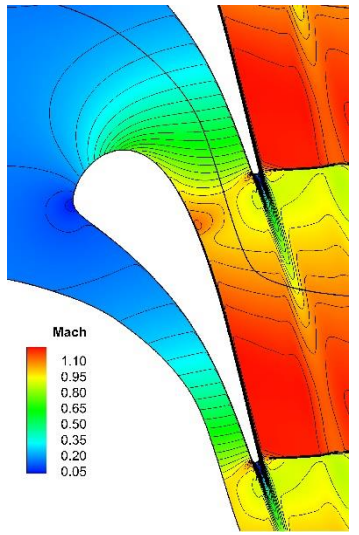


$$HTC = \frac{q^w}{T_{01} - T^w}, \quad \text{where} \quad T_{01} = 420.00\,\text{K}$$





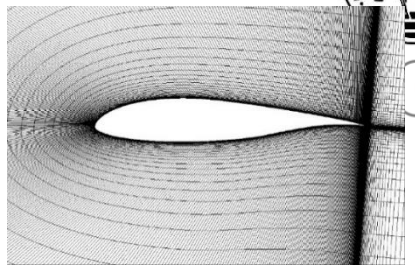




Verification / Validation of Transition Models (2/3)

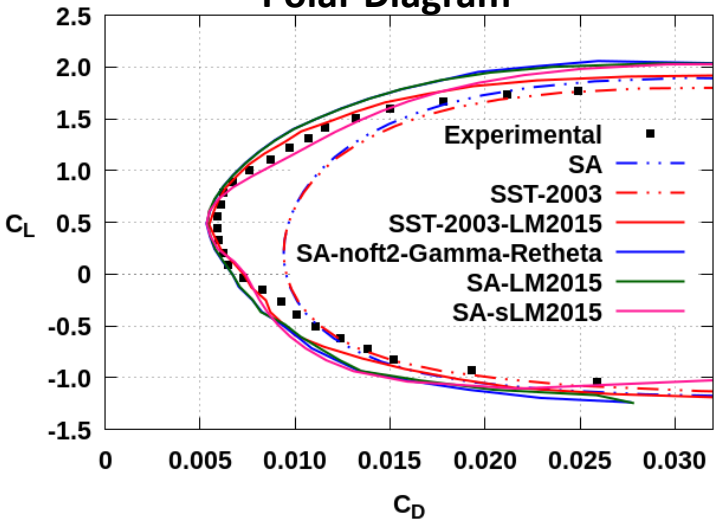
- **NLF(1)-0416 Isolated Airfoil (Natural Laminar Flow)**
  - **Re=4·10^6, M$_\infty$=0.1, AoA =[-17°,17°], Tu=0.15%. Mesh of ~150K nodes.**
  - **All Transition Models.**

PUMA by PCOpt/NTUA

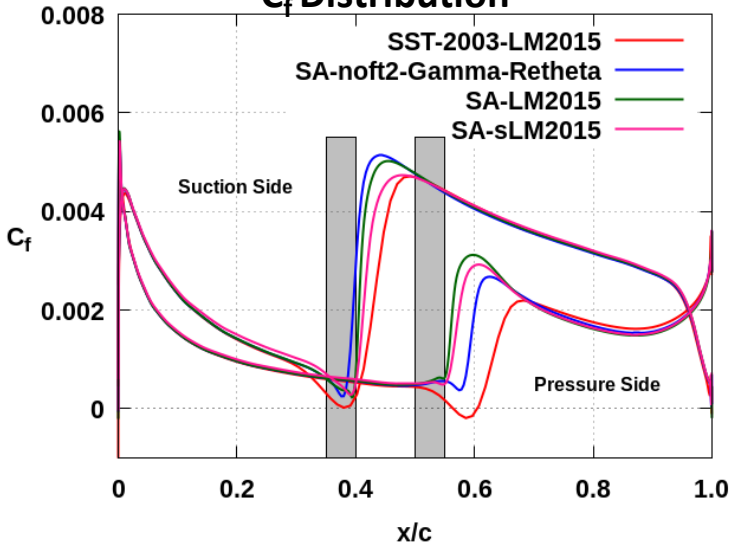


AoA=0°

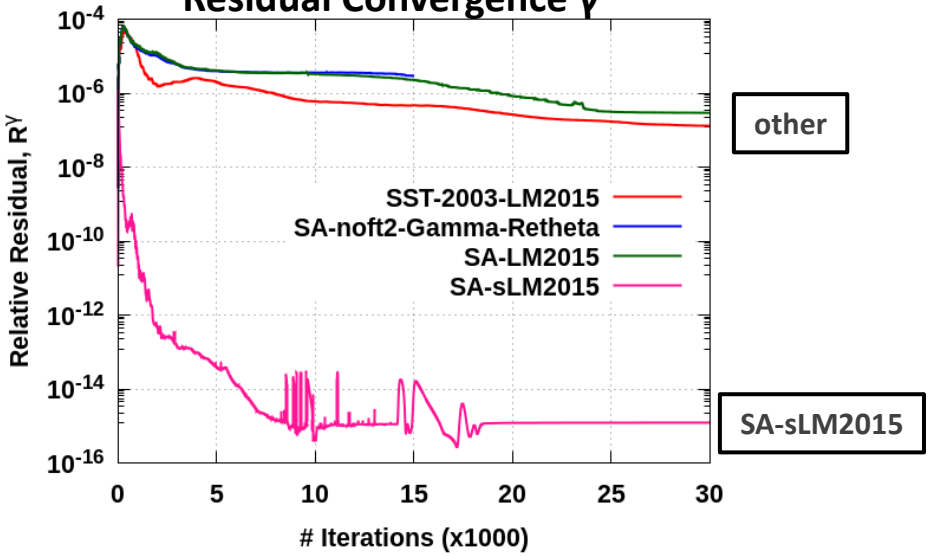
Polar Diagram



C$_f$ Distribution



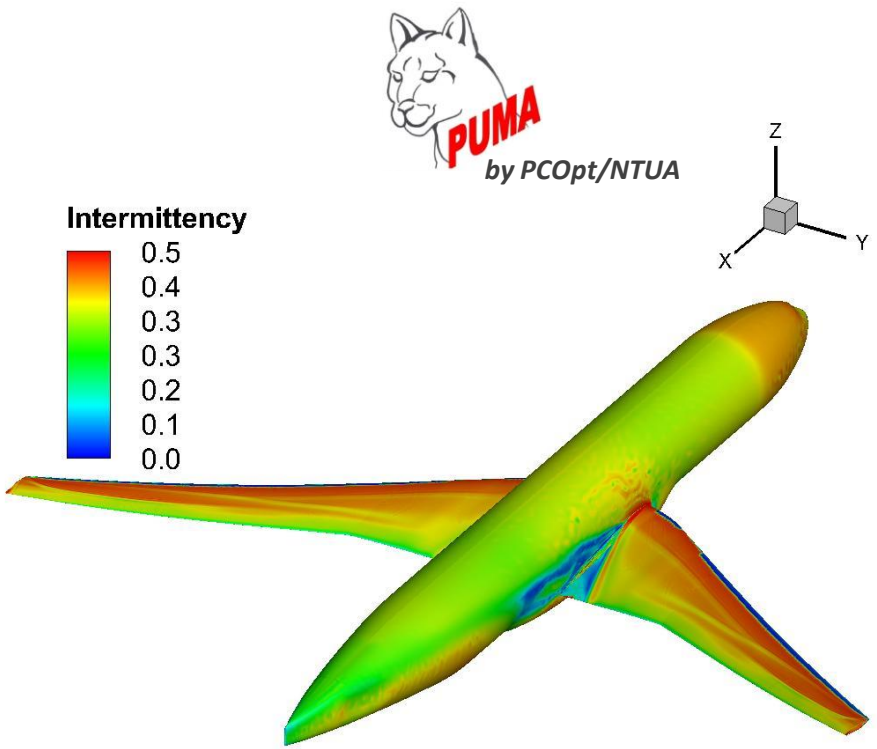
Residual Convergence γ



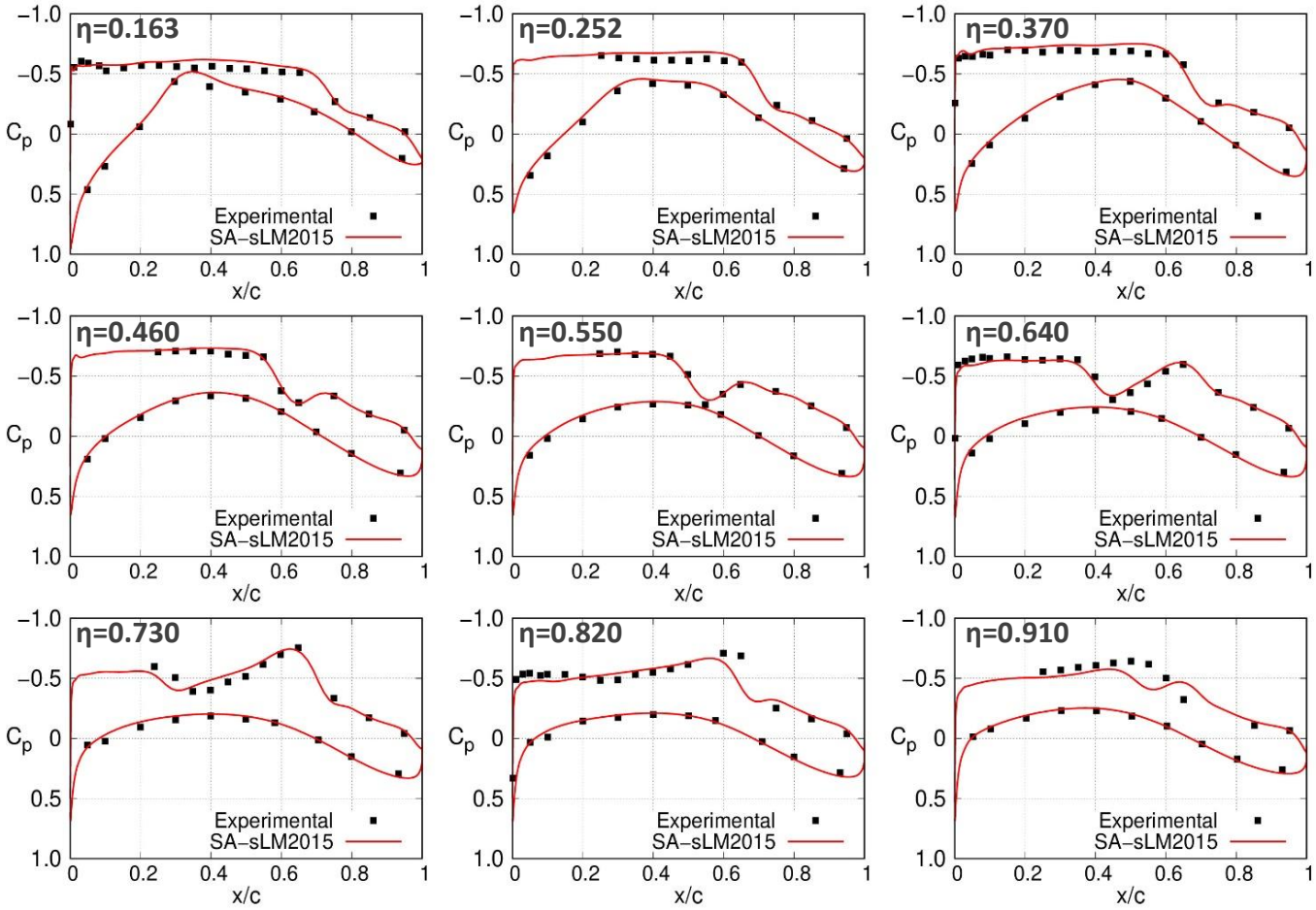
✍ **The smooth/differentiable variant SA-sLM2015 convergences by many orders of magnitude which iσ beneficial within an adjoint-based ShpO loop!**

Verification / Validation of Transition Models (3/3)

- **NASA Common Research Model (CRM)-NLF Transonic Wing**
  - **Re=15·10^6, M∞=0.86, AoA =[1.5o,2°,2.5°,3°], T_u=0.24%, h_{rms}=0.0254μm. Mesh of ~14M nodes.**
  - **Transition Model: SA-sLM2015.**



C_P Distribution



Continuous Adjoint Solver (1/2)

- **Spalart-Allmaras Turbulence Model coupled with the $\gamma - \tilde{R}e_{\theta t}$ Transition Model – 3 variants.**

$$J_{\text{aug}} = J + \int_{\Omega} \Psi_n R_n^{\text{MF}} \mathrm{d}\Omega + \int_{\Omega} \tilde{\nu}_a R^{\tilde{\nu}} \mathrm{d}\Omega + \int_{\Omega} \gamma_a R^{\gamma} \mathrm{d}\Omega + \int_{\Omega} \tilde{Re}_a R^{\tilde{Re}_{\theta t}} \mathrm{d}\Omega + \int_{\Omega} \Delta_a R^{\Delta} \mathrm{d}\Omega$$

Differentiation w.r.t. the design variables bi

$$\frac{\delta J}{\delta b_i} = \int_{\Omega} \text{FAE}_n \frac{\delta U_n}{\delta b_i} \mathrm{d}\Omega + \int_{S} \text{ABC}_n \frac{\delta U_n}{\delta b_i} \mathrm{d}S + \int_{\Omega} \text{SD}_{m\ell}^{\Omega} \frac{\partial}{\partial x_m} \left(\frac{\delta x_\ell}{\delta b_i} \right) \mathrm{d}\Omega + \int_{S} \text{SD}_{\ell}^{S} \frac{\delta x_\ell}{\delta b_i} \mathrm{d}S$$

- **Field Adjoint Equations**
- **Adjoint Boundary Conditions**
- **Sensitivity Derivative**
- **Mesh Morphing**
- **Sensitivity Derivative**
- **Parametrisation tool**

Continuous Adjoint Solver (2/2)

- **Field Adjoint Equations (Spalart-Allmaras + $\gamma - \tilde{Re}_{\theta t}$)**

$$R_m^{\Psi} = -A_{nmk}\frac{\partial \Psi_n}{\partial x_k} - \mathcal{K}_m + K_m^{\mathrm{SA}} + \boxed{\mathcal{K}_m^{\gamma-\tilde{Re}_{\theta t}}} + \mathcal{B}_m + \mathcal{B}_m^{\mathrm{SA}} + \boxed{\mathcal{B}_m^{\gamma-\tilde{Re}_{\theta t}}} = 0$$

Extra terms due to the (adjoint) transition

$$R^{\tilde{\nu}_a} = -v_k^A\frac{\partial \tilde{\nu}_a}{\partial x_k} - \mathcal{G}^{\mathrm{SA},diff} + \mathcal{G}^{\mathrm{SA},src} + \left[\mathcal{G}^{\mu_t} + \boxed{\mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}}}\right]\frac{\partial \mu_t}{\partial \tilde{\mu}} = 0$$

$$R^{\gamma_a} = -v_k^A\frac{\partial \gamma_a}{\partial x_k} - \mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},diff} + \mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},src} + \mathcal{H}_{\gamma}^{\mathrm{SA},src} = 0$$

$$R^{\tilde{Re}_a} = -v_k^A\frac{\partial \tilde{Re}_a}{\partial x_k} - \mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff} + \mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src} + \mathcal{H}_{\tilde{Re}_{\theta t}}^{\mathrm{SA},src} = 0$$

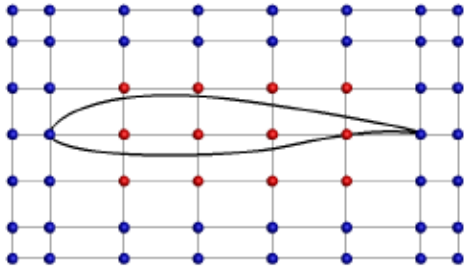
Adjoint Transition Eqs

$$R^{\Delta_a} = -2\frac{\partial}{\partial x_k}\left(\Delta_a\frac{\partial \Delta}{\partial x_k}\right) + \mathcal{M}^{\mathrm{SA},src} + \boxed{\mathcal{M}^{\gamma-\tilde{Re}_{\theta t},src}} = 0$$

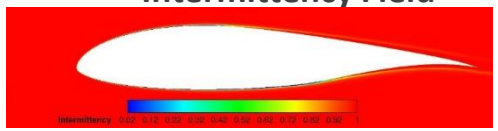
Could we skip the solution of the adjoint transition model PDEs?
“Frozen Transition” assumption

Continuous Adjoint Solver / The NLF(1)-0416 Isolated Airfoil (1/3)

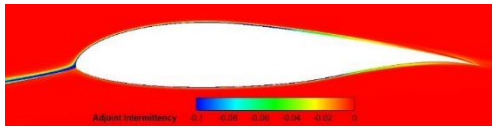
- **Parametrisation: 8x7 NURBS control lattice; 12 CPs→24 DVs.**
- **Quantities of Interest: δCD and δCL. Three transition models with and without the "Frozen Transition" Assumption.**



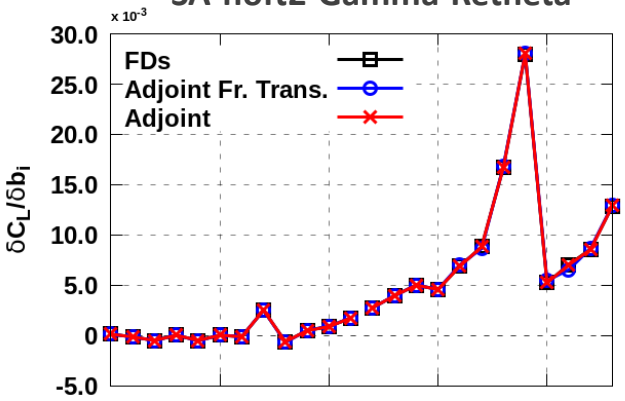
Intermittency Field

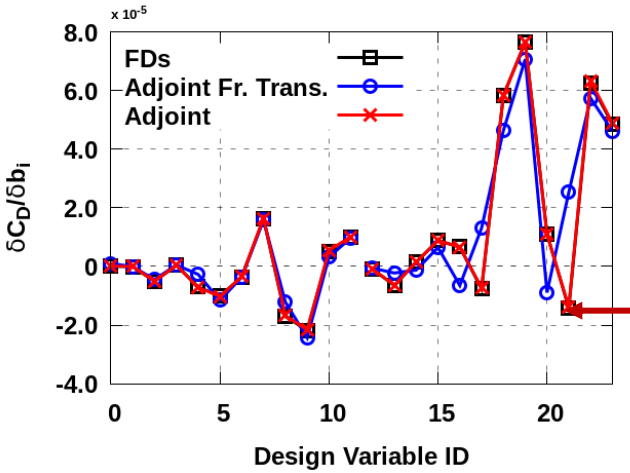


Adjoint Intermittency Field

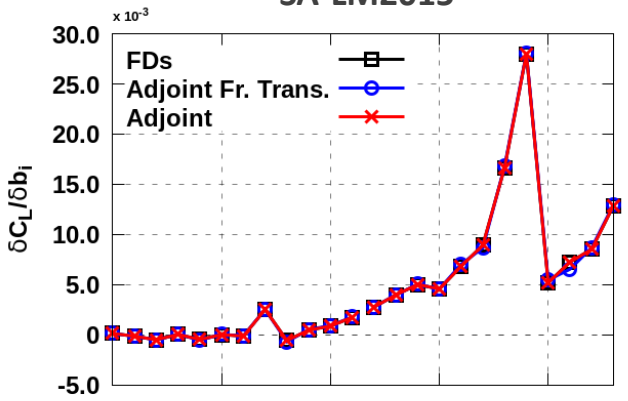


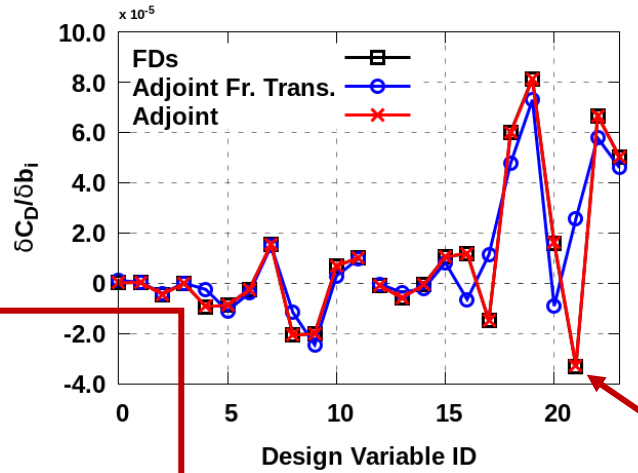
SA-noft2-Gamma-Retheta



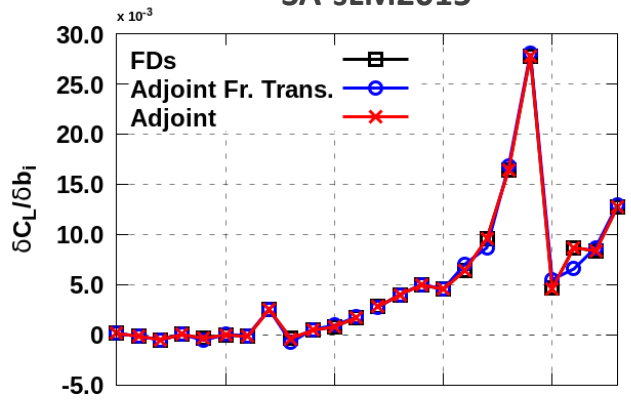


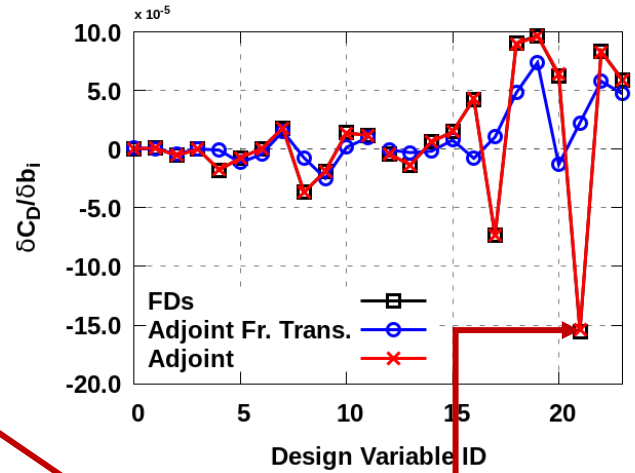
SA-LM2015





SA-sLM2015



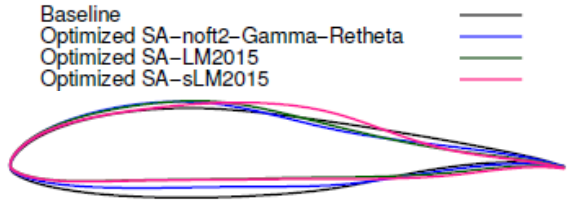


"Frozen Transition" Assumption

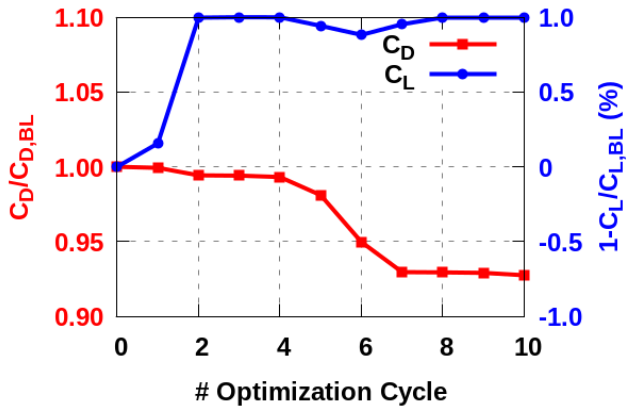
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vast

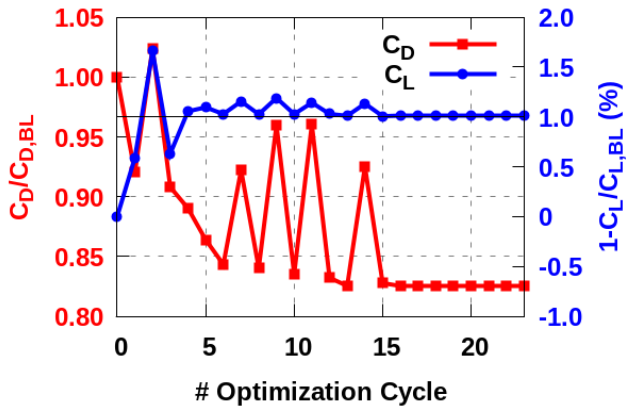
Continuous Adjoint Solver / ShpO of the NLF(1)-0416 Isolated Airfoil (2/3)

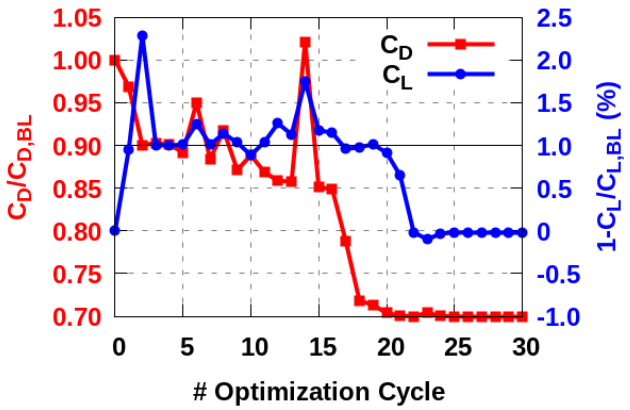
- **ShpO: min. C_D constrained by C_L (±1%) and area (≥90%) of the baseline values.**
- **Three transition models. Reduction ~8%, ~18% and ~30% in C_D, while meeting the constraints.**



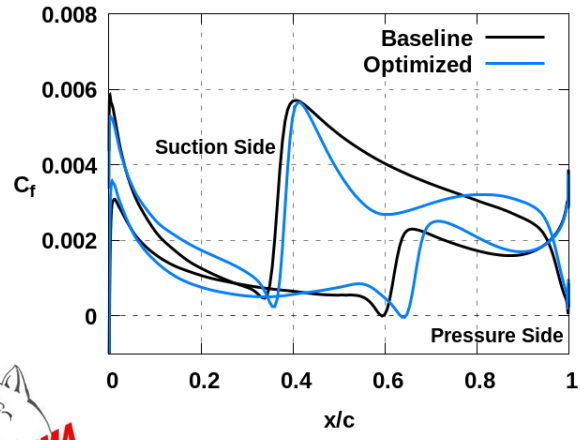
SA-noft2-Gamma-Retheta | **SA-LM2015** | **SA-sLM2015**

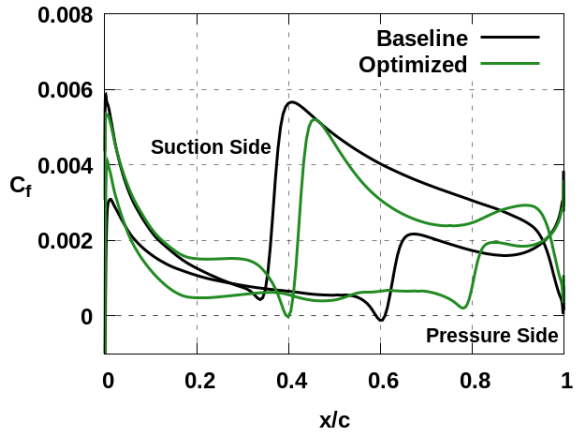


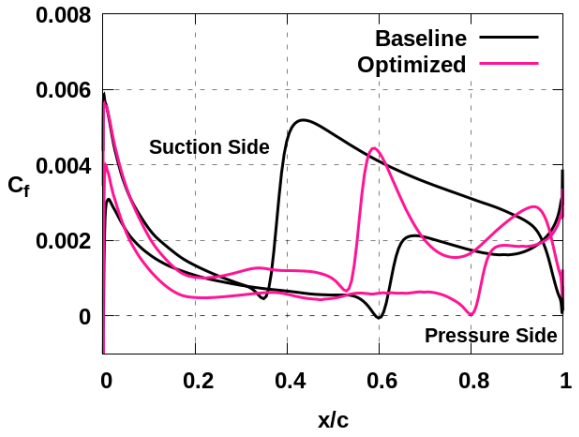




Optimisation Histories





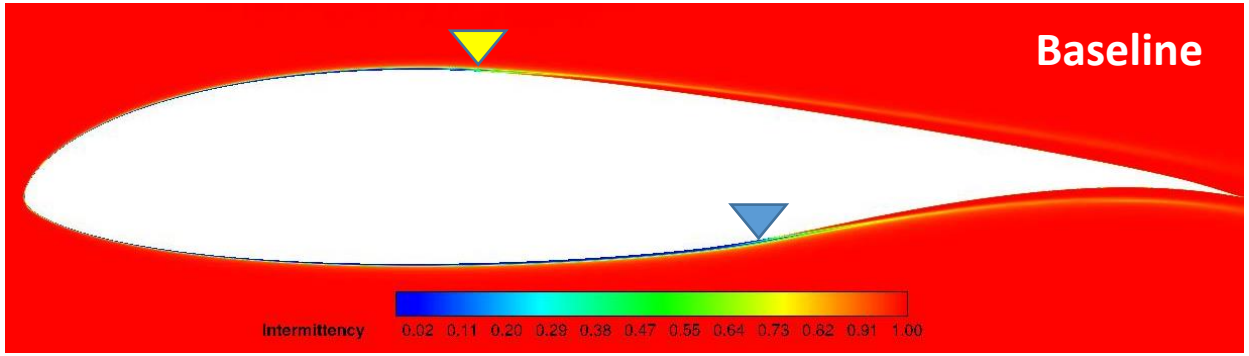


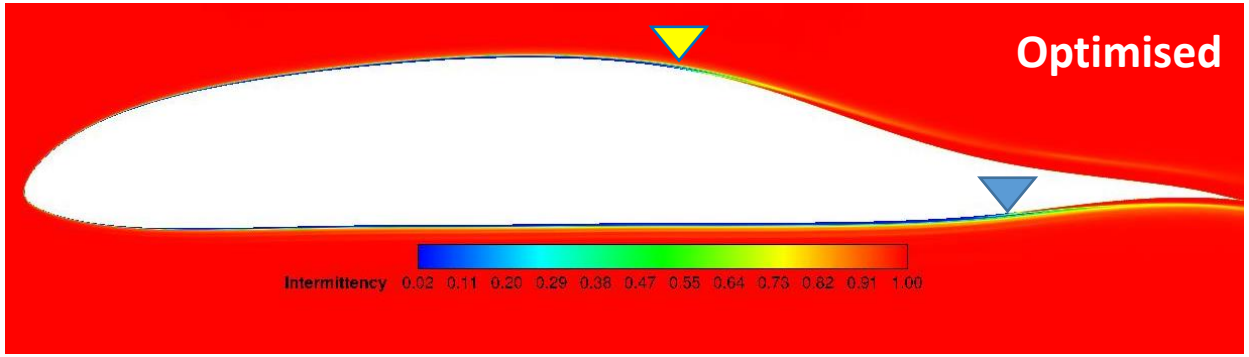
C_f Distributions

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

Continuous Adjoint Solver / ShpO of the NLF(1)-0416 Isolated Airfoil (3/3)

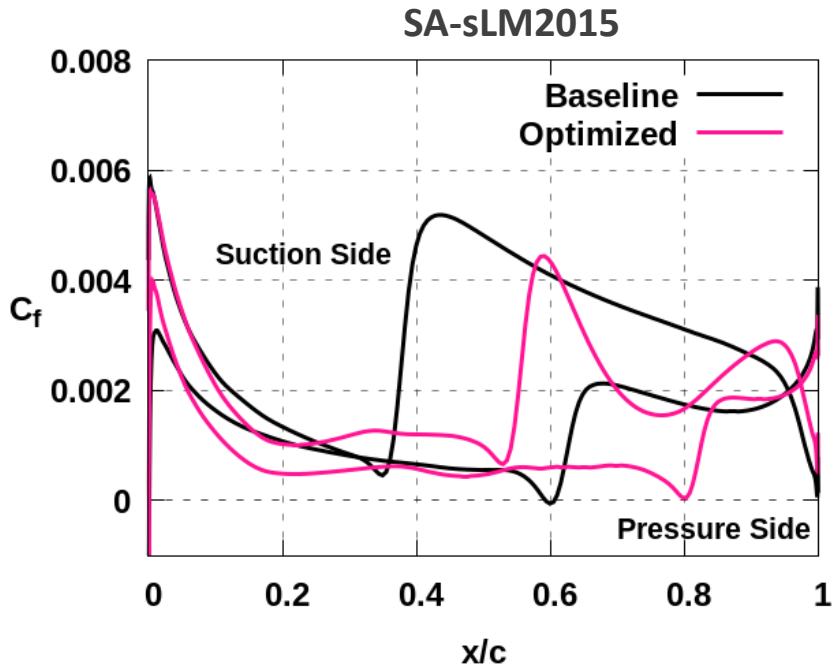
- **ShpO: min. C_D constrained by C_L (±1%) and area (≥90%) of the baseline values.**
- **Reduction ~30% on C_D meeting the C_L constraint.** All these have been computed using the TDDC adjoint!





Transition location at pressure ▼ and suction ▼ sides

	Baseline	Optimised
Transition Location on Pressure Side (%c)	~60%	~80%
Transition Location on Suction Side (%c)	~38%	~55%



High Aspect-Ratio Wing Aircrafts (1/8)

High-Aspect ratio Wings (HARWs) are very important for reaching NET ZERO 2050.

NET ZERO 2025 is the international goal to balance greenhouse gas emissions with emissions removed from the atmosphere, to prevent catastrophic climate change.

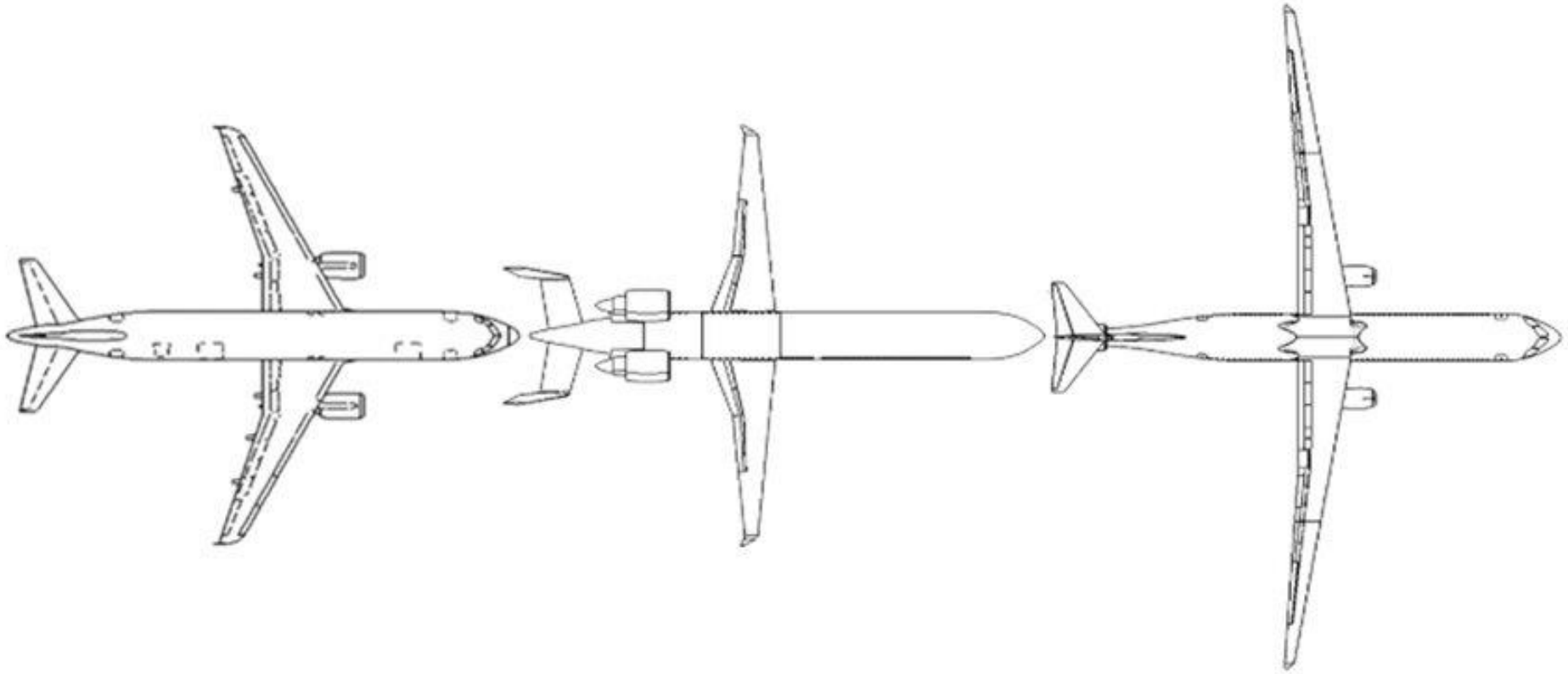
While aircraft wings generate lift, vortices generated at the wing tips increase drag that the propulsion systems should additionally overcome.

A HARW, i.e. a longer and likely slender wing may weaken the tip vortices, thus generating less drag and reducing fuel consumption, even by 10%. The reduction in fuel consumption has a positive impact on climate and, also, the cost of flying.

However, making the wings longer moves the loads they support outward from the fuselage, increasing the stresses at the wing roots, which should carefully be managed (adding trusses, etc).

Overall, the aeroelastic stability of HARWs constitutes a fundamental challenge in aerospace engineering.

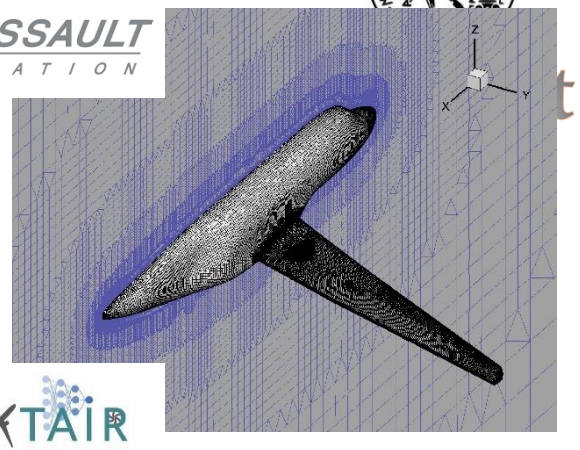
HARW Aircraft (2/8)



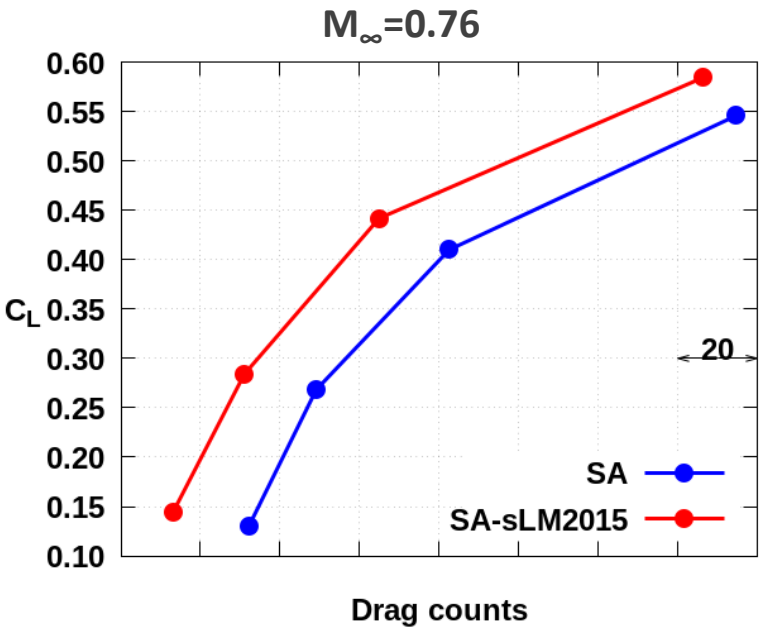
Potential Evolution toward HARWs from Airbus A320 baseline.

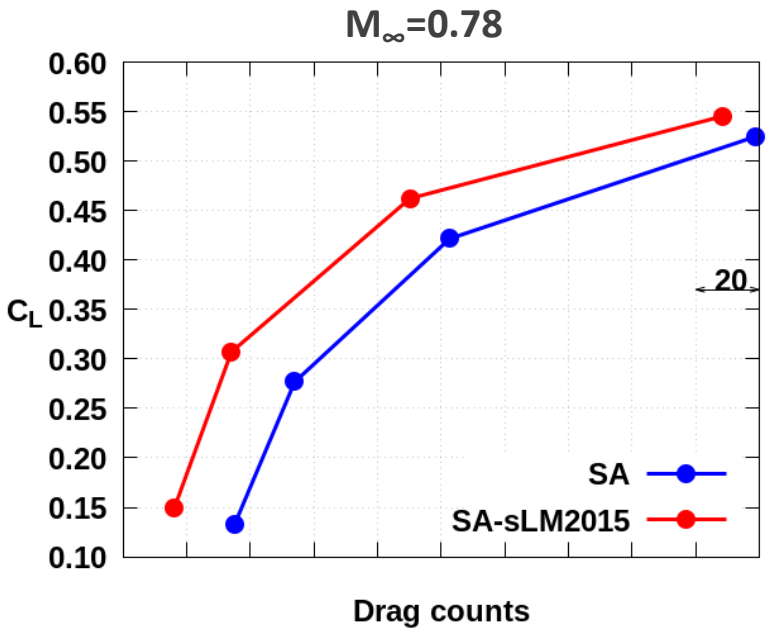
HARW Aircraft (3/8)

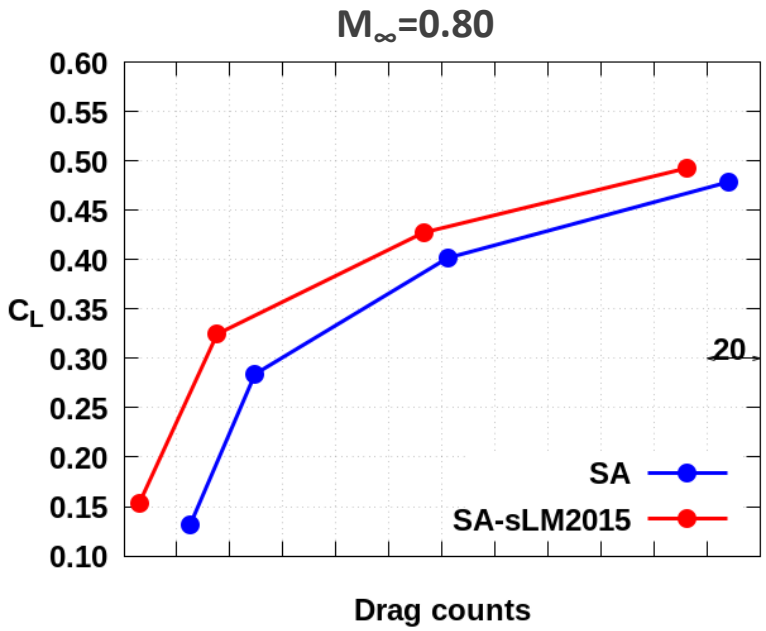
Analysis of a HARW Business Jet (4/8)



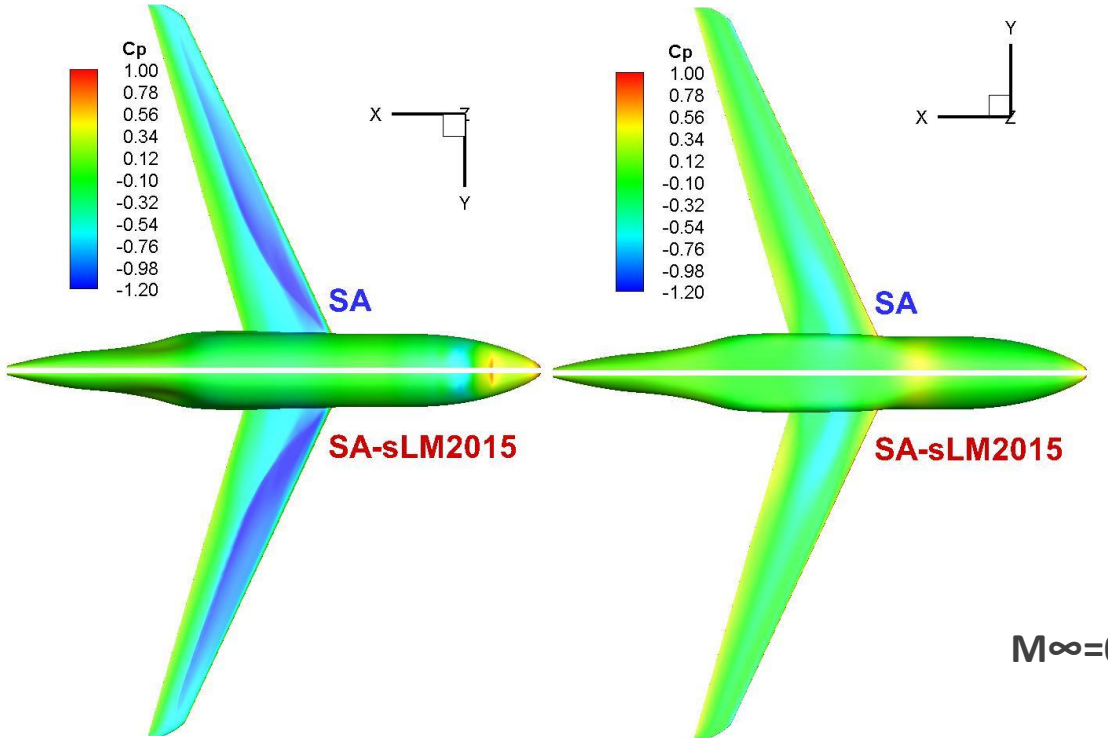
- Re=4.24·10^6, M_∞=[0.76,0.78,0.80], AoA =[0°,1°,2°,3°], Tu=0.15%. Mesh of ~12.4M nodes.
- Transition Model: SA-sLM2015, TDDC Adjoint used.
- Polar diagrams without/with transition model. $C_L=C_{L,WB}+\lambda C_{m,WB}$.
- Case and flow data provided by Dassault Aviation in the NEXTAIR project.

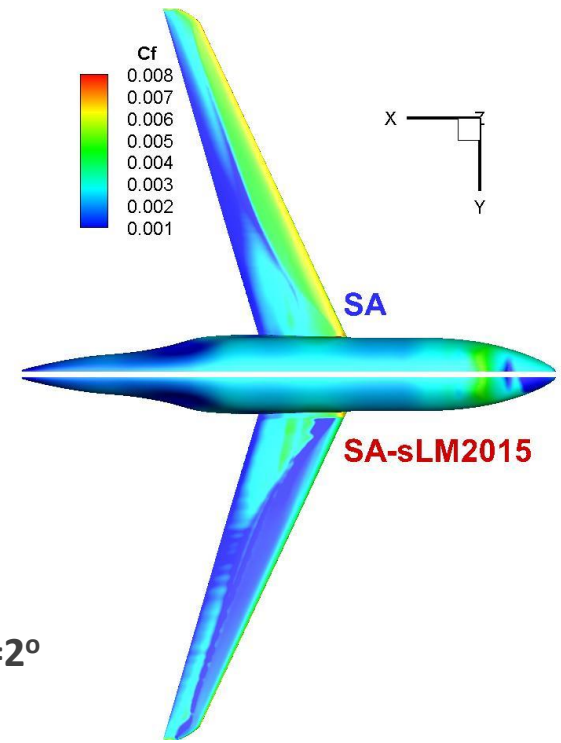


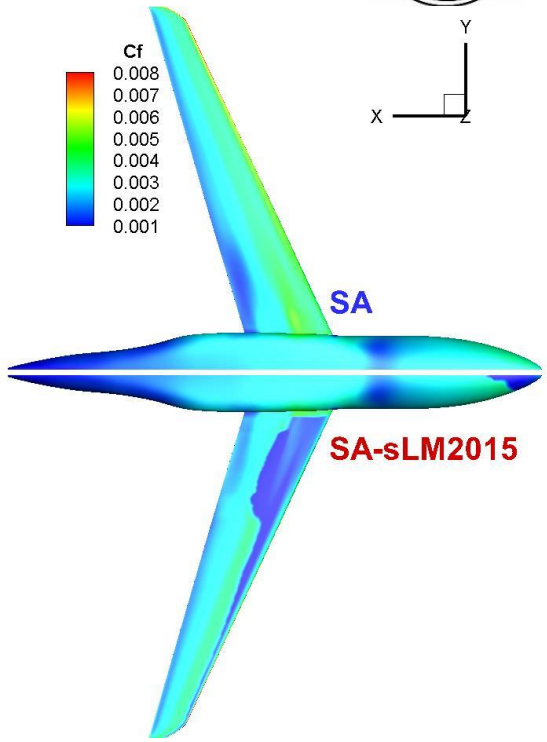




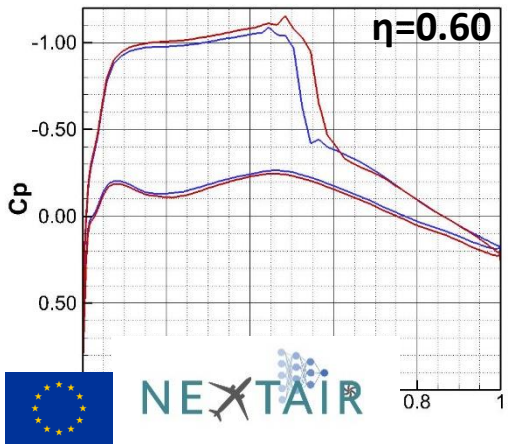
Analysis of a HARW Business Jet (5/8)

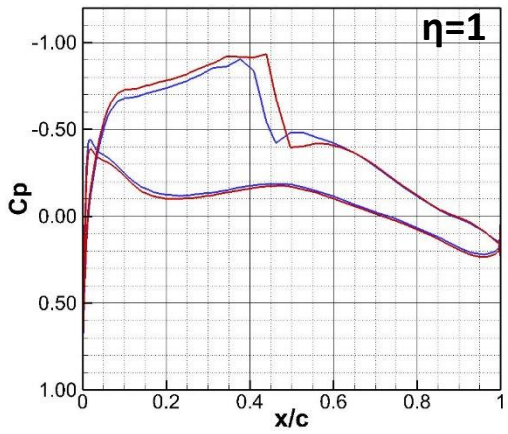




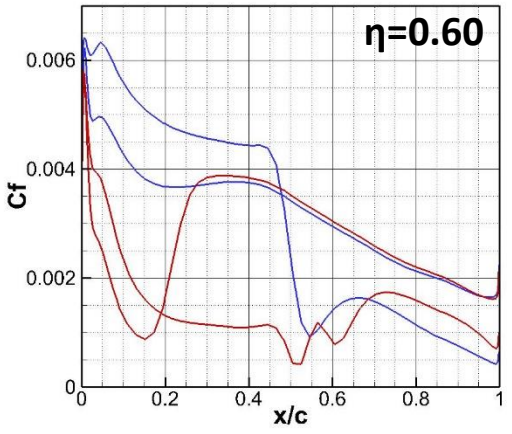


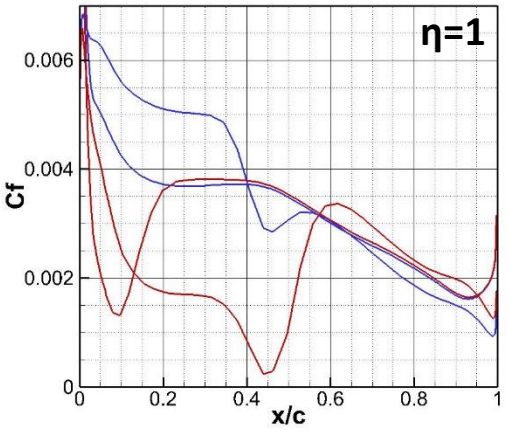
M∞=0.78, AoA=2°





PUMA by PCOpt/NTUA

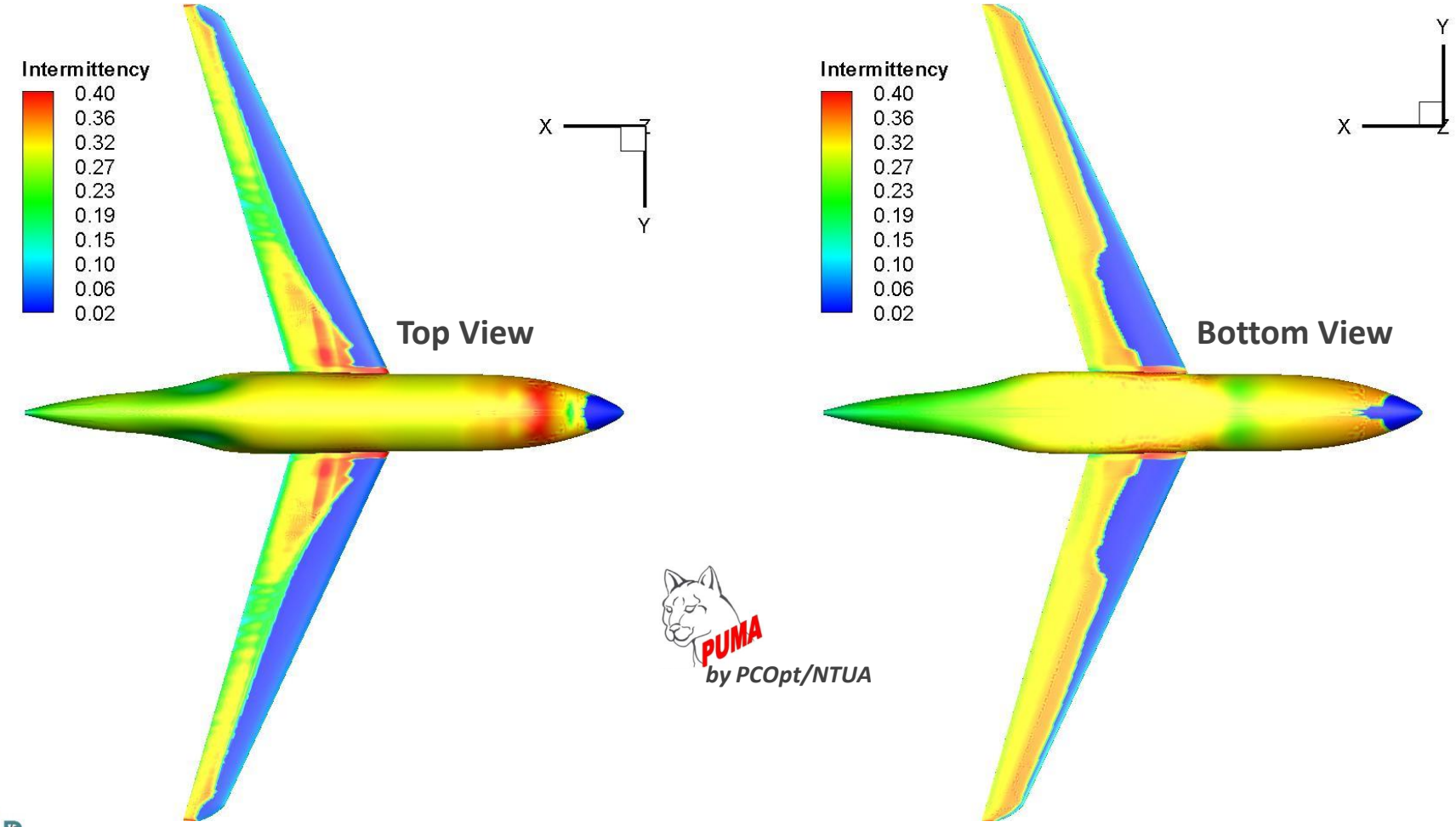




a@mail.ntua.gr, Dr.

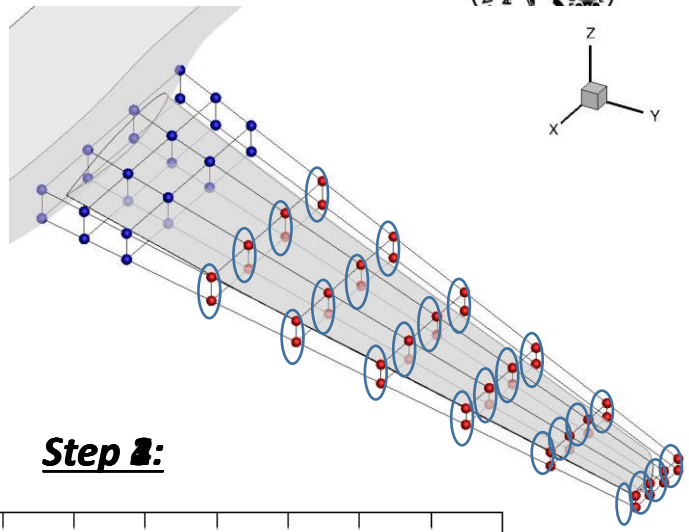
Analysis of a HARW Business Jet (6/8)

SA-sLM2015



Analysis of a HARW Business Jet (7/8)

- ShpO @M_∞ =0.78: min. C_D constrained by C_L=0.5±10^{-4}, $C_L=C_{L,WB}+\lambda C_{m,WB}$.
- Parametrisation: 4x9x2 volumetric NURBS control lattice; 48CPs are allowed to move in pairs in the z direction (maintain wing thickness).
- 24 DVs (parametrisation)+1 (AoA) → 25 DVs.



Four-Step Optimisation:

Spalart-Allmaras without Transition

Step 1: Compute (adjoint) the AoA (2.67°) for which C_L=0.5.

Step 2: Adjoint-based ShpO with fixed AoA.

Spalart-Allmaras with Transition

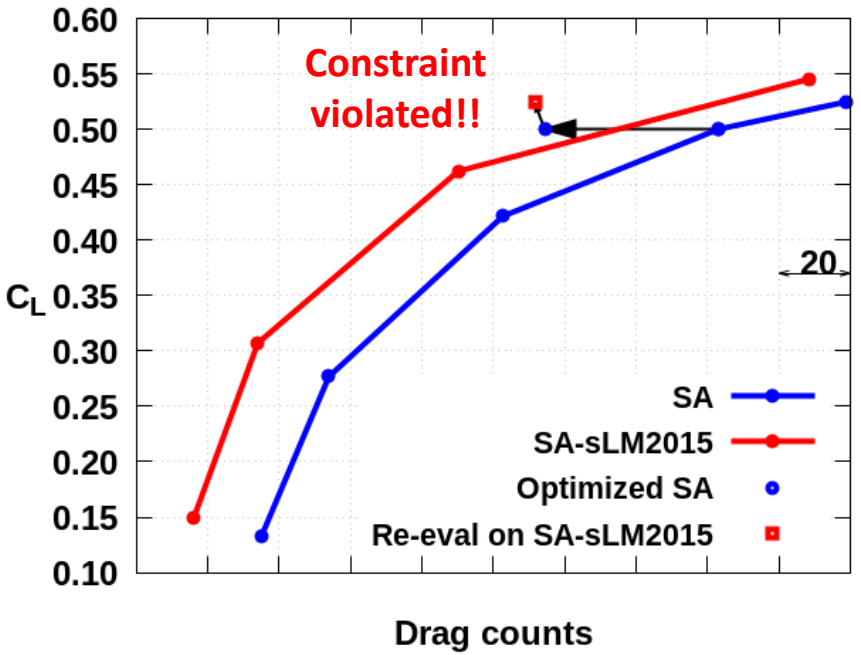
Step 3: Re-evaluate the optimised geometry.

Step 4: Adjoint-based optimisation.

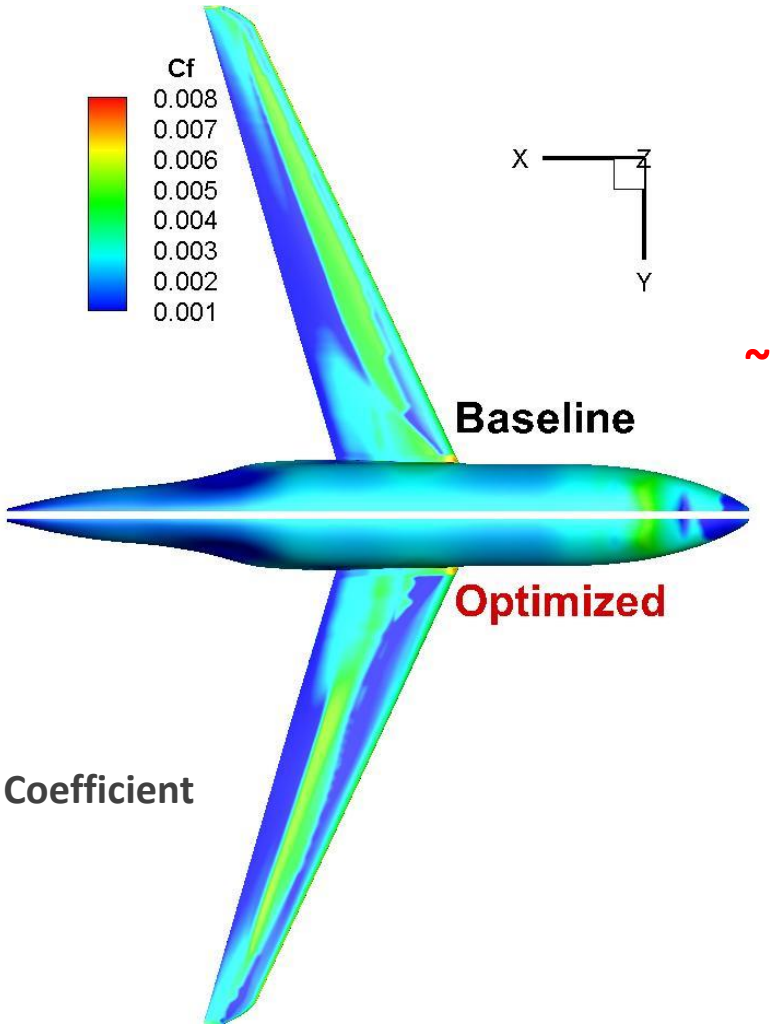
~17% reduction in C_D
C_L=0.5

PUMA by PCOpt/NTUA

Step 2:

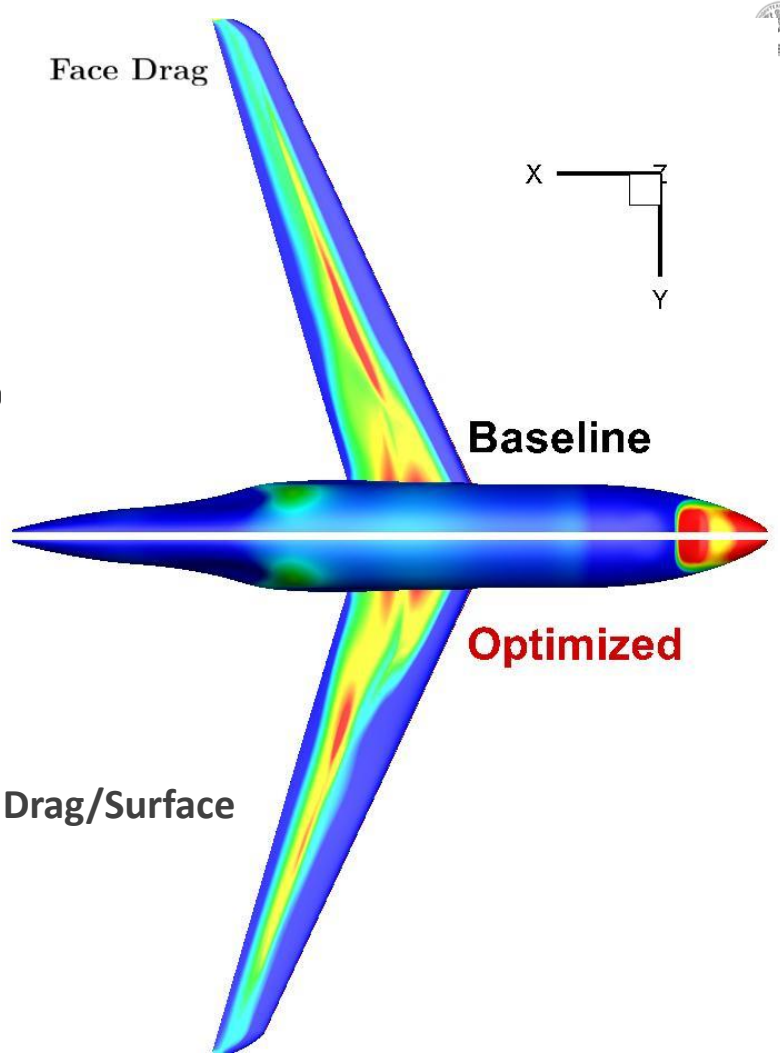


Analysis of a HARW Business Jet (8/8)



Skin Friction Coefficient

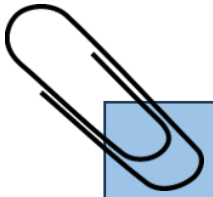
~17% reduction at C_D



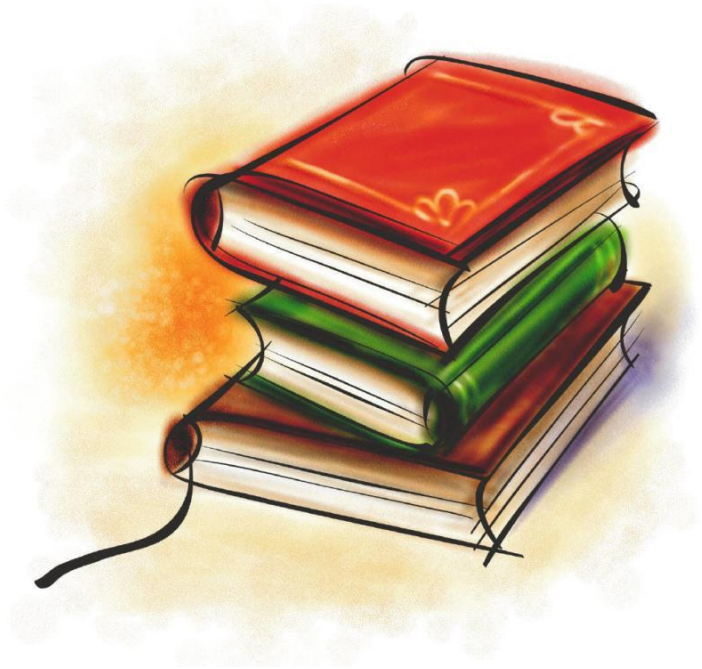
Drag/Surface

PUMA by PCOpt/NTUA

Read more in:



- **M.G. Kontou, "The Continuous Adjoint Method with Consistent Discretization Schemes for Transitional Flows and the Use of Deep Neural Networks in Shape Optimization in Fluid Mechanics", PhD Thesis, National Technical University of Athens, 2023.**
- **M.G. Kontou, X.S. Trompoukis, V.G. Asouti, K.C. Giannakoglou: 'The Continuous Adjoint Method to the γ-Reθt Transition Model Coupled with the Spalart-Allmaras Model for Compressible Flows', International Journal for Numerical Methods in Fluids ; doi.org/10.1002/fld.5319, 2024.**
- **M.G. Kontou, X.S. Trompoukis, V.G. Asouti, K.C. Giannakoglou: 'The Continuous Adjoint Method to the γ-Reθt Transition Model Coupled with the Spalart-Allmaras Model for Compressible Flows', International Journal for Numerical Methods in Fluids ; doi.org/10.1002/fld.5319, 2024.**



The Think-Discrete Do-Continuous (TDDC) Adjoint in Multi-dimensional Flows

The Think-Discrete Do-Continuous (TDDC) Adjoint – Why?

- **Continuous Adjoint (First differentiate-then discretize)**
- **Discrete Adjoint (First discretize-then differentiate)**

Adjoint	Physical Insight	Low Memory Footprint	Consistency	Possible Modifications
Continuous	✓	✓	✘	✓
Discrete	✘	✘	✓	✘

The Think-Discrete Do-Continuous (TDDC) Adjoint – Why?

- **Continuous Adjoint (First differentiate-then discretize)**
- **Discrete Adjoint (First discretize-then differentiate)**

Adjoint	Physical Insight	Low Memory Footprint	Consistency	Possible Modifications
Continuous	✓	✓	✗	✓
Discrete	✗	✗	✓	✗
Think-Discrete Do-Continuous	✓	✓	✓	✓

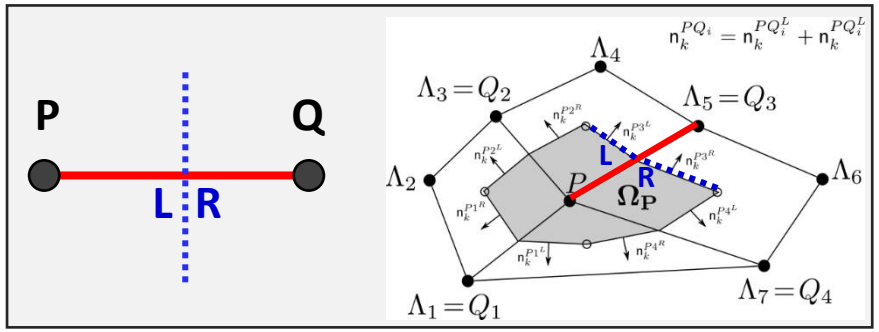
The *TDDC* Adjoint bridges the gap between the two adjoint variants and combines the best of both worlds. The concept is to develop consistent discretisation schemes for the continuous adjoint equations, inspired by (hand-differentiated) discrete adjoint.

TDDC Adjoint – Multi-Dimensional Flows

❑ **Discretisation of the Primal inviscid fluxes (Roe):**

$$F_n^{PQ}=\frac{1}{2}\left(A_{nmk}^{P}U_m^{P}+A_{nmk}^{Q}U_m^{Q}\right)\mathsf{n}_k^{PQ}-\frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(U_m^{R}-U_m^{L}\right)$$

$$V_m^L=V_m^P+\frac{1}{2}\mathsf{t}_\ell^{PQ}\left.\frac{\partial V_m}{\partial x_\ell}\right|^P$$



❑ **"Standard" Continuous Adjoint Discretisation:**

$$\Phi_{m,stdCA}^{PQ}=-\frac{1}{2}A_{nmk}^{P}\mathsf{n}_k^{PQ}\left(\Psi_n^P+\Psi_n^Q\right)-\frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(\Psi_n^R-\Psi_n^L\right)$$

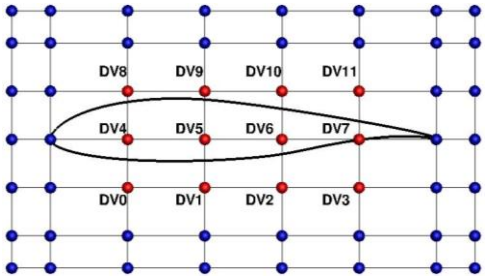
❑ ***TDDC*** **Adjoint Discretisation:**

$$\Phi_{m,TDDC}^{PQ}=\Phi_m^{PQ}=-\frac{1}{2}A_{nmk}^{P}\mathsf{n}_k^{PQ}\left(\Psi_n^P+\Psi_n^Q\right)-\frac{1}{2}\left[\left[\left|\tilde{A}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right]^{R,\mathrm{adj}}-\left[\left|\tilde{A}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right]^{L,\mathrm{adj}}\right]\left.\frac{\partial V_\ell}{\partial U_m}\right|^P$$

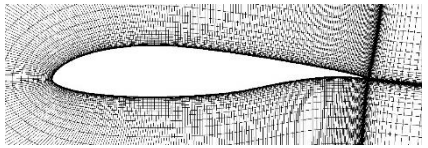
$$\left|\tilde{A}_{n\ell k}^{LR}\mathsf{n}_k\right|=\left|\tilde{A}_{nqk}^{LR}\mathsf{n}_k\right|\frac{\partial U_q^L}{\partial V_\ell^L}-\left(U_q^R-U_q^L\right)\frac{\partial\left|\tilde{A}_{nqk}^{LR}\mathsf{n}_k\right|}{\partial V_\ell^L}$$

$$\left[\beta_{n\ell}^{LR}\Psi_n\right]^{L,\mathrm{adj}}=\beta_{n\ell}^{LR}\Psi_n^P+\frac{1}{2}\left[\beta_{n\ell}\mathsf{t}_r\frac{\partial\Psi_n}{\partial x_r}\right]_{PQ}^{P,\mathrm{adj}}$$

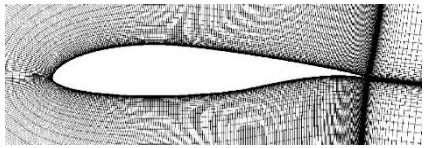
TDDC Adjoint: ShpO of the NLF(1)-0416 Isolated Airfoil (1/3)



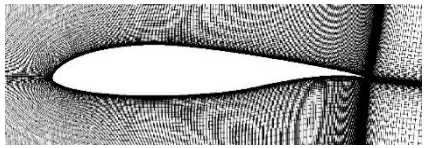
- Transitional Flow: M_∞=0.1, Re=4·10^6, Tu=0.15%, AoA=2^o. Parametrisation: 8x7 NURBS lattice.
- Performance of "Standard" continuous and *TDDC* adjoint on three grids. Comparison with FDs.
- **The TDDC adjoint is accurate even on coarse grids!**



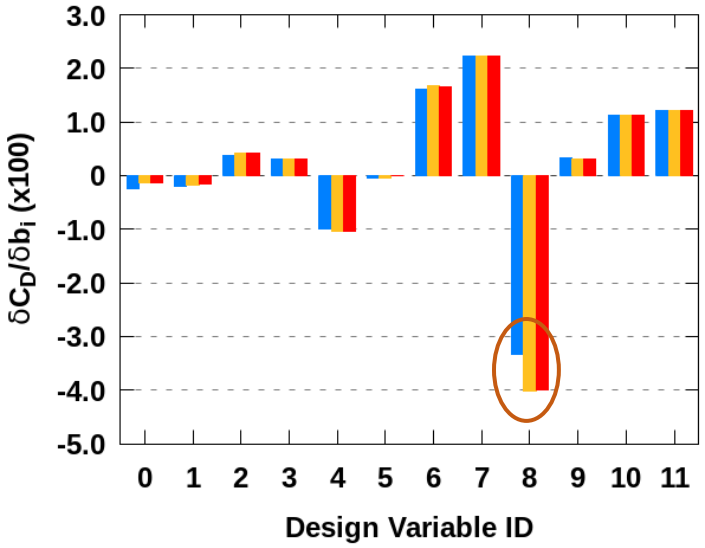
529x73 nodes

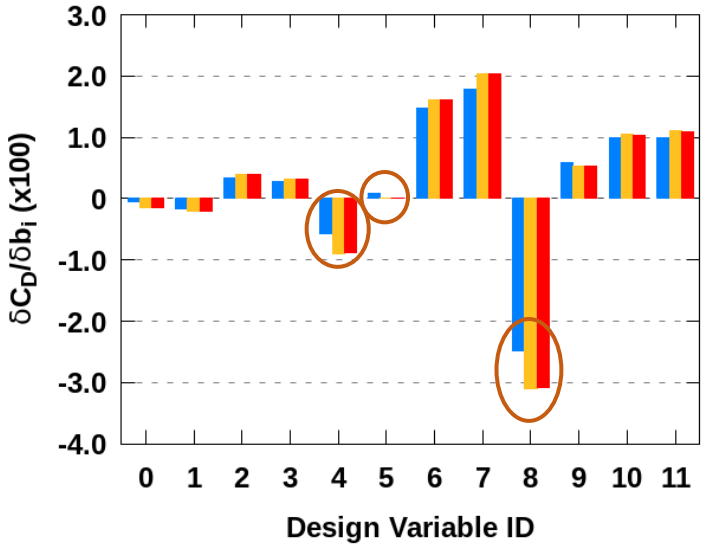


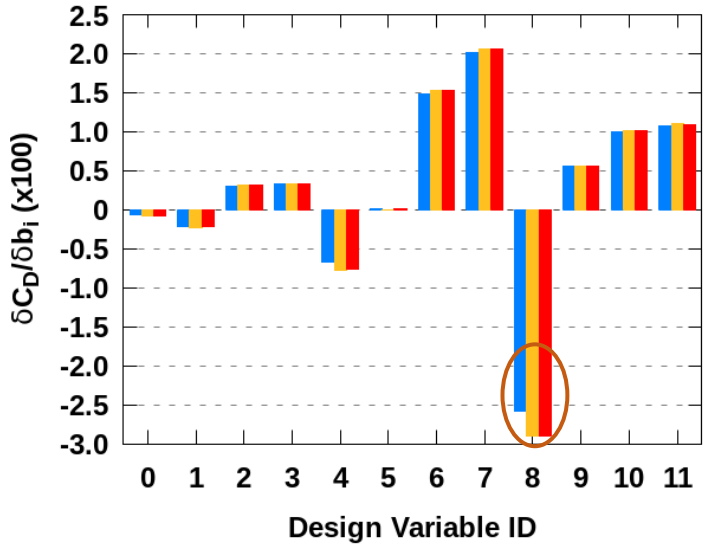
705x97 nodes



1057x145 nodes

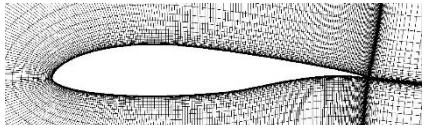




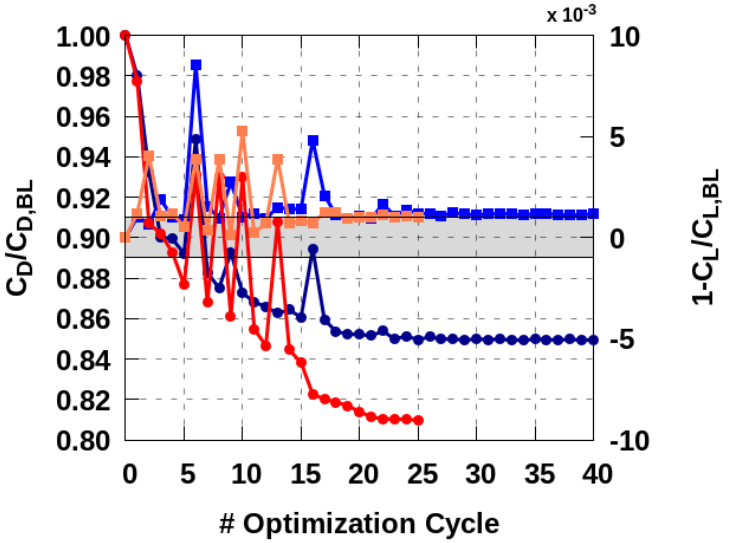


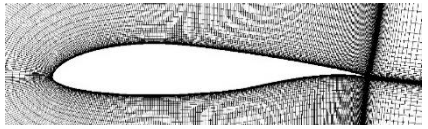
TDDC Adjoint: ShpO of the NLF(1)-0416 Isolated Airfoil (2/3)

- ShpO: min. C_D constrained by C_L (±1‰) and area (≥90%) of the baseline values.
- Comparison of "Standard" continuous and *TDDC* adjoint.

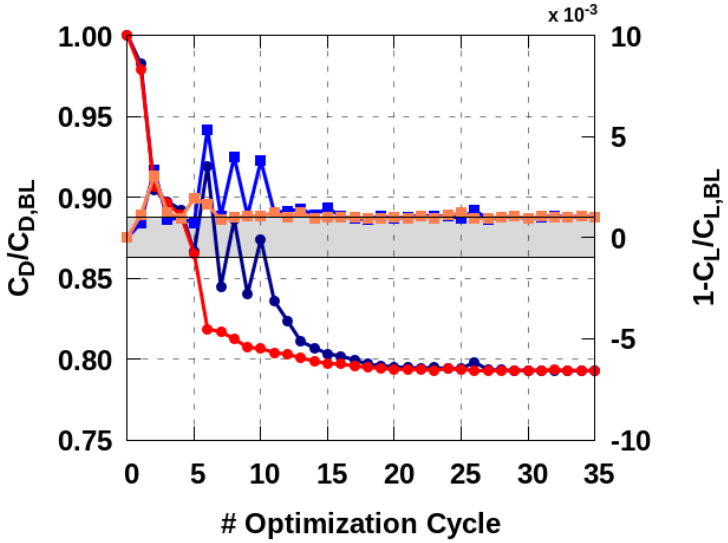


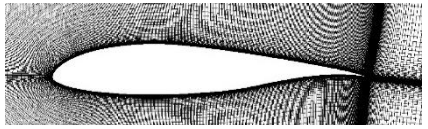
529x73 nodes



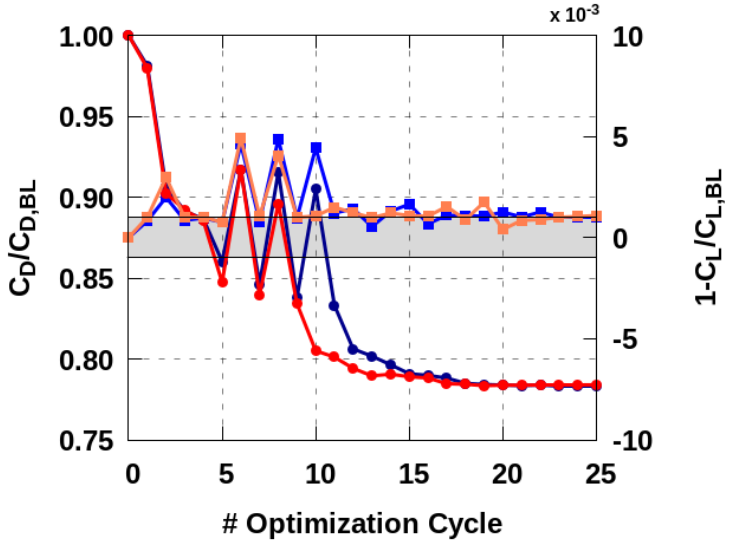


705x97 nodes



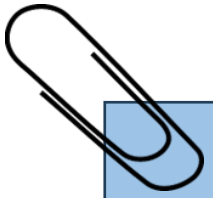


1057x145 nodes

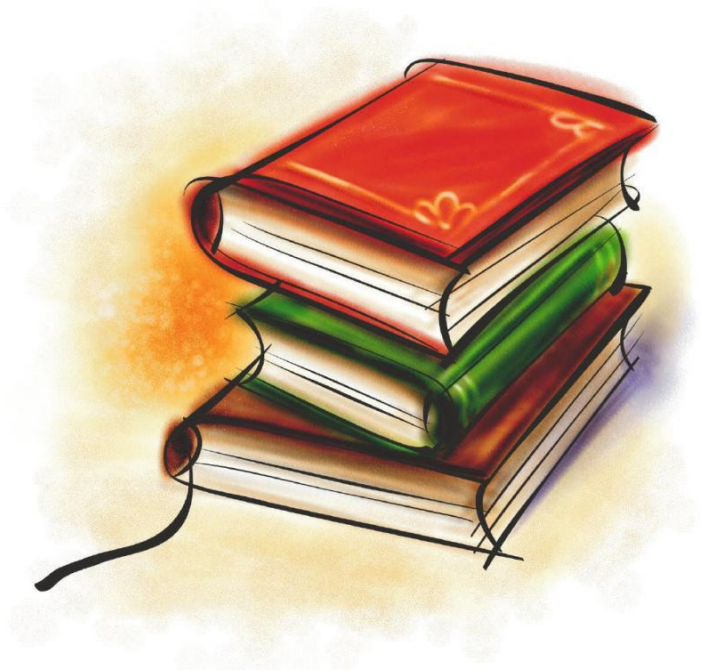


The TDDC reaches a solution of same/better quality in less optimisation cycles

Read more in:



- **M.G. Kontou, "The Continuous Adjoint Method with Consistent Discretization Schemes for Transitional Flows and the Use of Deep Neural Networks in Shape Optimization in Fluid Mechanics", PhD Thesis, National Technical University of Athens, 2023.**
- **M.G. Kontou, V.G. Asouti, X.S. Trompoukis, K.C. Giannakoglou, "Consistently discretized continuous adjoint equations: The Think-Discrete Do-Continuous adjoint", Computer Methods in Applied Mechanics and Engineering 2026; 449:118529.**



The Recursive Projection Method (RPM) in Primal & Adjoint Solvers

Benefits of using RPM in both Flow & Adjoint Solvers

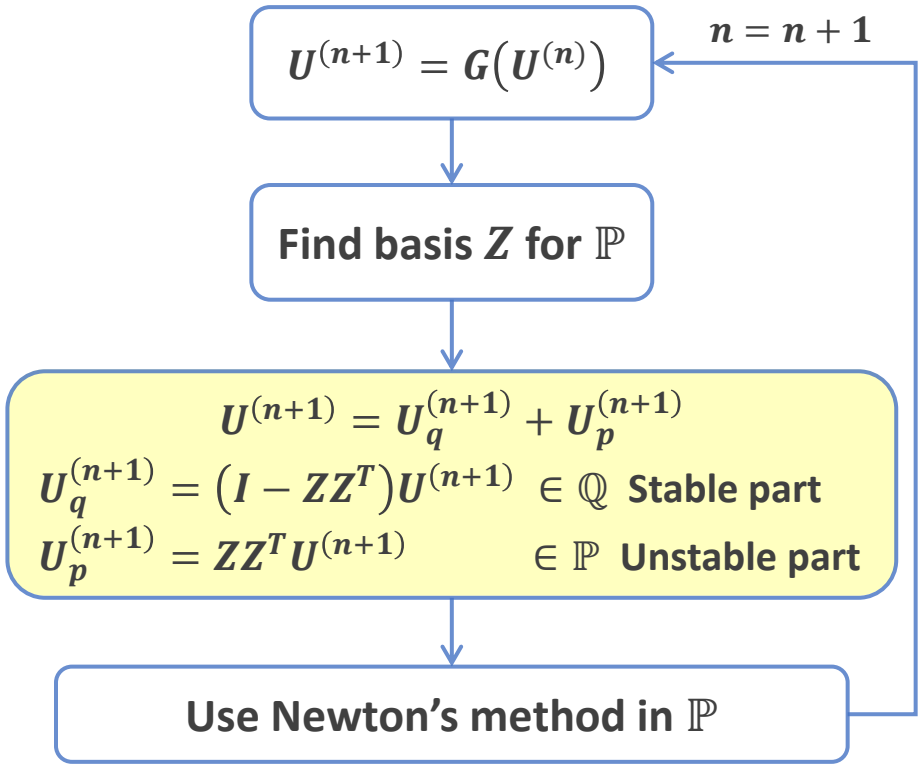
The Recursive Projection Method (RPM) is a stabilisation method for iterative solvers:

$$U^{new} = G(U^{old}).$$

In cases with mild flow unsteadiness, the RPM allows steady primal and/or adjoint solvers to converge. Gain: steady solvers can be used in the optimisation.

Even in steady cases, RPM appears to be the remedy to stability issues caused by the Adjoint Transpose Term (ATC) in the adjoint equations!

$$R_i^u = u_j \frac{\partial v_j}{\partial x_i} L - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial \tau_{ij}^{\alpha}}{\partial x_j} + \frac{\partial q}{\partial x_i} = 0$$



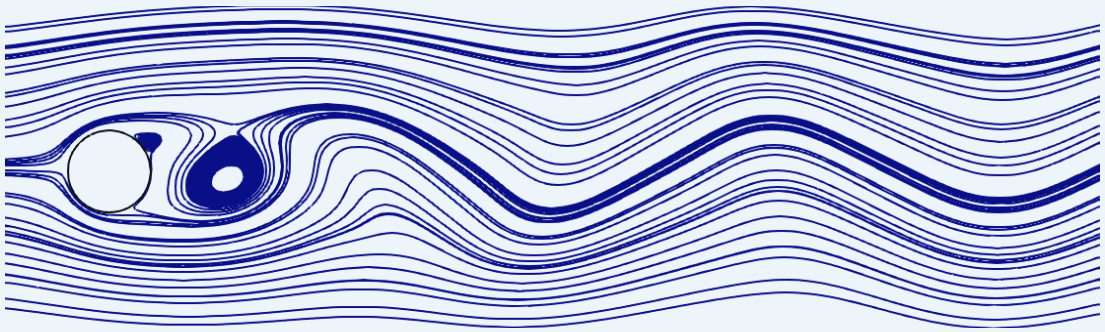
Research funded by

Int J Numer Meth Fluids. 93: 2677– 2693. 2021.

Benefits of using RPM in both Flow & Adjoint Solvers

A ShpO example:

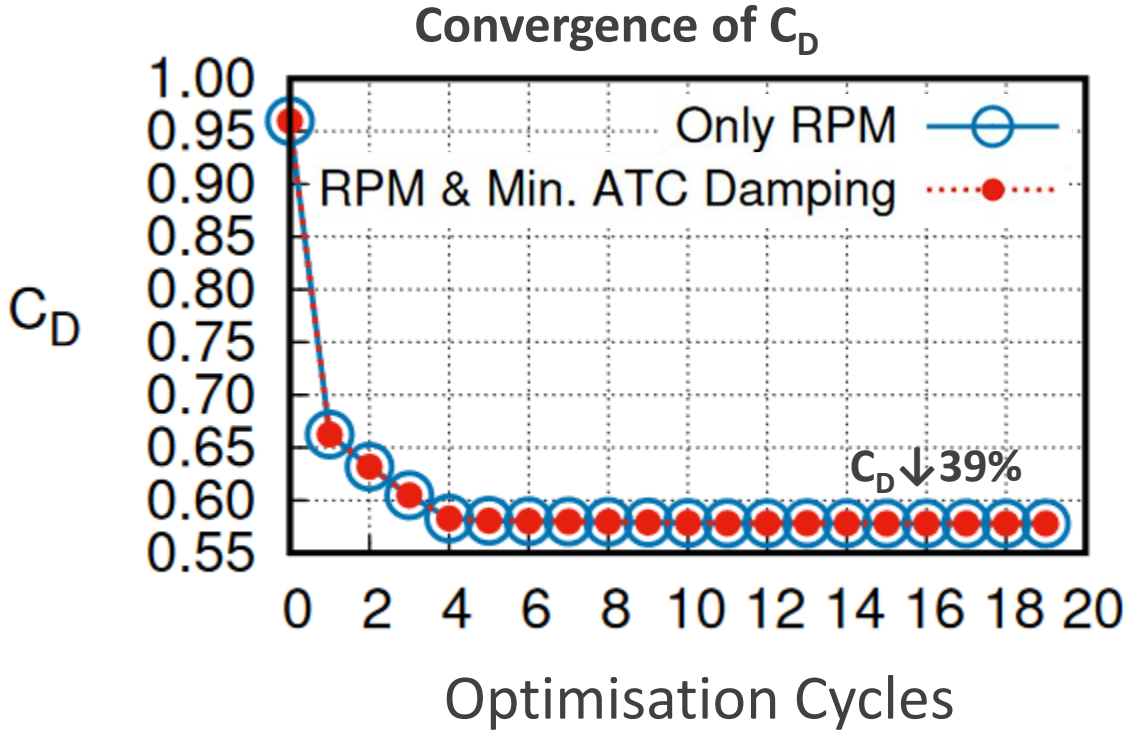
- **Re=140 - periodic flow**

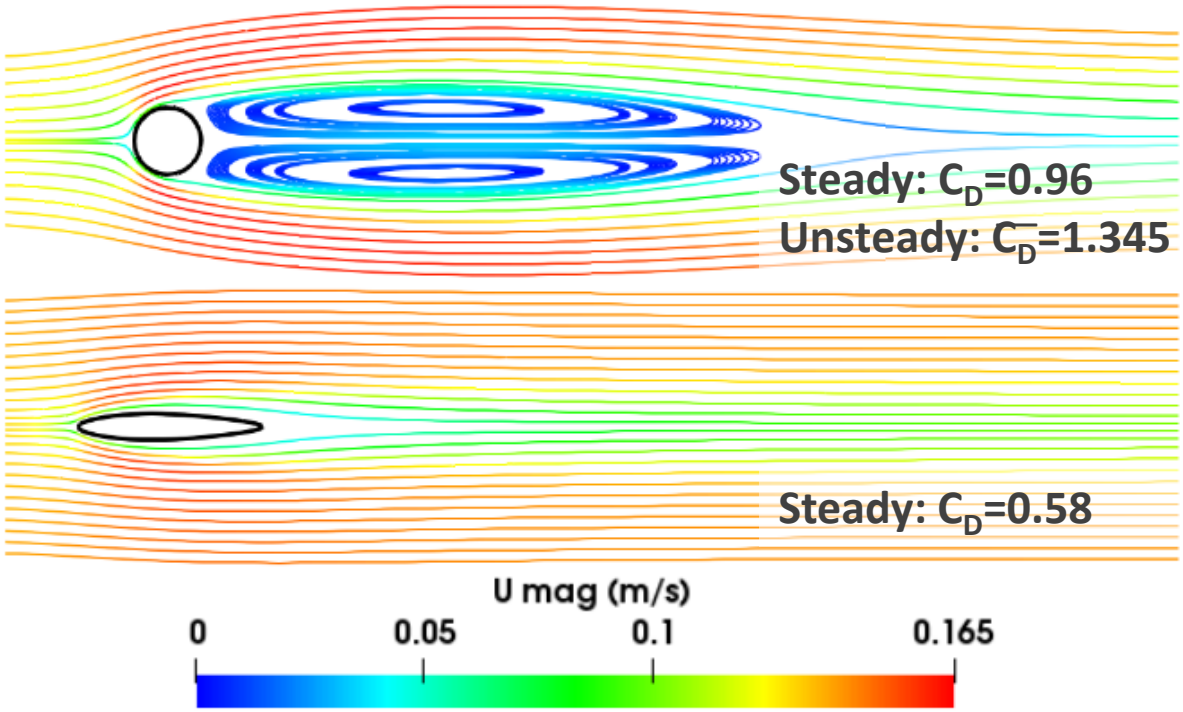


- **Objective: min C_D**
- **Constraint: Constant cross-sectional area (V)**
- **Parametrisation: Volumetric B-Splines (VBS)**

Int J Numer Meth Fluids. 93: 2677– 2693. 2021.

Benefits of using RPM in both Flow & Adjoint Solvers

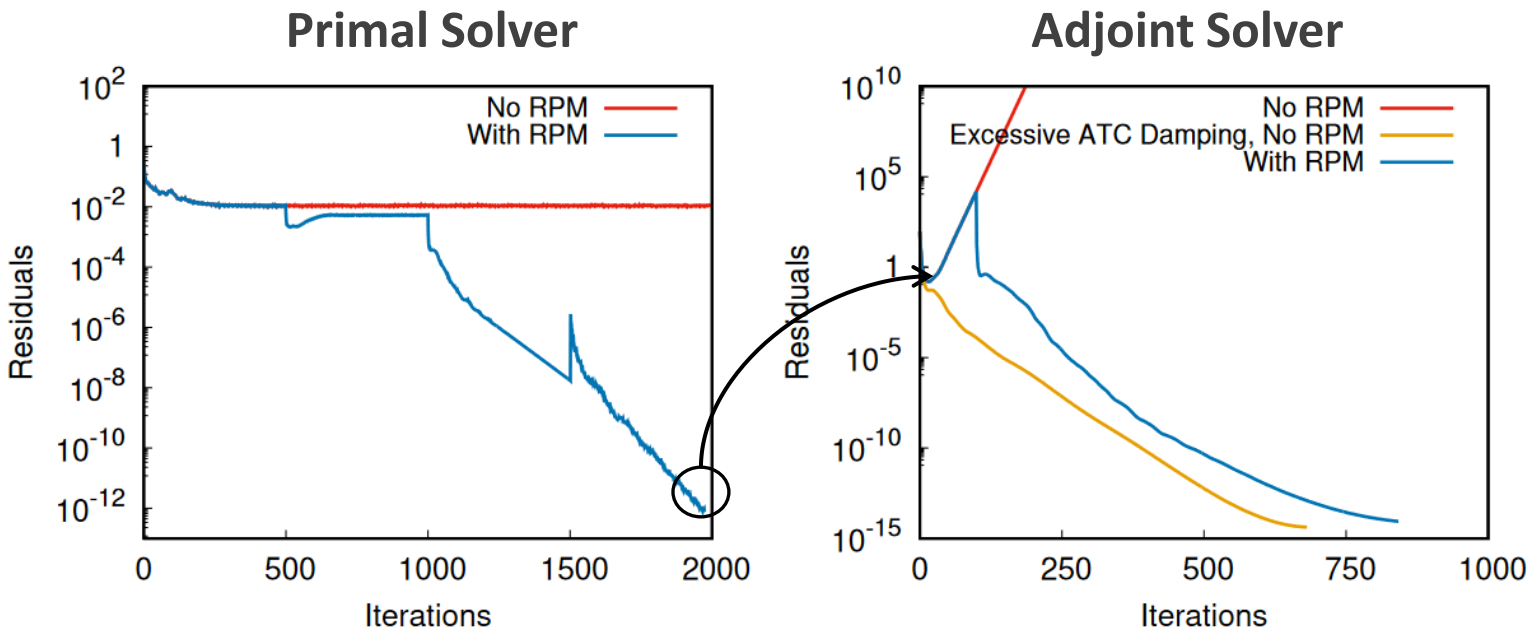




ShpO was performed at low cost, exclusively using steady solvers assisted by the RPM.
See convergence plot (next slide).

Int J Numer Meth Fluids. 93: 2677– 2693. 2021.

Benefits of using RPM in both Flow & Adjoint Solvers

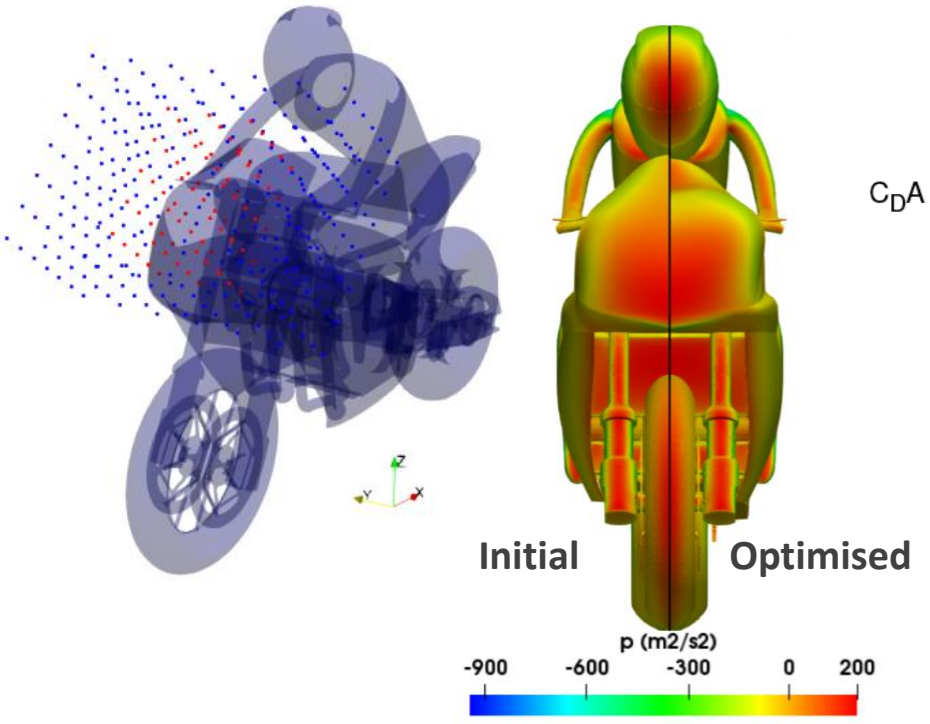


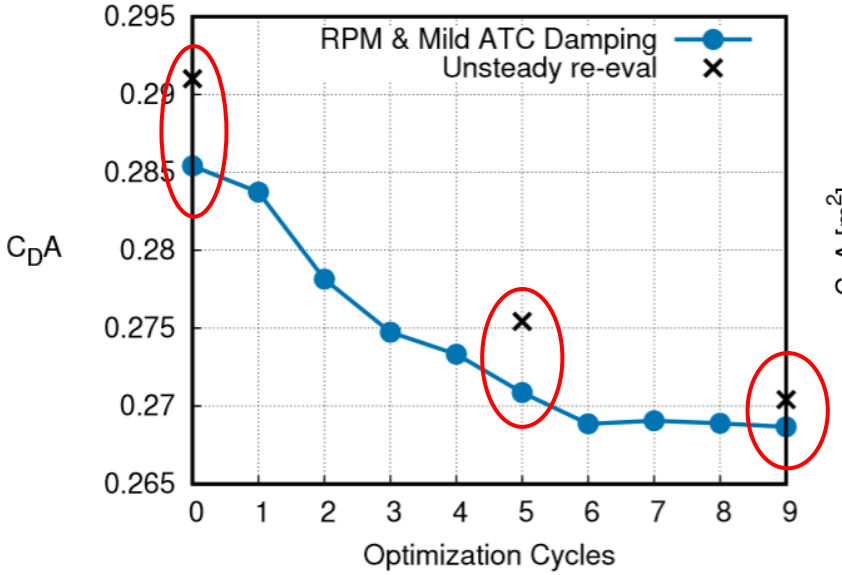
- **The RPM stabilised the primal solver. A steady flow field was computed.**
- **With this steady primal solution, the adjoint solver diverged. The RPM stabilised it.**
- **Without the RPM, excessive ATC damping was required to stabilise the adjoint equations.**

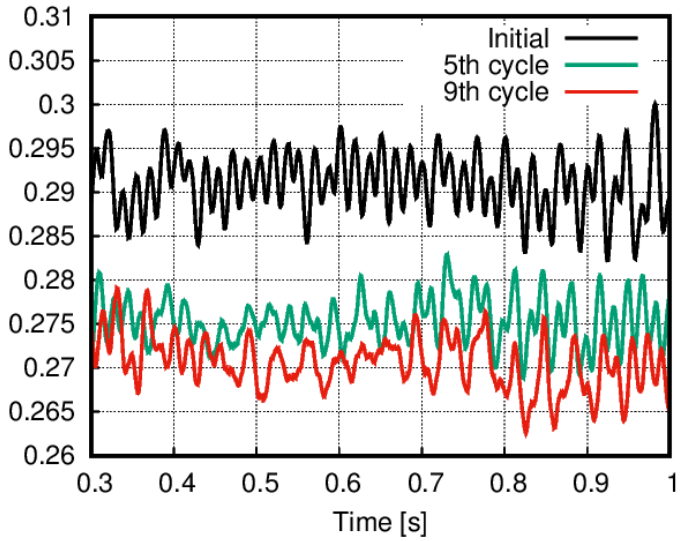
Int J Numer Meth Fluids. 93: 2677– 2693. 2021.

Benefits of using RPM in both Flow & Adjoint Solvers

Use of the RPM to stabilize steady primal & adjoint solvers in case of convergence issues due to mild flow unsteadiness. The gain is that steady solvers can be used in the optimisation.



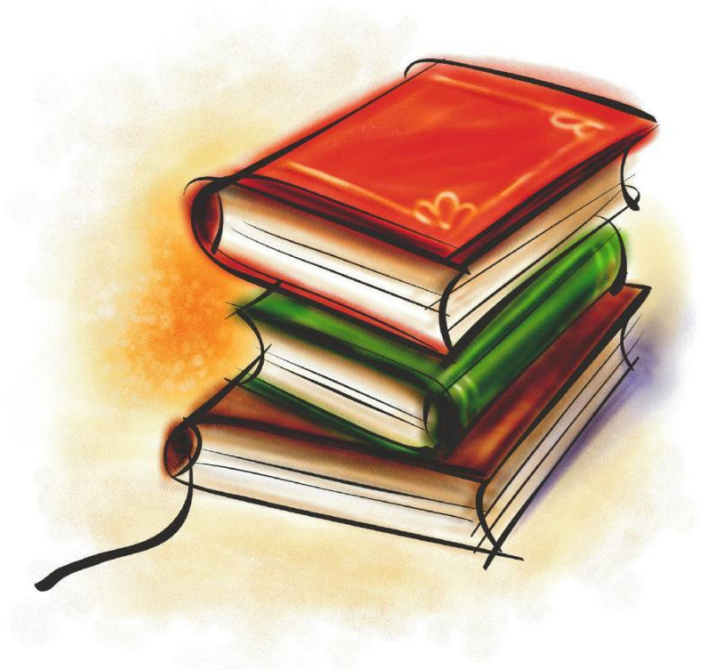




Optimisation cost (on 156 cores):

- **Steady: 20h for 9 opt. cycles**
- **Unsteady: 28h for 1 flow evaluation**

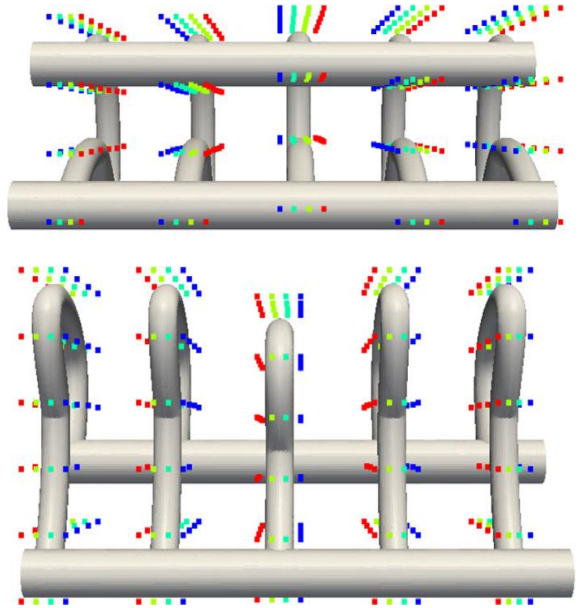
Int J Numer Meth Fluids. 93: 2677– 2693. 2021.



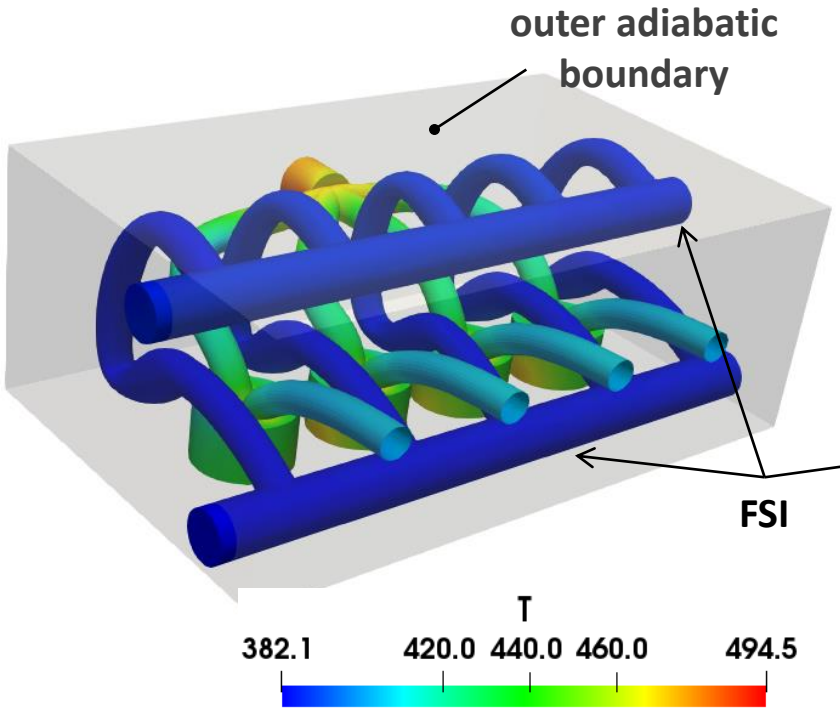
ShpO with Conjugate Heat Transfer (CHT)

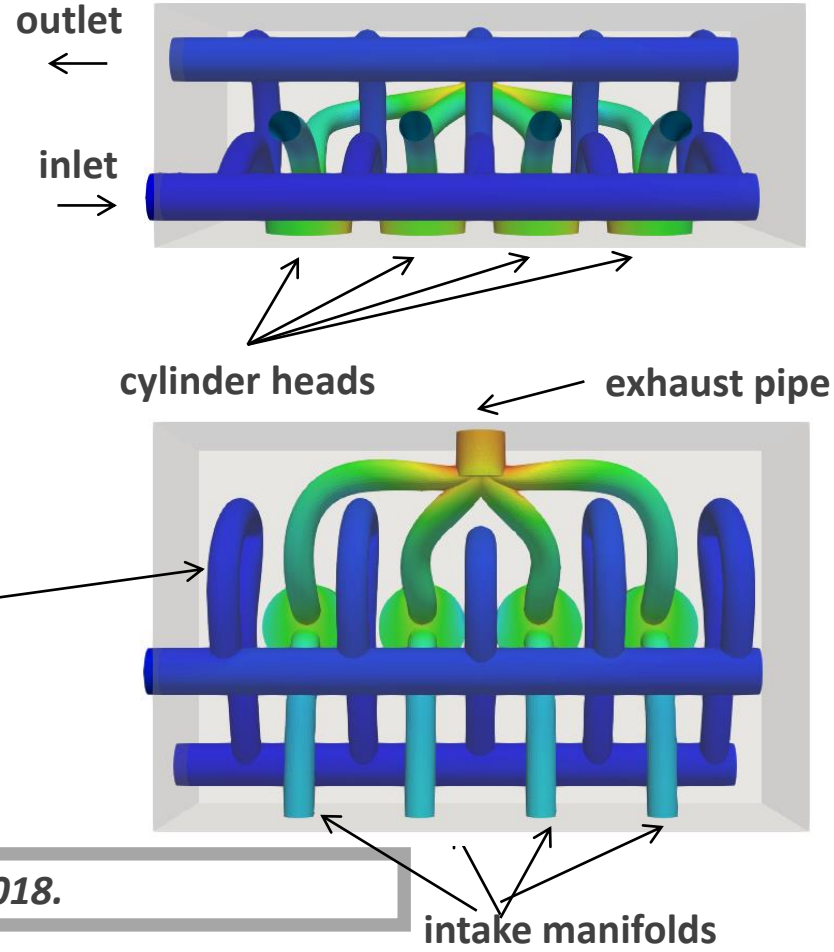
ShpO with CHT- Car Engine Cylinder Head Cooling

- **Redesign/optimisation of the cooling channel in the upper block of the piston engine. Trade-off between best cooling (min. solid volume with $T>T_{threshold}$) and min. total pressure losses. In short: $J^{pen,T}$ and Δp_t.**
- **Along the air-intake manifolds, the four cylinders and the exhaust pipe, a constant heat flux is imposed.**



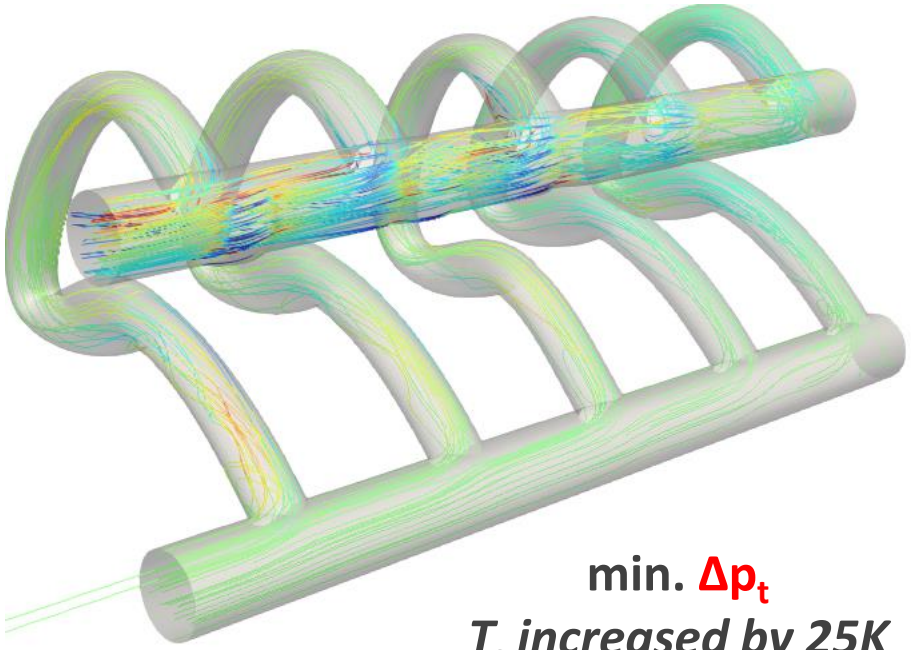
Parametrisation





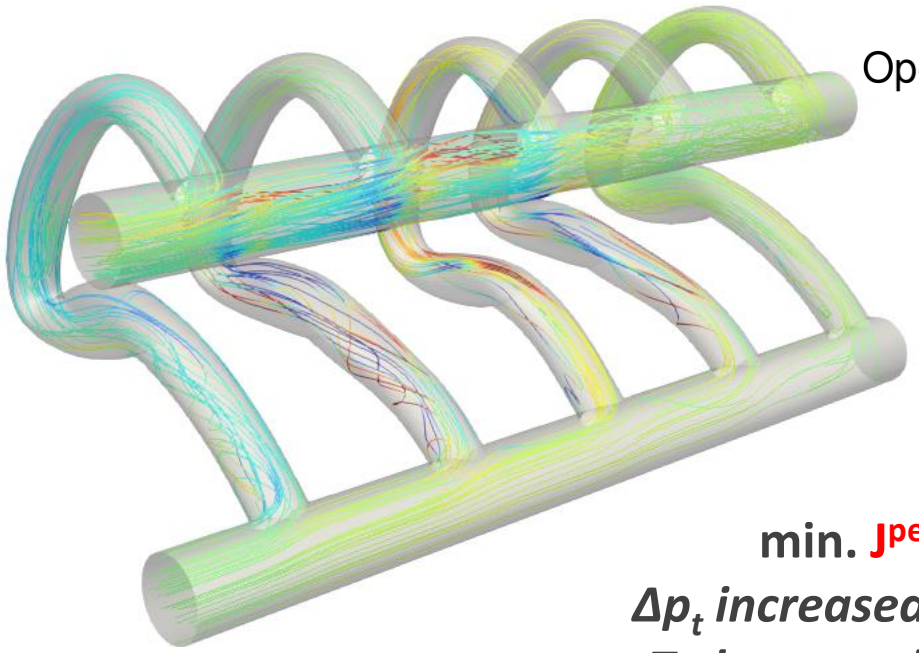
Applied Thermal Engineering, 140:351-362, 2018.

ShpO with CHT- Car Engine Cylinder Head Cooling



min. Δp_t

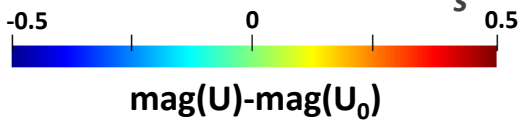
T_s increased by 25K



min. $J^{pen,T}$

Δp_t increased by 28%

T_s decreased by 32K

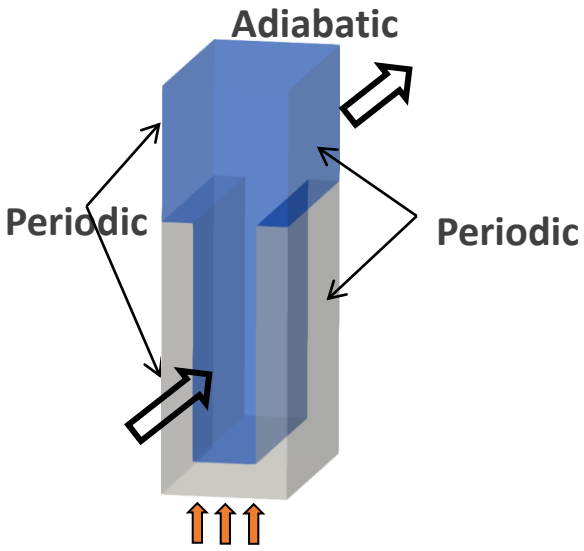


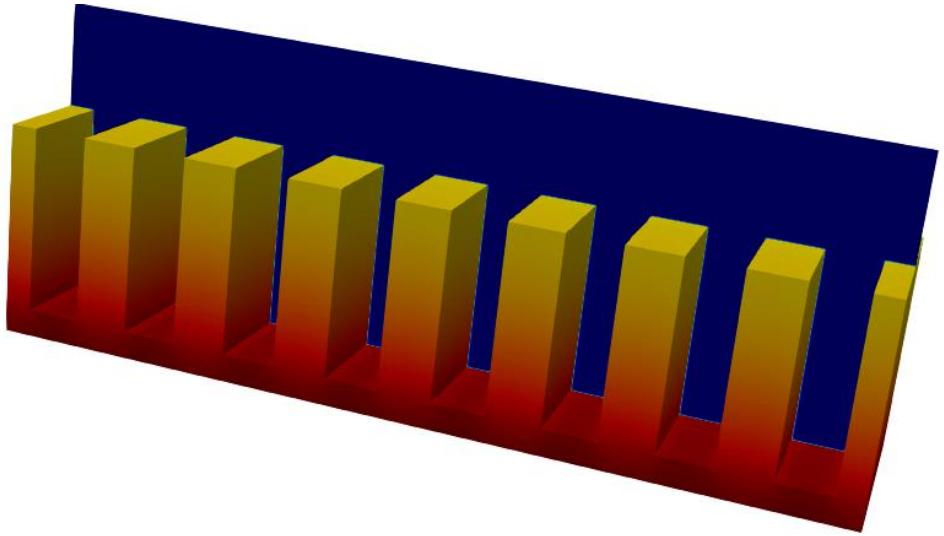
To minimise ΔP_t, the velocity magnitude is reduced (wider cooling channels). To minimise max. T_{solid}, the velocity magnitude slightly increases (narrower channels), and the cooling ducts come closer to the cylinder heads.

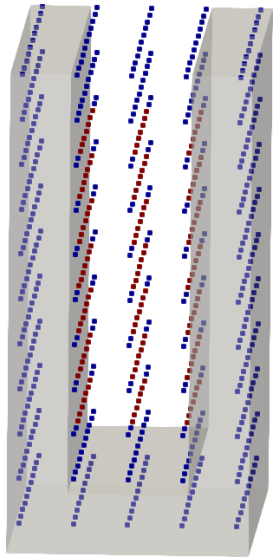
Applied Thermal Engineering, 140:351-362, 2018.

Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

ShpO with CHT- Two-Objective ShpO of Cooling Fins







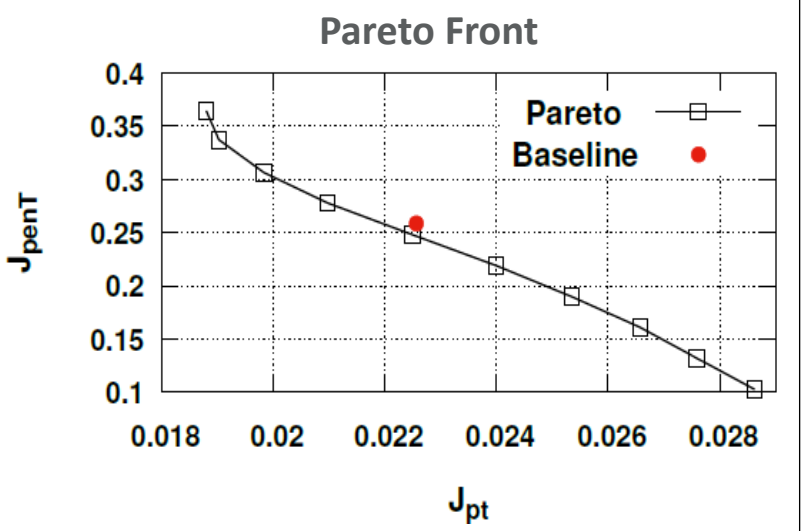
Parametrisation using a morphing box

Two targets:

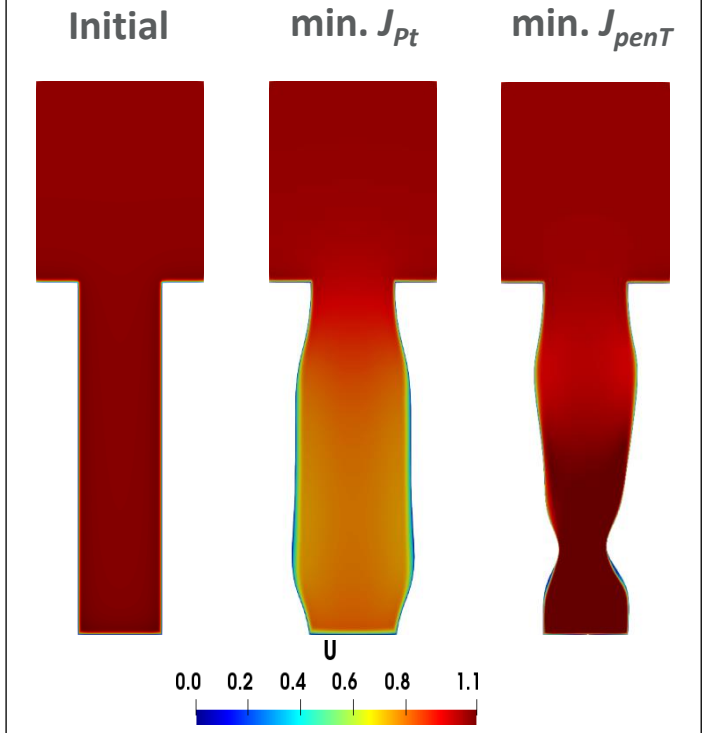
1. min. max. T_{solid} (J^{penT})
2. min. total pressure losses (Δp_t or J^{pt})

Applied Thermal Engineering, 140:351-362, 2018.
7th OpenFOAM Conference, Berlin, 2019.

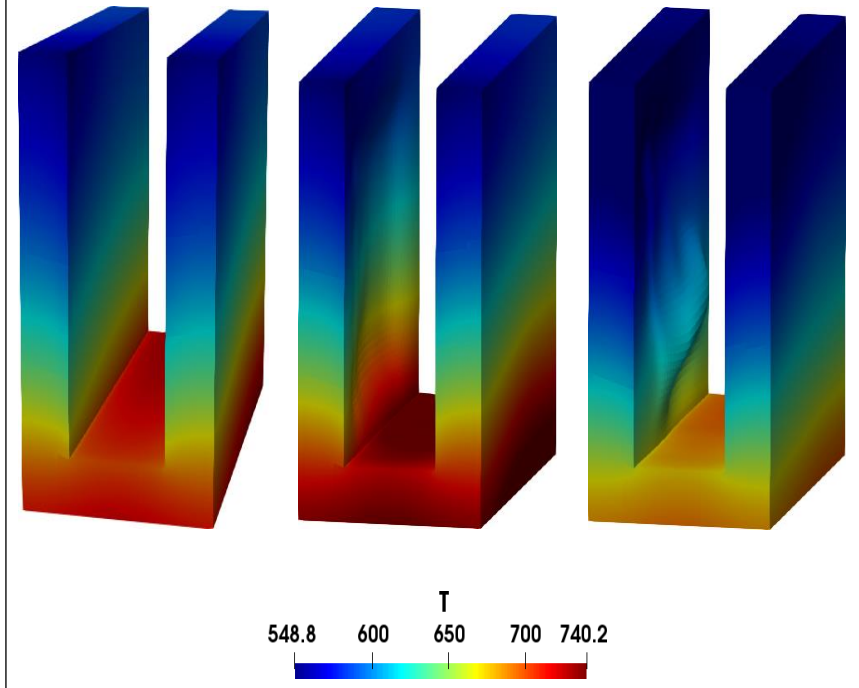
ShpO with CHT- Two-Objective ShpO of Cooling Fins



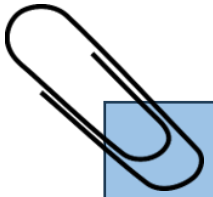
- 1st Pareto Point by solving min. J_{pt}
- Gradient-based optimisation



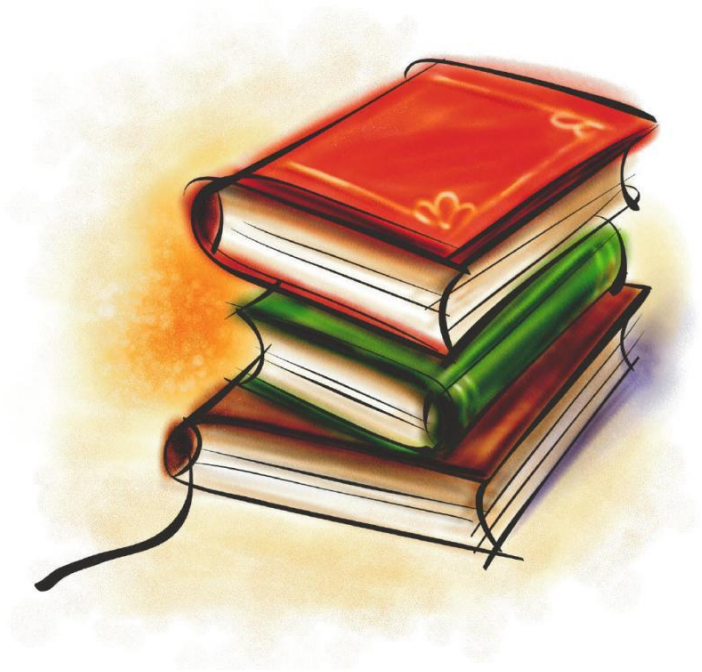
Cross-section at middle of fluid passage length.



Read more in:



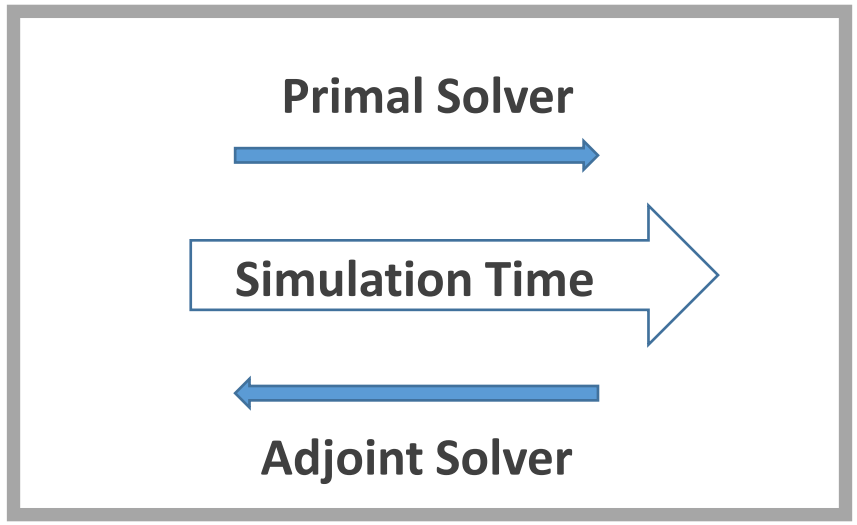
- **K. Gkaragkounis, "The Continuous Adjoint Method in Aerodynamic and Conjugate Heat Transfer Shape Optimization, for Turbulent Flows", PhD Thesis, National Technical University of Athens, 2020.**
- **K.T. Gkaragkounis, E.M. PapoutsisKiachagias, and K.C. Giannakoglou, ''The continuous adjoint method for shape optimization in conjugate heat transfer problems with turbulent incompressible flows''. Applied Thermal Engineering, 140:351362, 2018.**
- **K.T. Gkaragkounis, E.M. PapoutsisKiachagias, A.G. Tsolovikos, K.C Giannakoglou, ''Effect of Grid Displacement Models on Sensitivity Derivatives computed by Continuous Adjoint in Aerodynamic and Conjugate Heat Transfer Shape Optimization''. Engineering Optimization (doi 10.1080/0305215X.2020.1796998).**



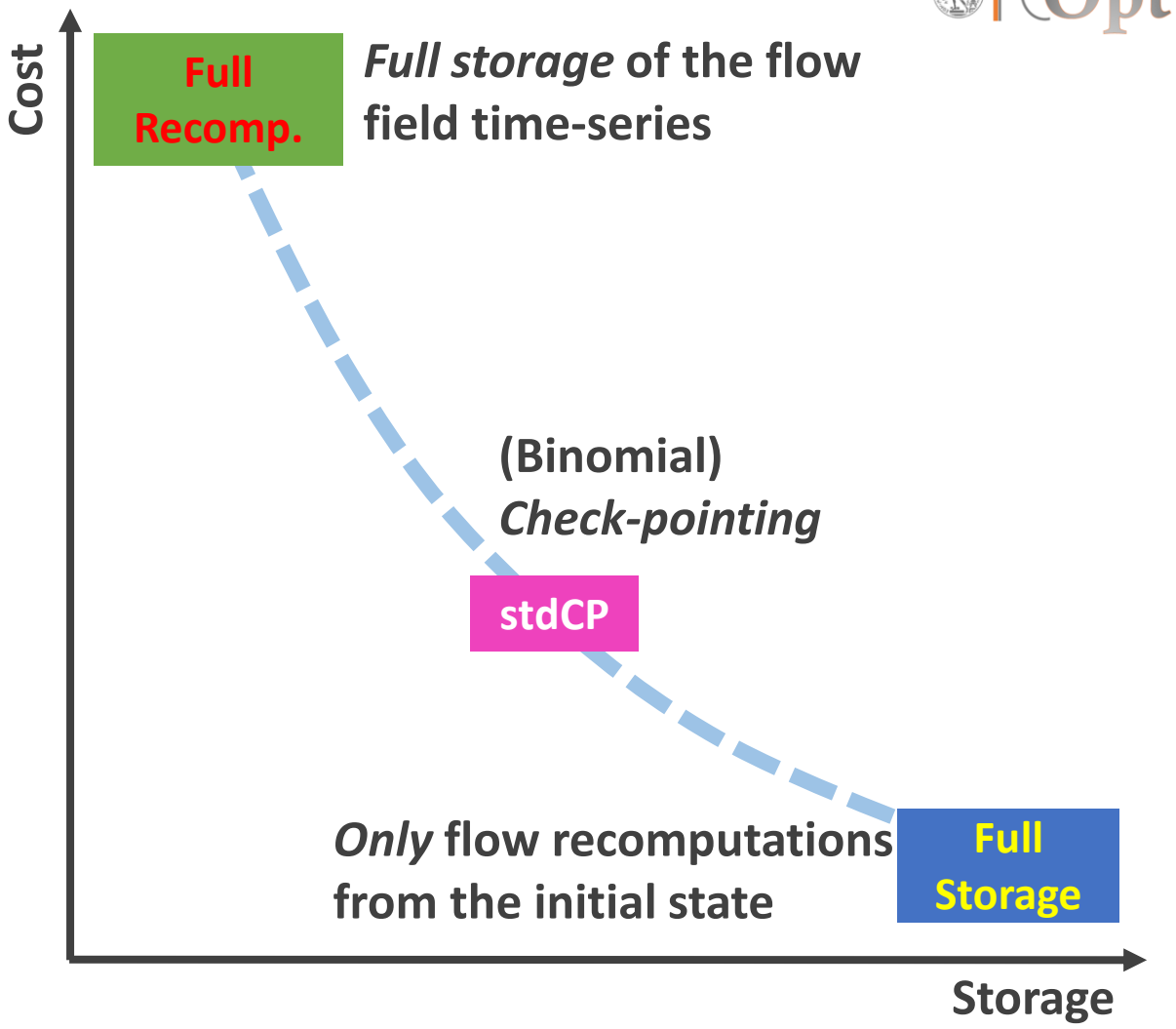
Unsteady Adjoint - Challenges

Unsteady Adjoint

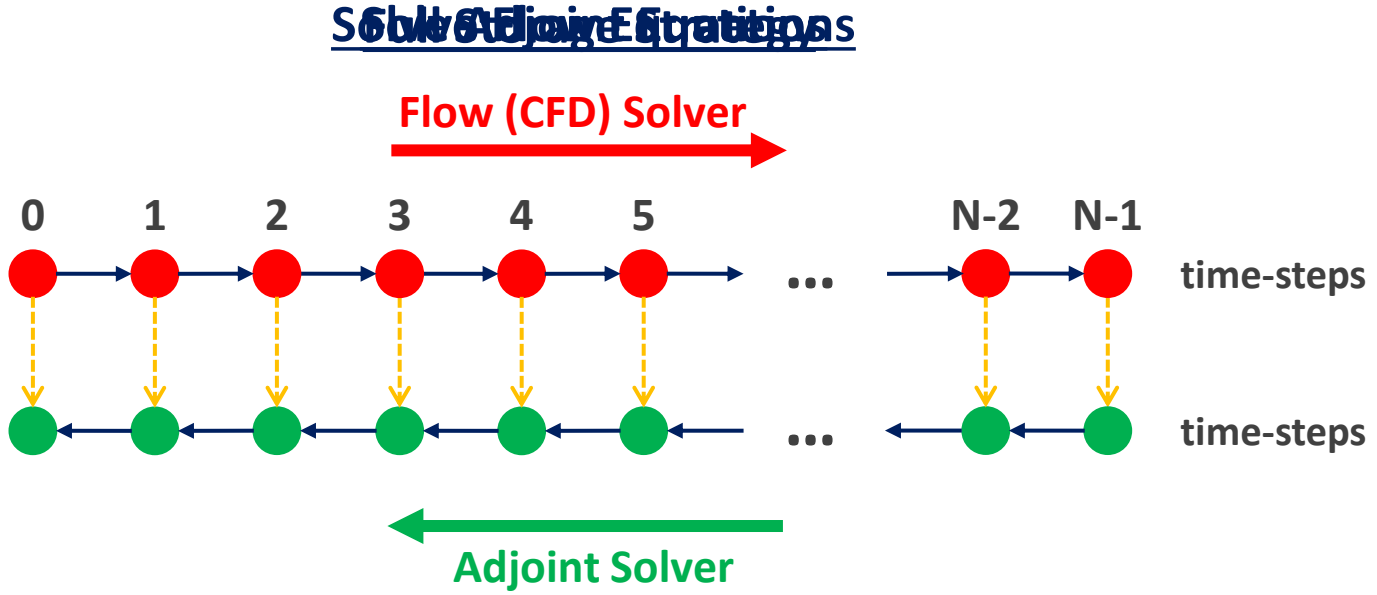
Directionσ of integration of the primal and adjoint equations



<u>Challenge:</u> data management of the primal flow series

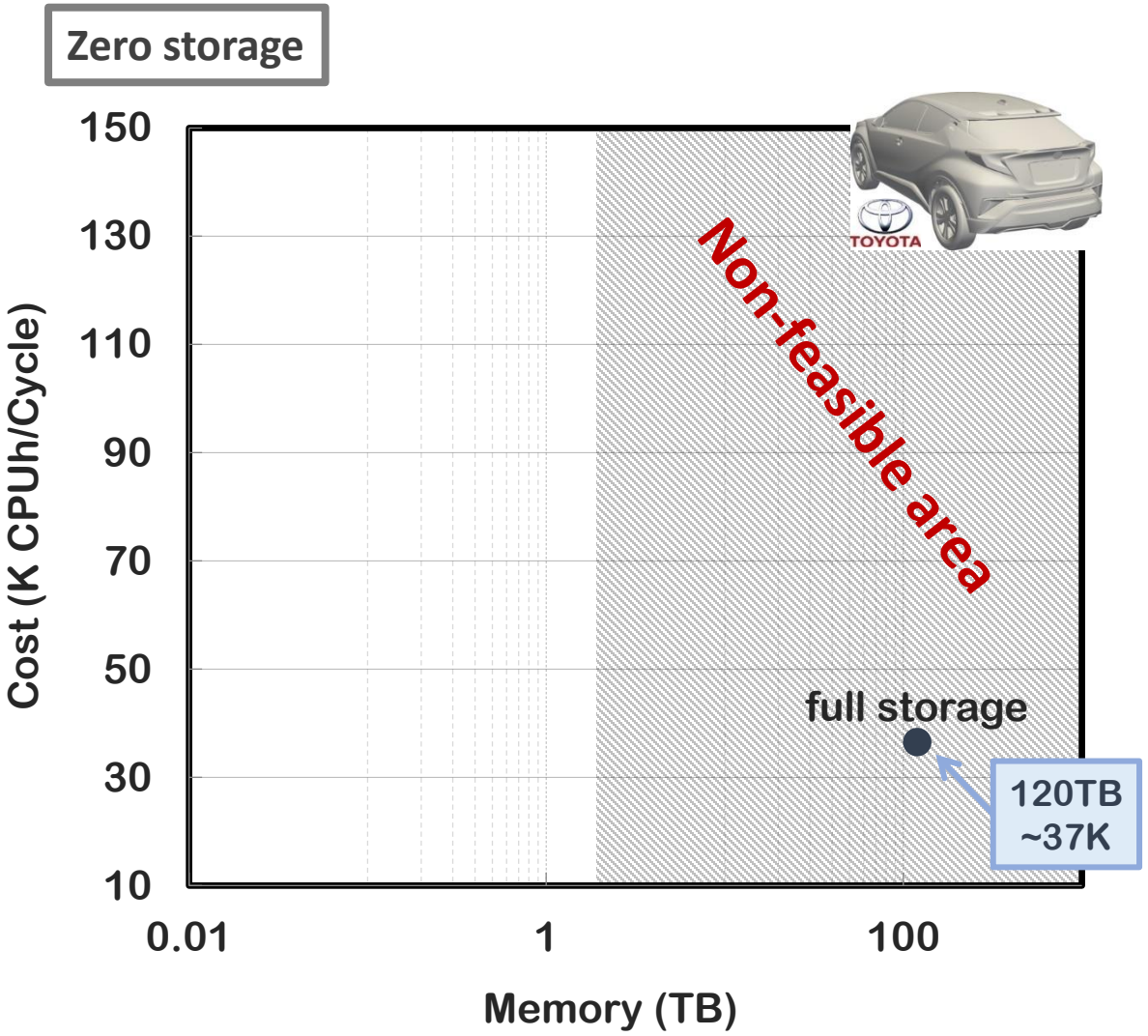


Unsteady Adjoint



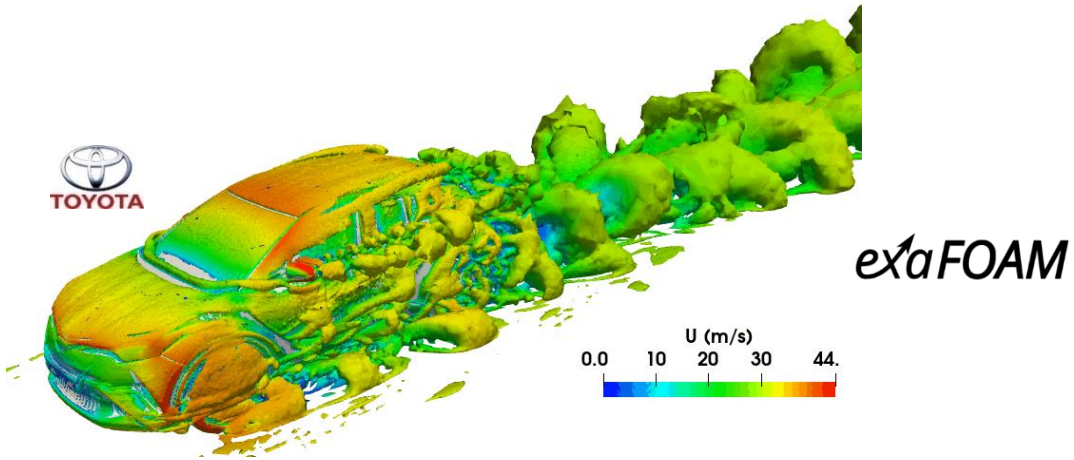
Storage

Unsteady Adjoint – Towards Data Compression (1/3)



Available treatments:

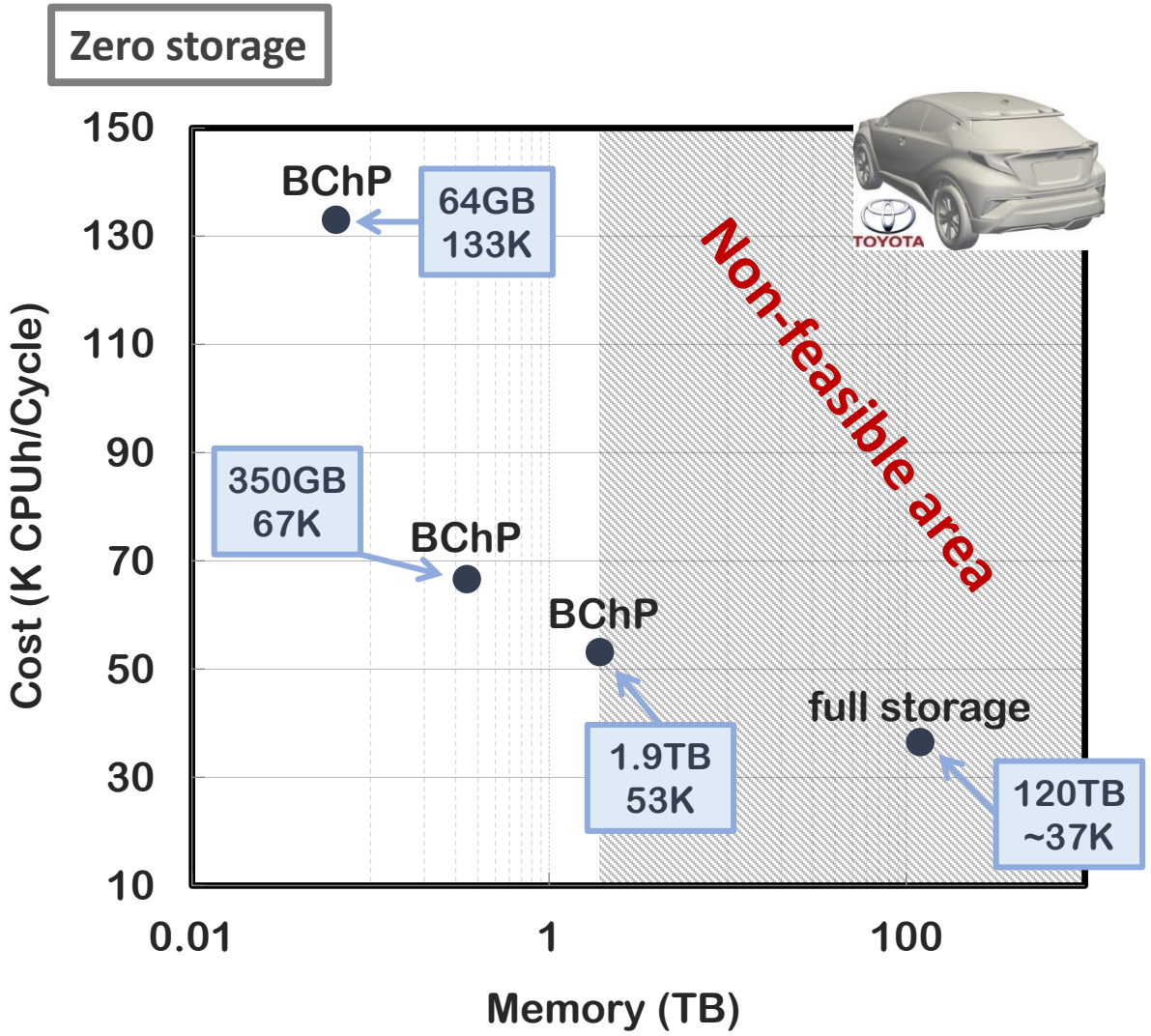
1) **Full storage:** Smallest cost but huge memory requirements especially for large scale problems.
2) **Zero storage** + flow recomputations from initial state.
3) …



75M cells, 28K time-steps

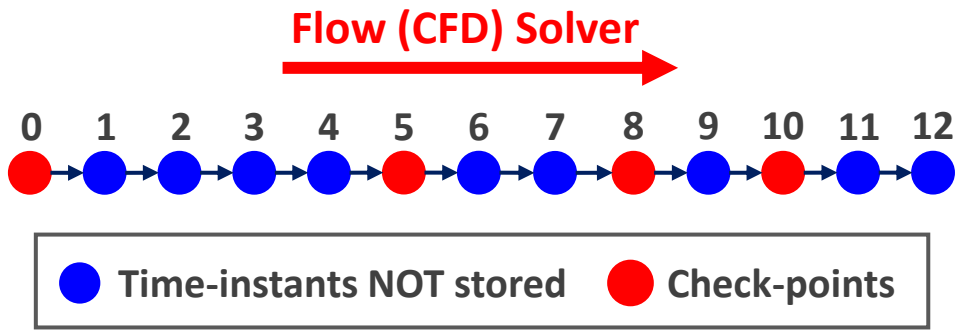
Computation on the National HPC facility - ARIS using 960 cores

Unsteady Adjoint – Towards Data Compression (2/3)



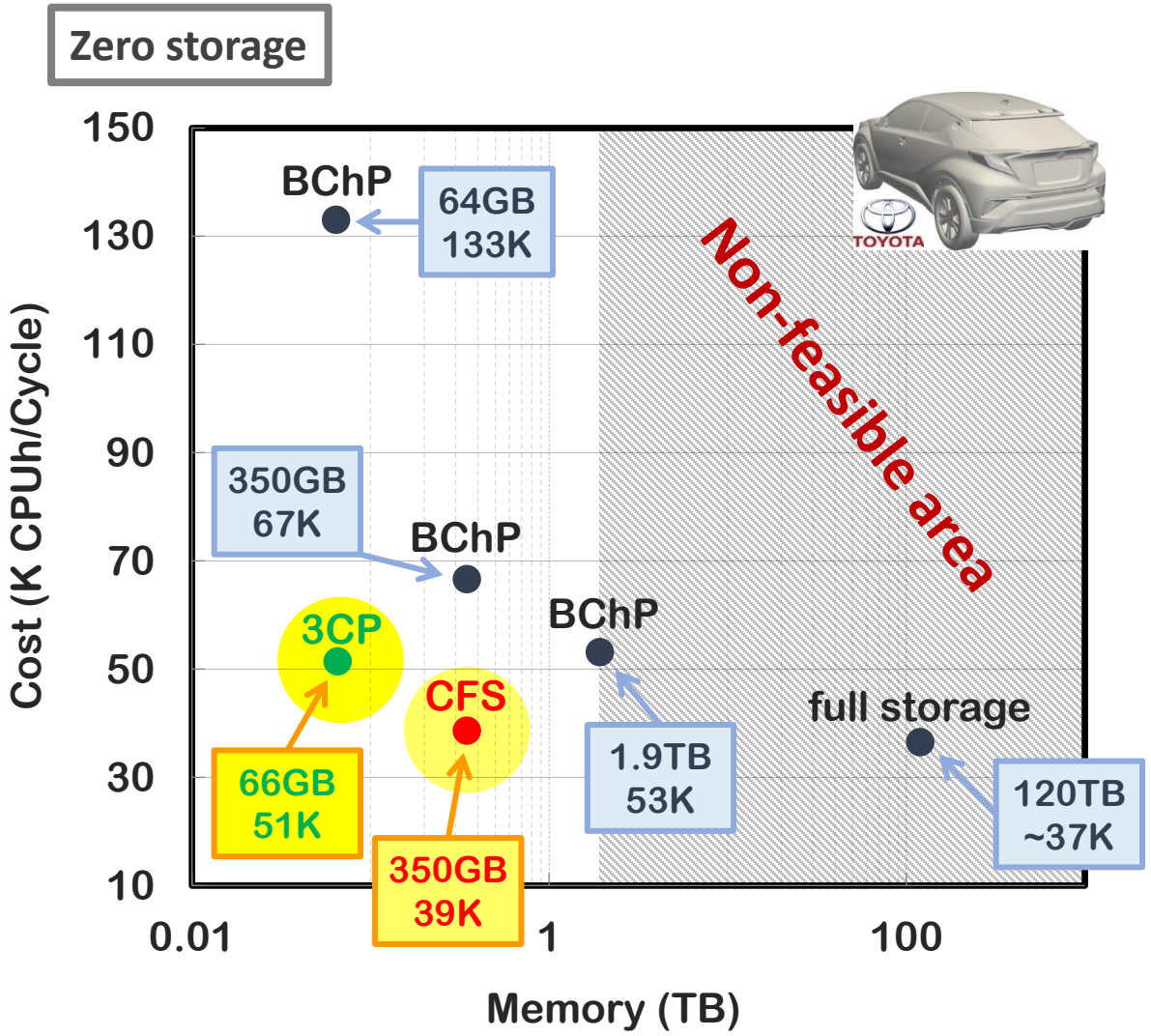
Available treatments:

1) **Full storage:** Smallest cost but huge memory requirements especially for large scale problems.
2) **Zero storage** + flow recomputations from initial state.
3) **(Binomial) Check-pointing (BChP)**: Trade-off between memory & cost.



Storage

Unsteady Adjoint – Towards Data Compression (3/3)



Available treatments:

1) **Full storage:** Smallest cost but huge memory requirements especially for large scale problems.
2) **Zero storage** + flow recomputations from initial state.
3) **(Binomial) Check-pointing (BChP)**: Trade-off between memory & cost.

Storage in compressed form with/without check-pointing:

4) Compressed Full Storage (CFS).
5) Compressed Coarse-grained Check-Pointing (3CP).

Storage

The iPGD Algorithm at a Glance

- **Proper Generalised Decomposition (PGD) approximates a flow field f using a finite sum decomposition:**

$$f(i,k) \equiv f_{i,k} \simeq \sum_{\mu=1}^{M} X_i^{\mu} T_k^{\mu}, \quad i \in [1, I], \; k \in [1, L]$$

- **Why incremental PGD (iPGD)?**
  - **At each time-step, compresses flow fields on the fly, by enriching the previously computed modes.**
  - **How? Min. approximation error at each time-step of the flow solver.**

weight factor — **Previous approx. to $f_{i,k}$**

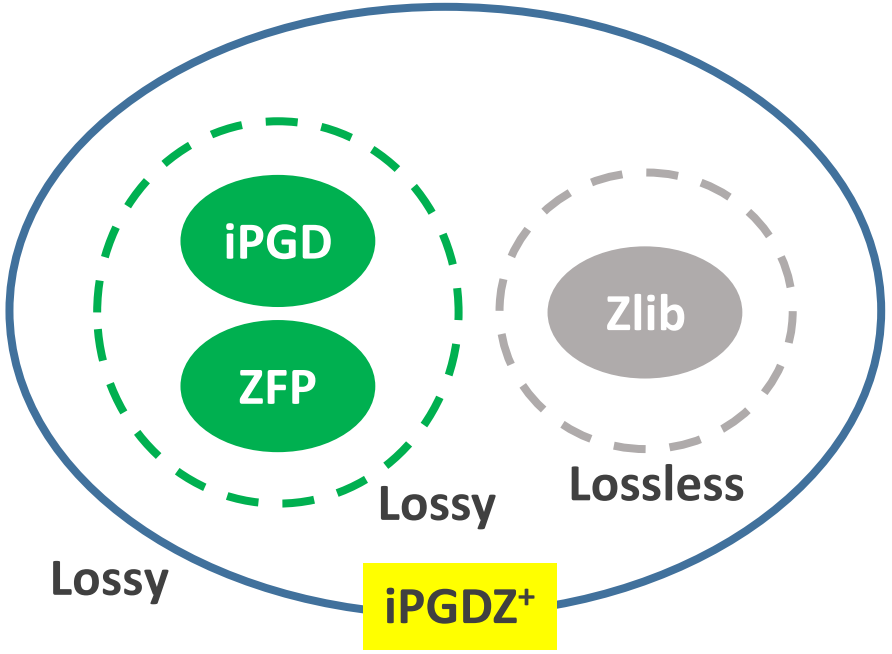
$$E_m = \frac{1}{2} \sum_{i=1}^{I} \left[\sum_{\mu=1}^{m} X_i^{\mu} T_{L+1}^{\mu} - f_{i,L+1} \right]^2 + \frac{w}{2} \sum_{i=1}^{I} \sum_{k=1}^{L} \left[\sum_{\mu=1}^{m} X_i^{\mu} T_k^{\mu} - f_{i,k}^{iPGD} \right]^2$$

Approximation error of current time-step $(L+1)$ — ***Cumulative error of all previous time-steps***

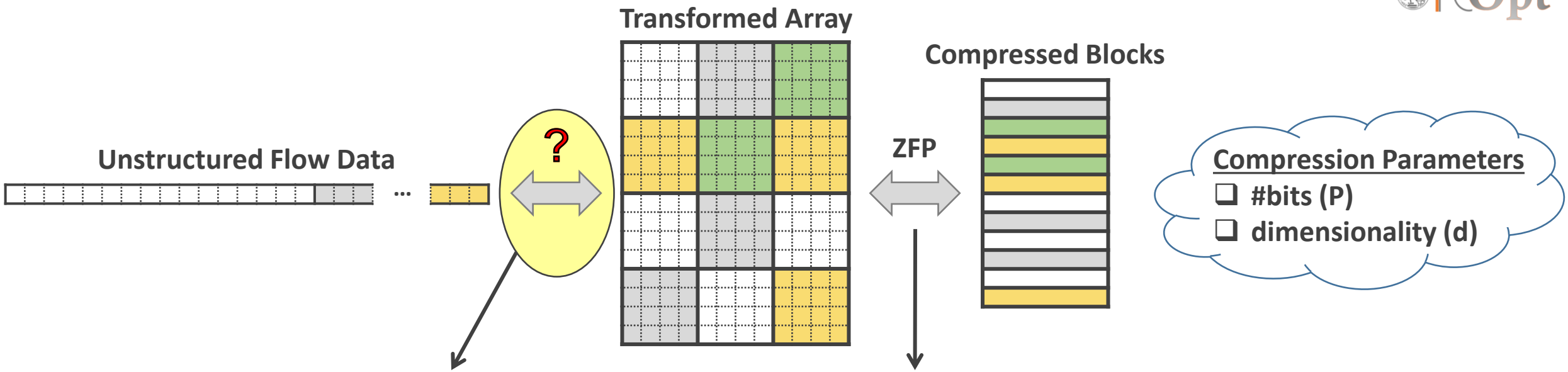
Modes X_i^m, T_k^m and T_{L+1}^m computed by setting:

$$\frac{\partial E_m}{\partial X_i^m} = \frac{\partial E_m}{\partial T_k^m} = \frac{\partial E_m}{\partial T_{L+1}^m} = 0$$

Our Compression Kernel: The iPGDZ⁺ Algorithm at a Glance



Compression Techniques: The ZFP Algorithm



ZFP variants for unstructured flow data

- ***ZFP-raw data***: Retains unstructured grid elements' ordering.
- ***ZFP-adjacency***: Fills blocks with values at adjacent grid cells.
- ***ZFP-sorting***: Value-based rearrangement.

ZFP:

- Integer/Floating-point data stored in d-dimensional arrays, d $\in [1, 4]$.
- Each block of 4^d values compressed independently.
- ***Fixed-precision*** mode: Bounds relative error using P bits.

Two Implementations of iPGDZ⁺: CFS & 3CP

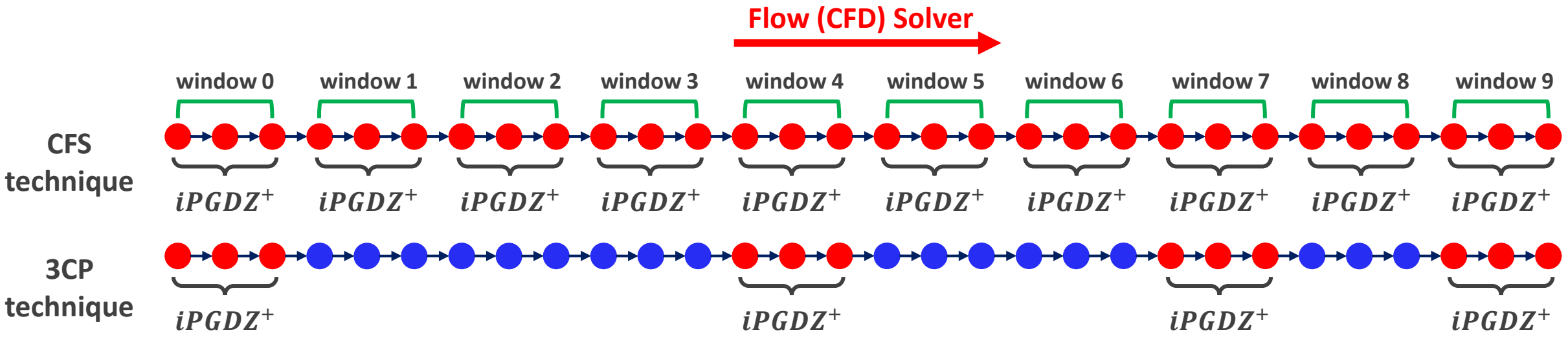
- User-defined compression parameters:
  - Number of iPGD modes.
  - Number of time-steps per time-window.
  - ZFP, Zlib set-up.
  - Number of check-windows (for 3CP).

Compressed Coarse-Grained Check-Pointing (3CP)

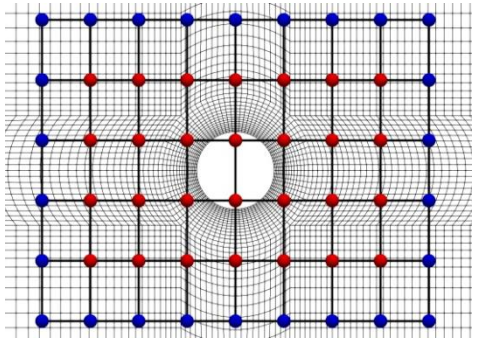
Combines (binomial) check-pointing & iPGDZ⁺.

- Selects check-windows among time-windows.
- Compresses flow solution within each check-window.

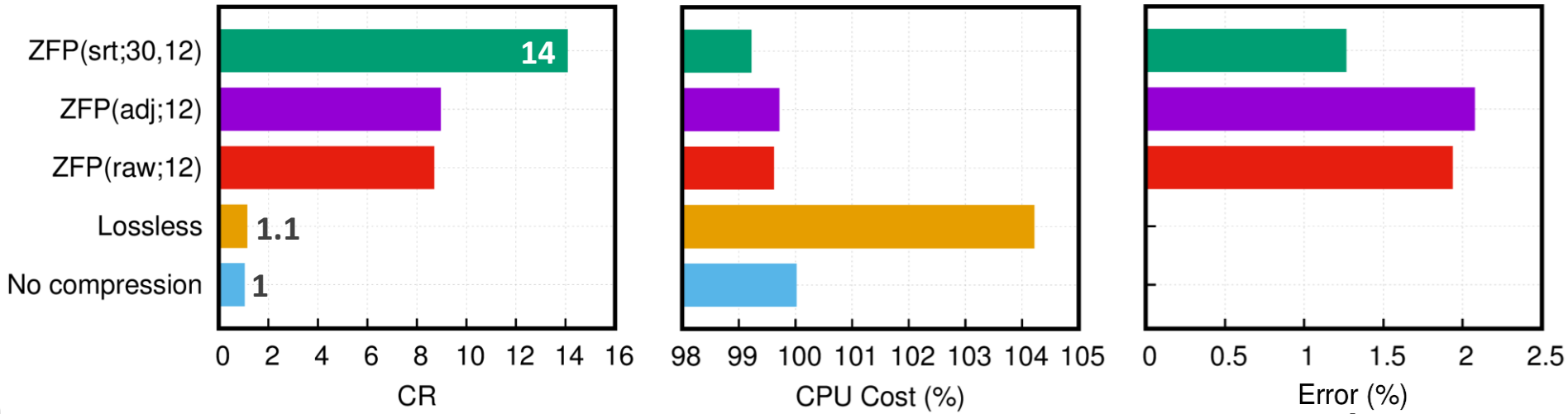
CFS	Min. CPU Cost
3CP	Min. Storage Requirements



2D Cylinder: Lossy or Lossless compression? The ZFP Algorithm (1/3)



Vol. B-Splines lattice



Target: Min. drag, Re = 90

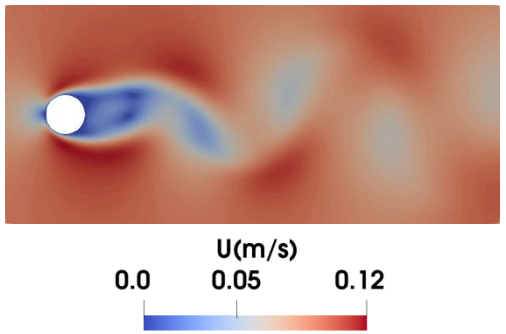
# Cells	13K
# Time-steps	4.8K
Flow Period T	0.68sec
Strouhal	0.163
T_s	7T = 4.76sec
Opt. Method	SQP
# Cores	4

$$\text{Compr. Ratio} = \frac{\text{uncompressed size}}{\text{compressed size}}$$

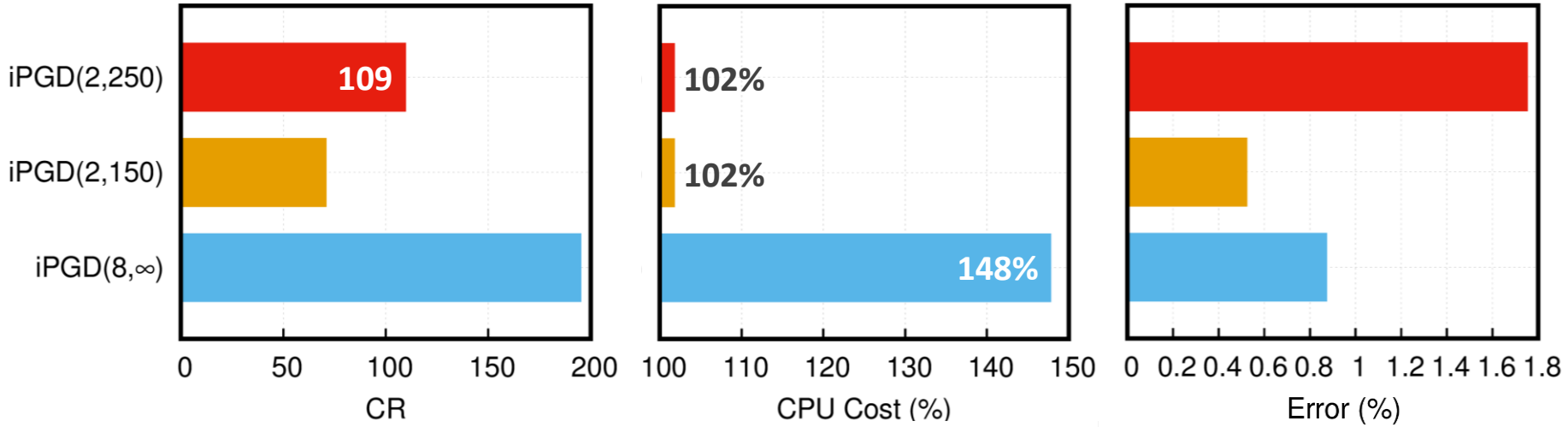
% Error in SDs

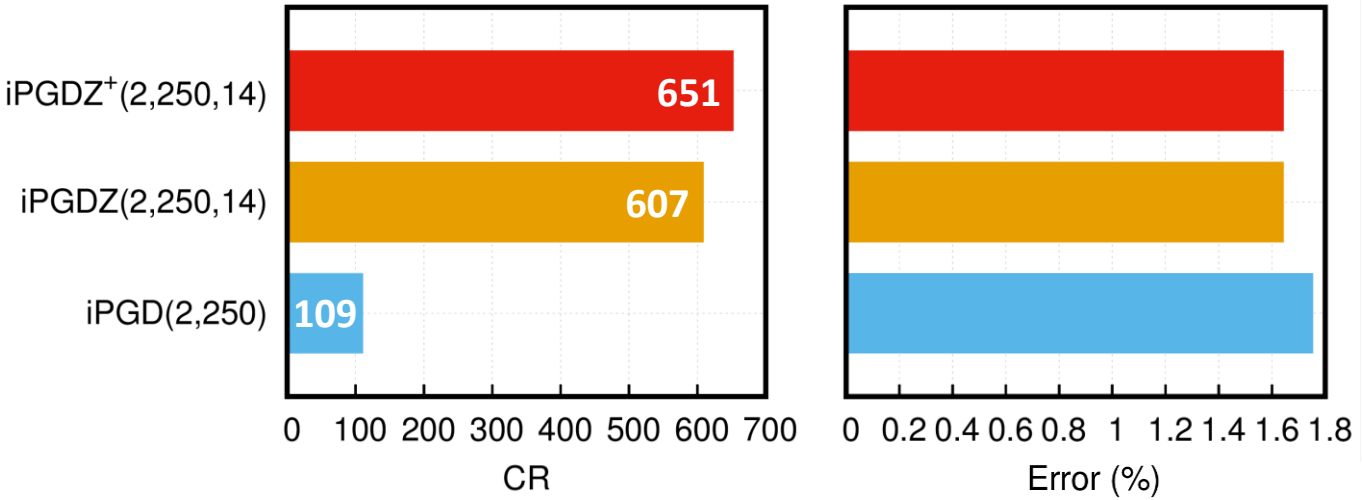
Comments:

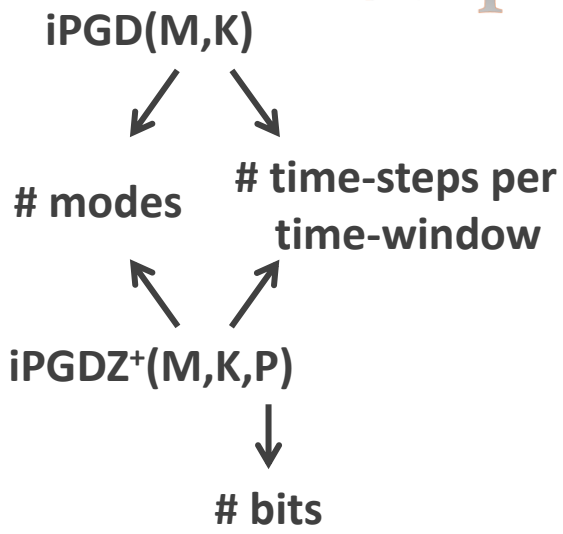
- Lossless compression ineffective.
- ZFP-sorting outperforms ZFP-raw data & ZFP-adjacency.
- Max. $CR = 14$, without compromising SDs.



2D Cylinder: From iPGD to iPGDZ⁺ (2/3)



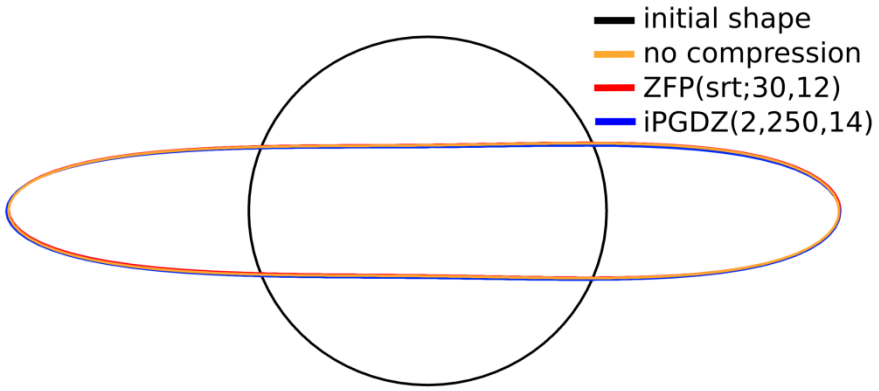




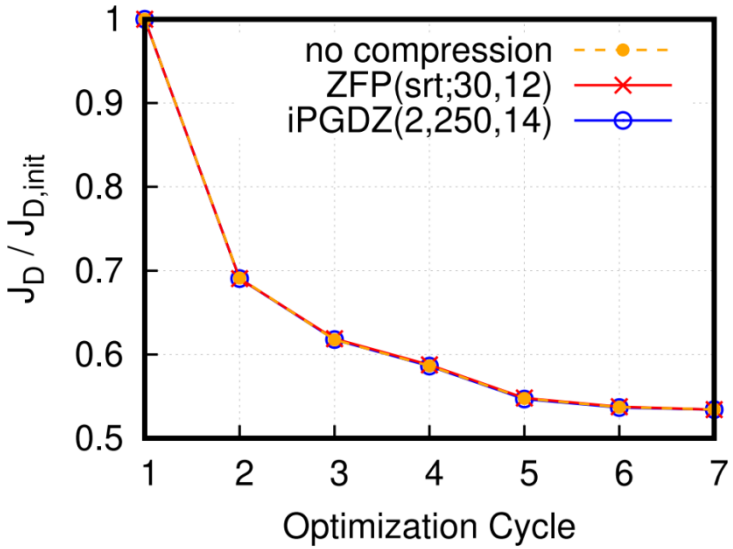
Comments:

- ❑ **Time-windows crucial for iPGD to decrease CPU overhead from 48% to just 2%**
- ❑ **iPGDZ⁺ > iPGDZ > iPGD > ZFP**
- ❑ **Max. CR ≈ 650, without compromising SDs**

2D Cylinder: ShpO Results (3/3)



Starting & Optimised Shapes



$J_D \downarrow 47\%$

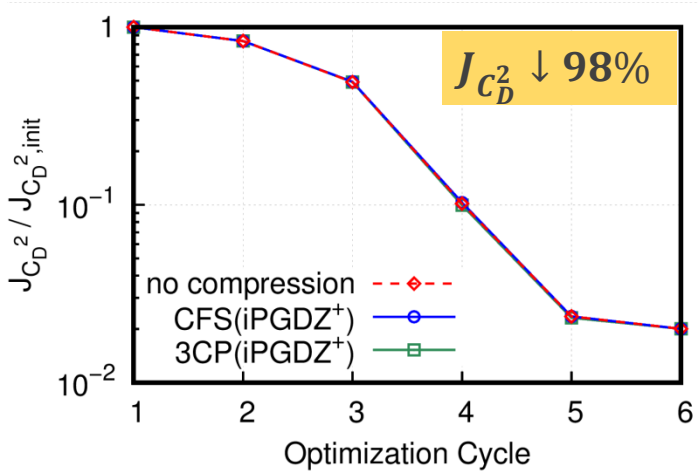
Comments:

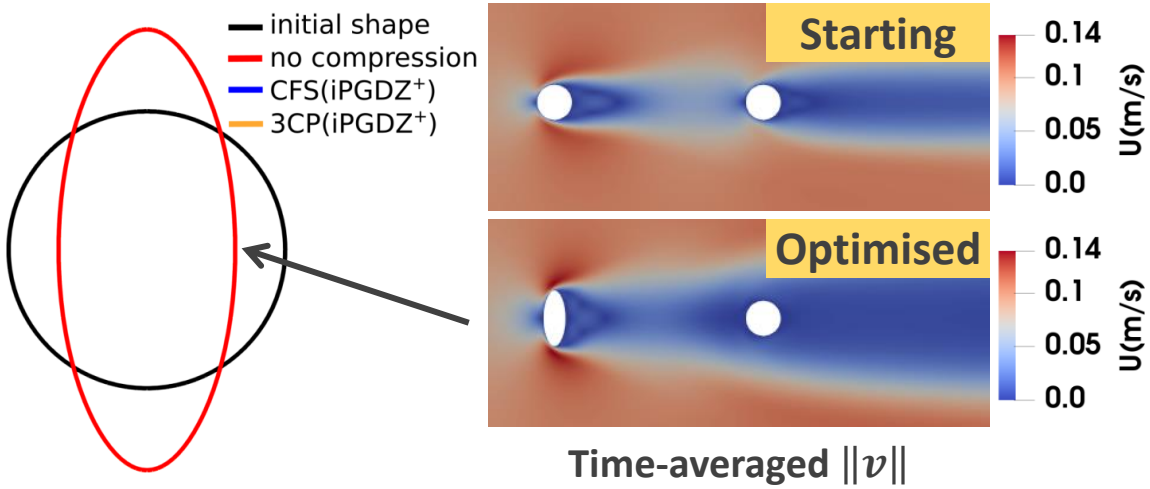
- **Lossy compression does not alter the evolution of J_D throughout the ShpO.**
- **The optimised solution is not affected!**
- **iPGDZ⁺ is the most effective compression algorithm => Main compression kernel.**

Tandem Cylinders: Squared Drag Minimisation at Re = 100

# Cells	76K
# Time-steps	65K
Flow Period T	0.63sec
Strouhal	0.159
T_s	13sec
Opt. Method	Constraint projection
# Cores	12

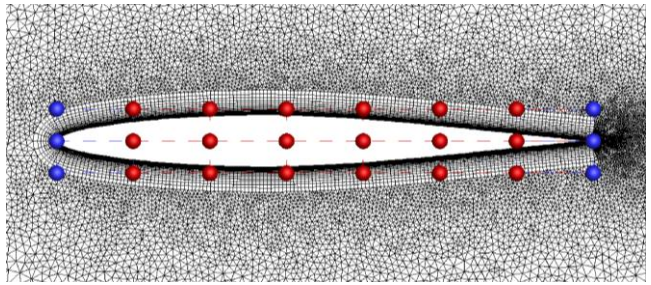
Flow Fields' Storage Strategy	CPU cost/cycle	Compr. Ratio	Error in ∇J
full storage	100%	1	-
CFS(iPGDZ$^+$)	110%	615	0.5%
3CP(iPGDZ$^+$)	143%	3750	0.2%



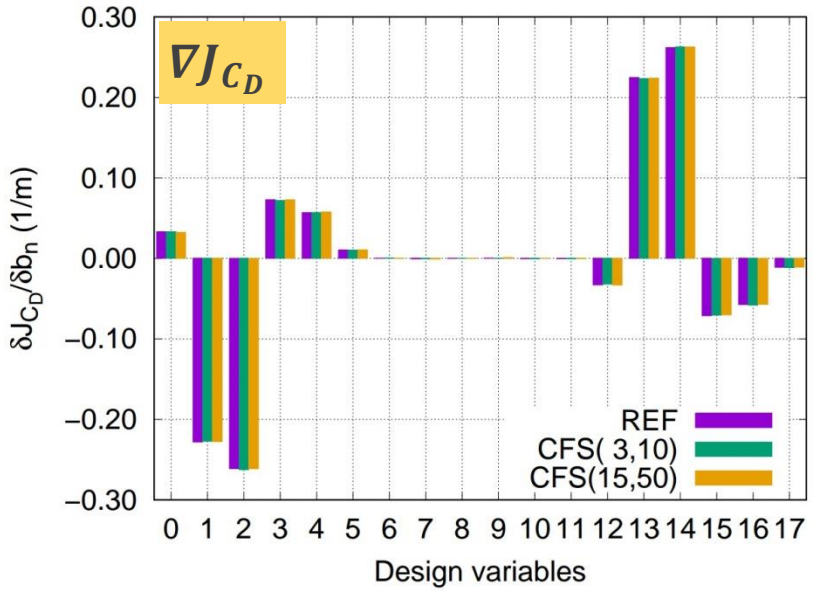


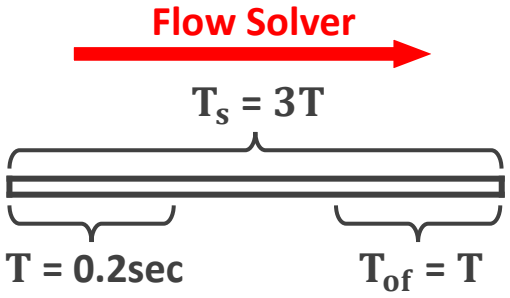
❑ Evolution of $J_{C_D^2}$ & optimised shape not affected by CFS & 3CP.

Compressible Flow: Pitching NACA64A010 Airfoil



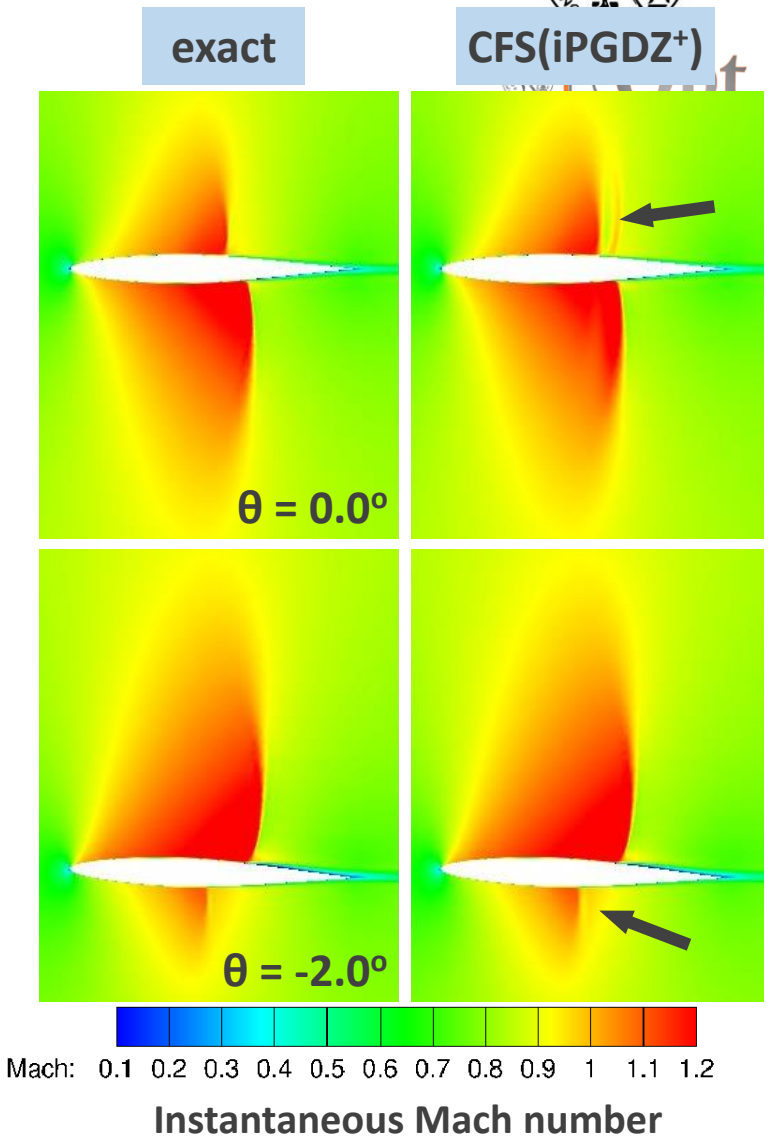
Control Lattice & Grid





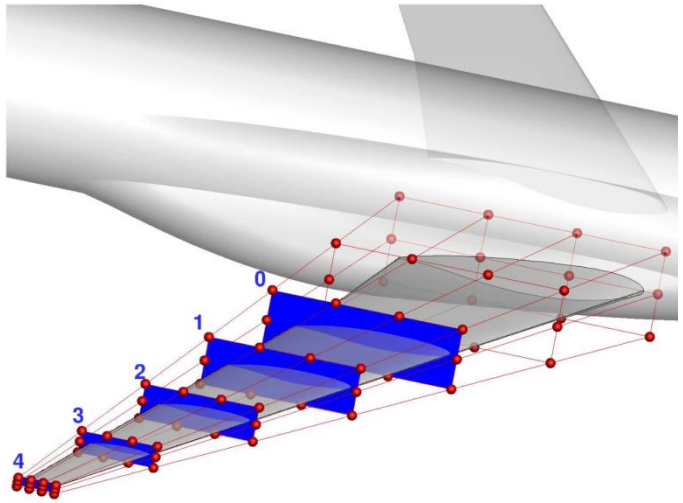
Comments on CFS:

- ❑ Shock wave accurately captured.
- ❑ SDs practically unaffected by lossy compression.
- ❑ $CR \approx 20$, from 1GB to 50MB.



Instantaneous Mach number

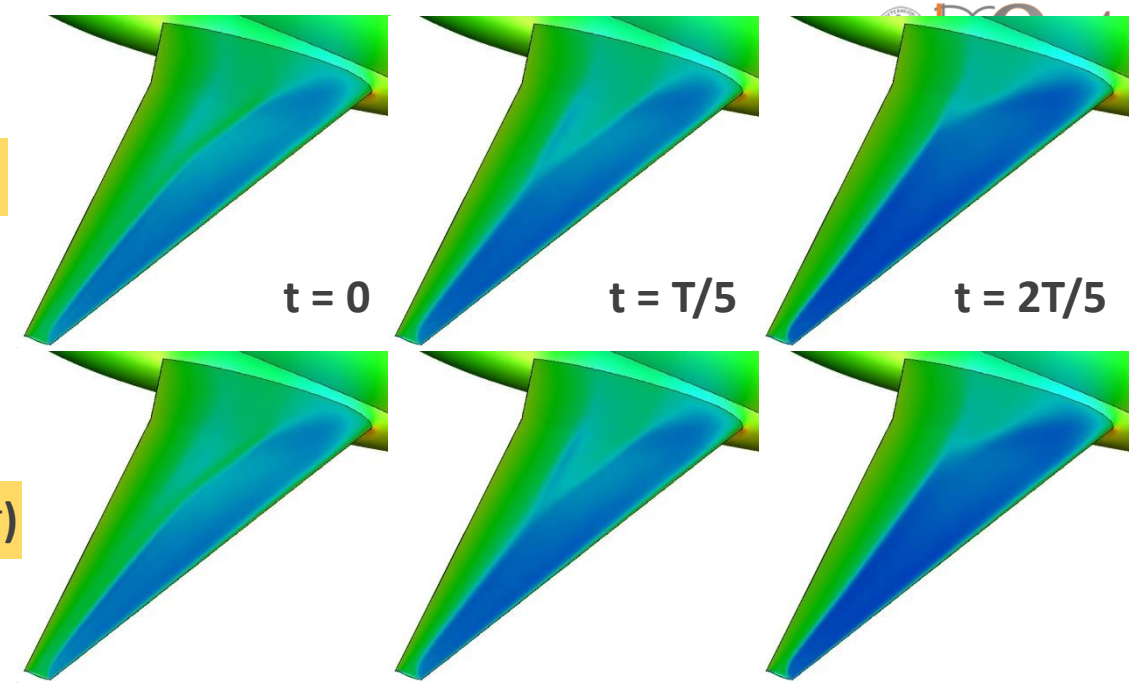
Compressible Flow: Wing of DLR-F25 Aircraft (1/2)



Lattice controling the twist of 5 wing sections

exact

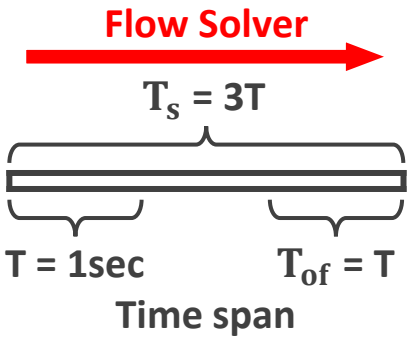
CFS(iPGDZ⁺)



Instantaneous C_p

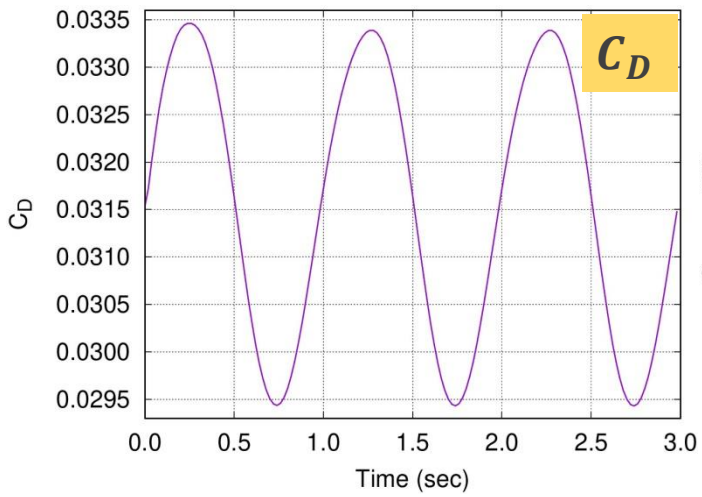
Setup:

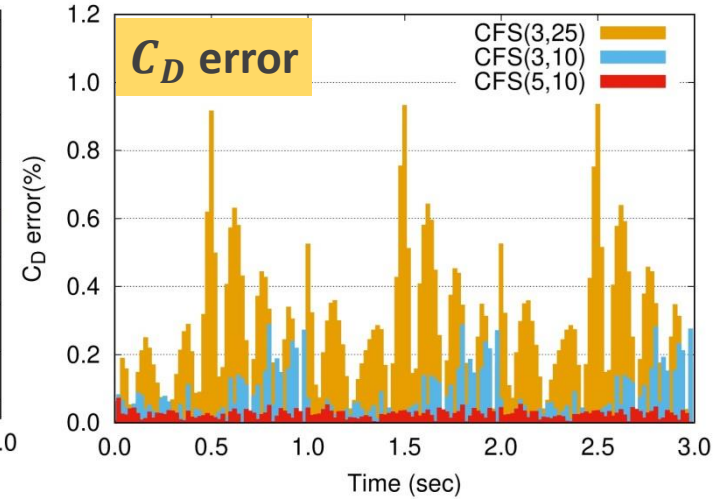
- ❑ Grid: 4M nodes.
- ❑ 50 time-steps/period.
- ❑ Mach = 0.78, Re = 21×10⁶,
  AoA: $\alpha(t) = 1.5^o + 0.5^o \sin(2\pi t)$.

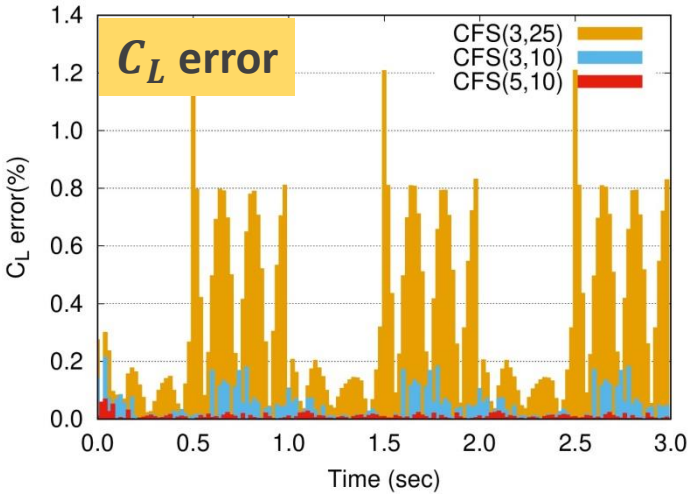


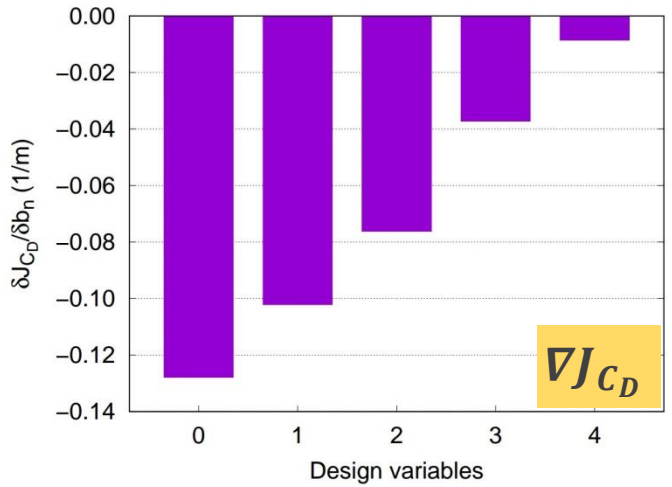
- ❑ C_p based on the reconstructed fields in perfect agreement with reference values (CFD solution).

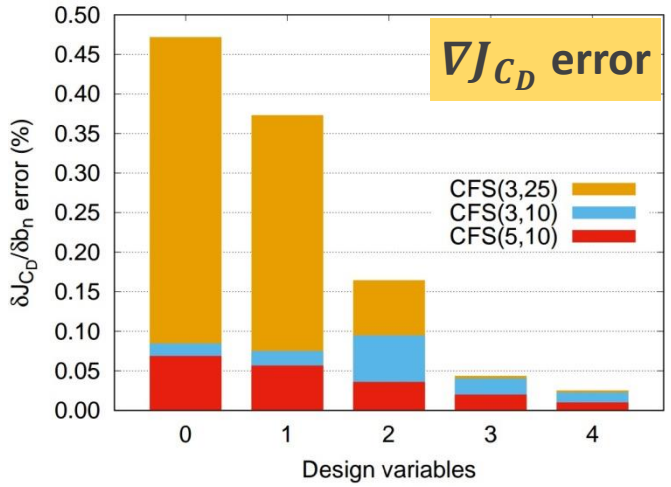
Compressible Flow: Wing of DLR-F25 Aircraft (2/2)









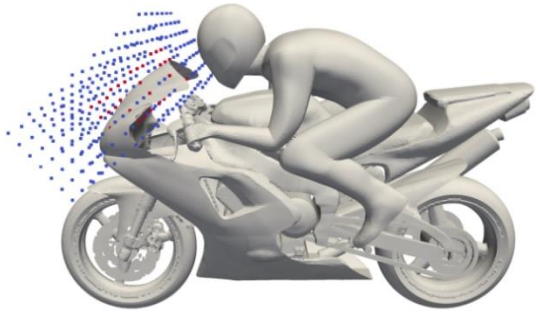


Comments on CFS:

- ❑ Error in reconstructed C_D, C_L < 1.2%.
- ❑ Error in SDs < 0.5%.
- ❑ $CR \approx 60$, from 32GB to 0.5GB.

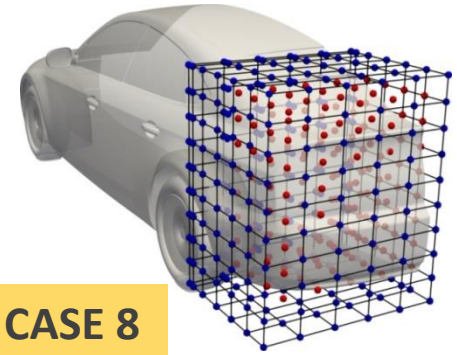
URANS Applications: Drag Minimisation of Motorbike, DrivAer, ID.3 & CHR (1/7)

	Case 7: Motorbike	Case 8: DrivAer	Case 9: ID.3	Case 11: CHR	
				half-car	full-car
# Cells	1.1M	5.3M	16.6M	37.3M	74.7M
# Time-steps	28K	40K	10K	28K	
Δt(sec)	$2.5 \cdot 10^{-4}$	$6 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	
Opt. Method	CG	SQP	CG	SQP	
Flow Model	URANS Spalart-Allmaras				
Reynolds	$3 \cdot 10^{6}$	$1 \cdot 10^{7}$	$9 \cdot 10^{6}$	$9 \cdot 10^{6}$	
# Cores	32	132		960	

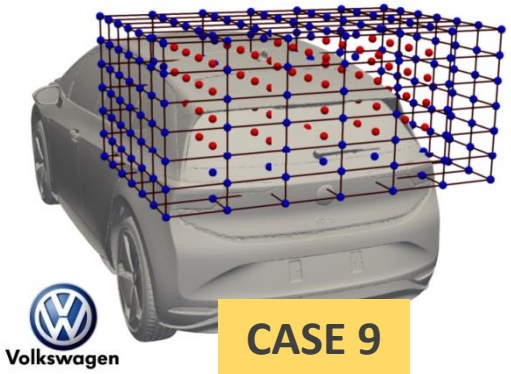


CASE 7

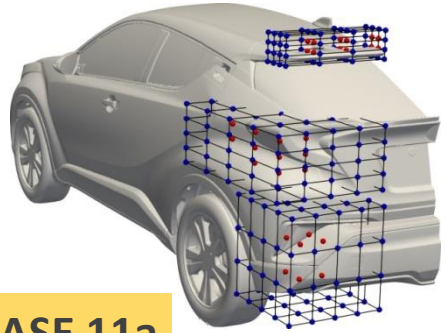
B2 Industrial Benchmark of exaFOAM:
DaRUS, V1, 10.18419/darus-3714, 2024.



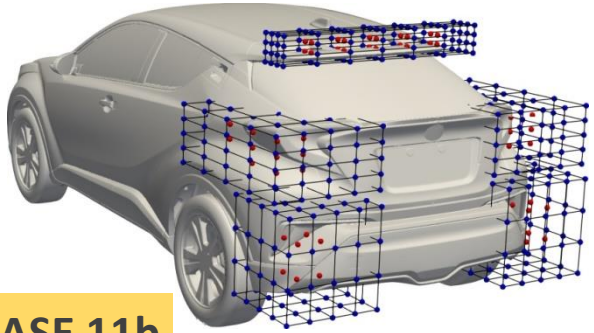
CASE 8



CASE 9



CASE 11a

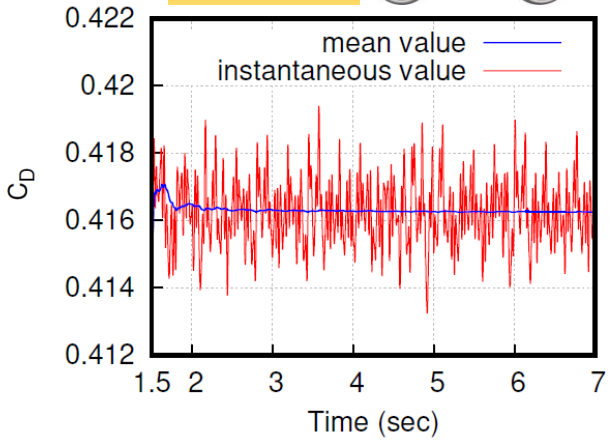


CASE 11b

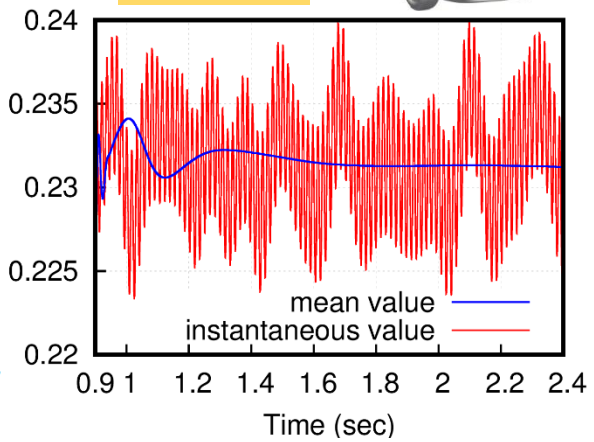
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

URANS Applications: Instantaneous & Mean C_D on Baseline Shapes (2/7)

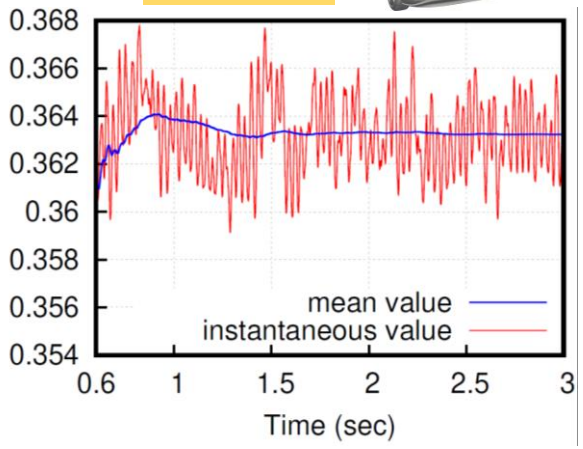
CASE 7

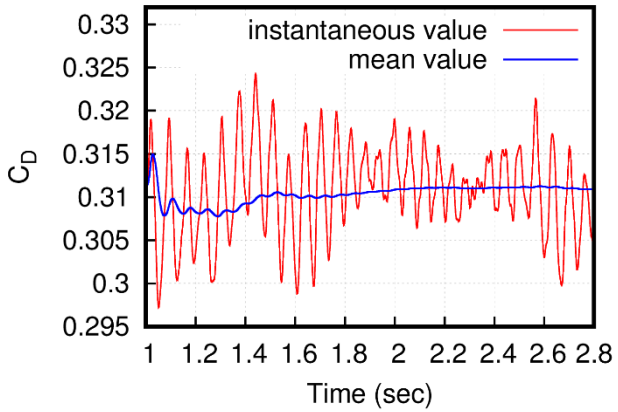


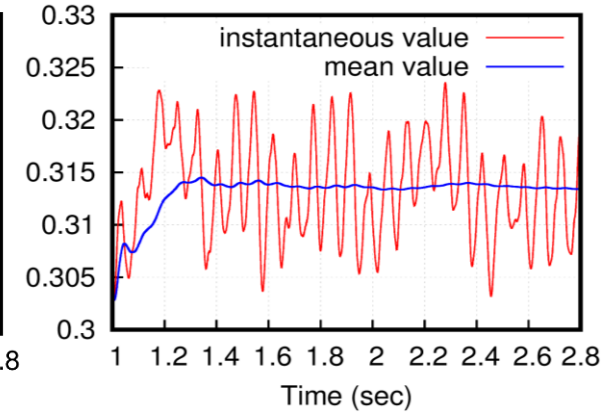
CASE 8



CASE 9





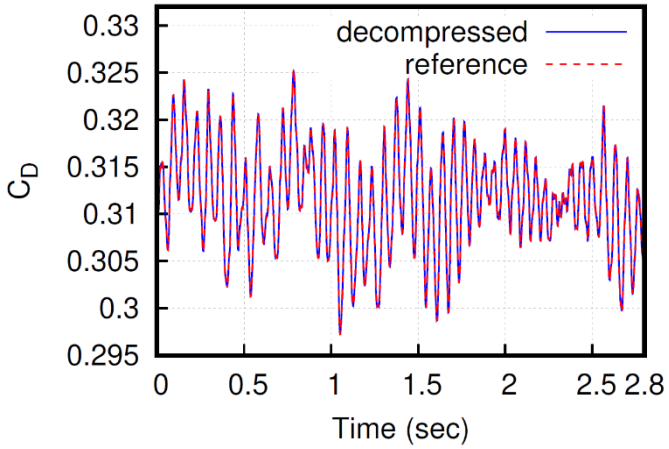


CASE 11a

CASE 11b

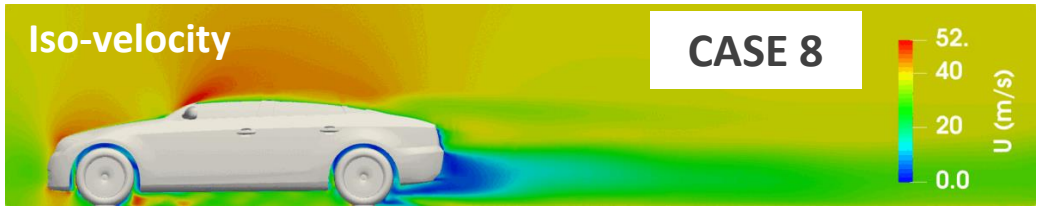
- **Fluctuations in inst. C_D justifies the adoption of an unsteady solver.**
- **Mean C_D converged in the time-span.**

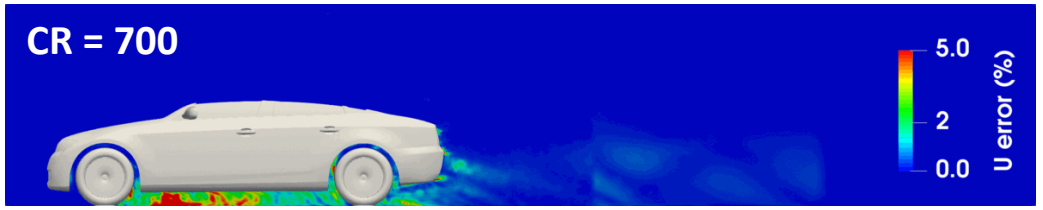
- **CFS: Instantaneous C_D using exact and decompressed fields practically coincide.**

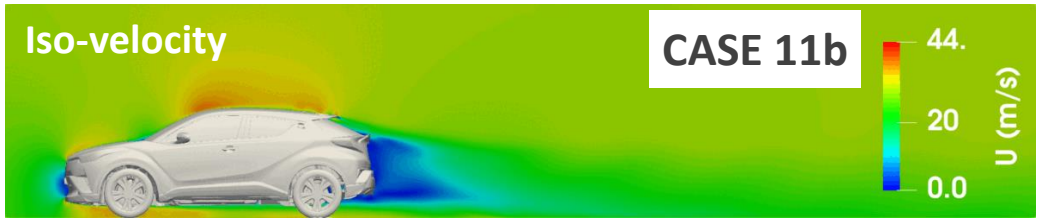


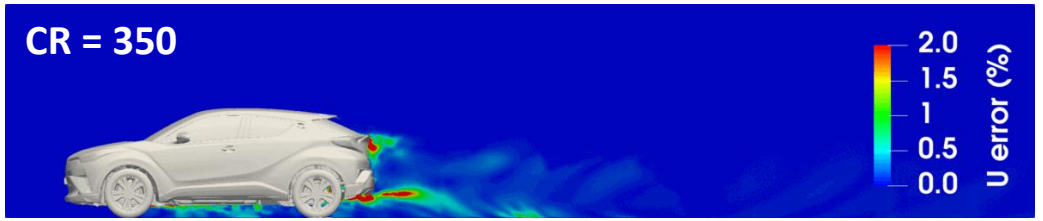
CASE 11a

URANS Applications: Effect of Compression on Velocity Field (3/7)





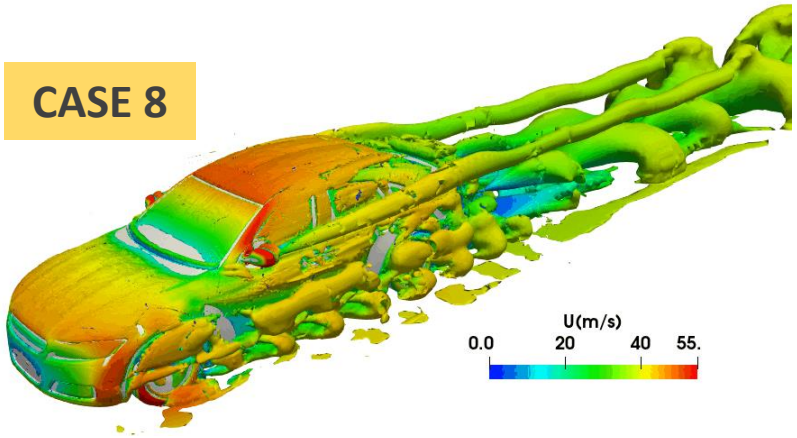


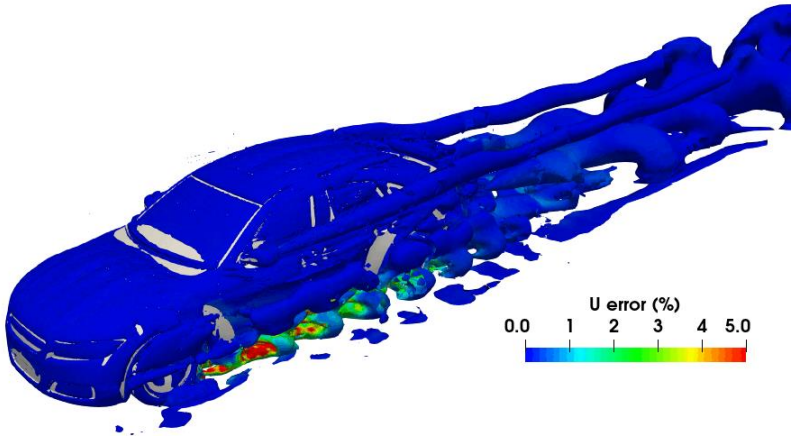


v magnitude

Relative error of v after decompression by CFS

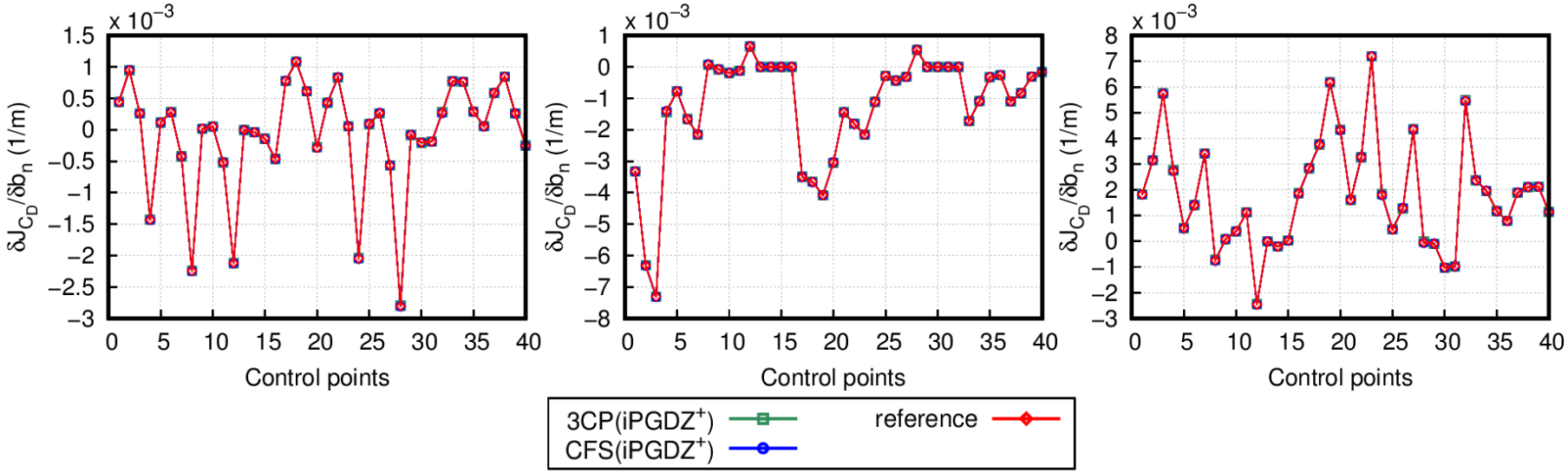
CASE 8

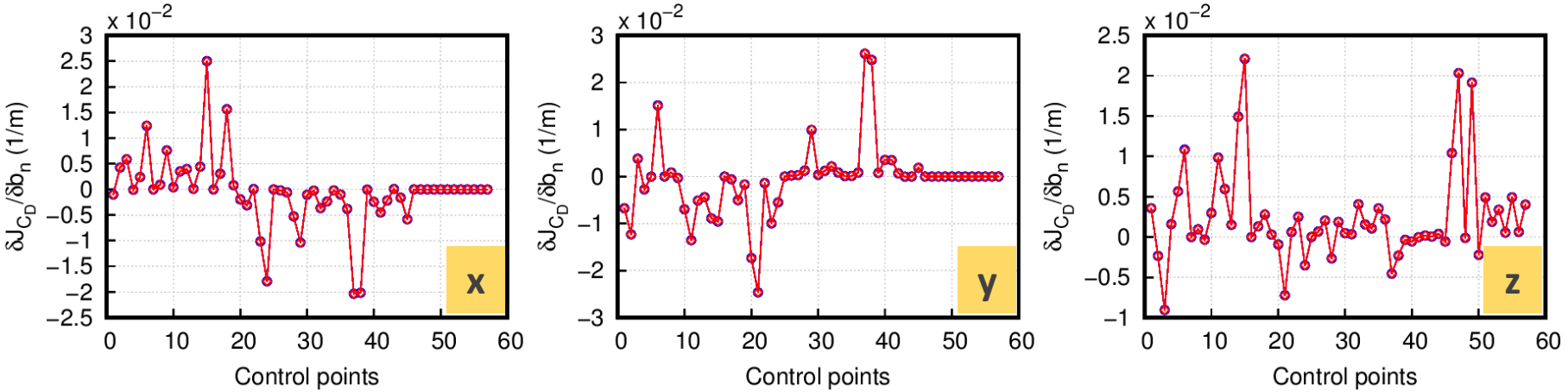




Max. deviation in v, after compression/decompression by CFS, does not exceed 5%, for the selected compression settings.

URANS Applications: Sensitivity Derivatives (4/7)





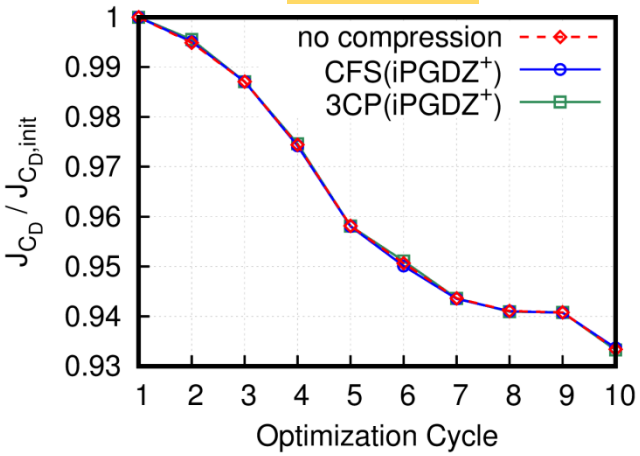
CASE 11a

- **CFS & 3CP do not compromise SDs' accuracy, with error <0.6% in all cases!**

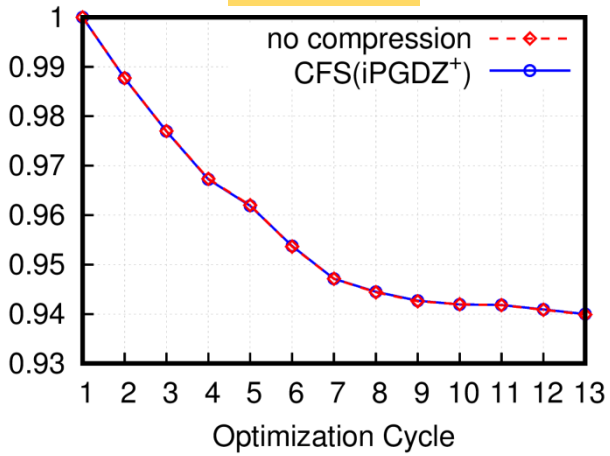
SDs w.r.t. to the x, y and z coordinates of the control points

URANS Applications: Optimisation Results (5/7)

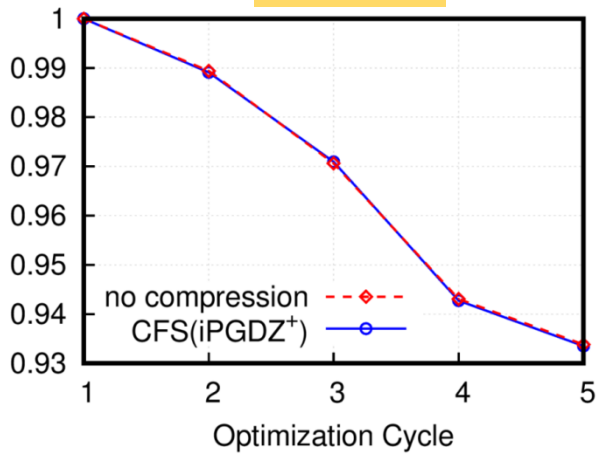
CASE 7



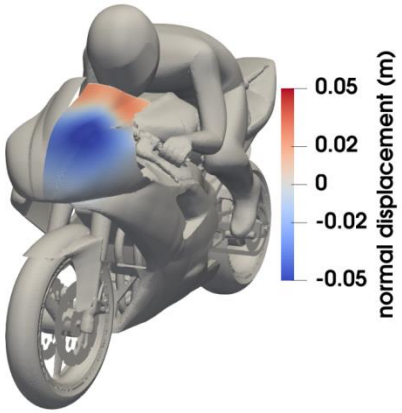
CASE 8



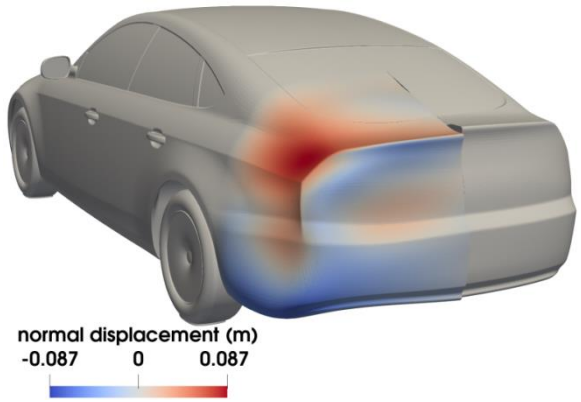
CASE 9



$J_{C_D} \downarrow 6.7\%$



$J_{C_D} \downarrow 6.0\%$



$J_{C_D} \downarrow 6.7\%$

- **CFS & 3CP successfully support unsteady adjoint and do not alter the evolution of J_{C_D} throughout the optimisation.**
- **The optimised solution is not affected either!**

Shape	Side
Baseline	Starboard
Optimised	Port

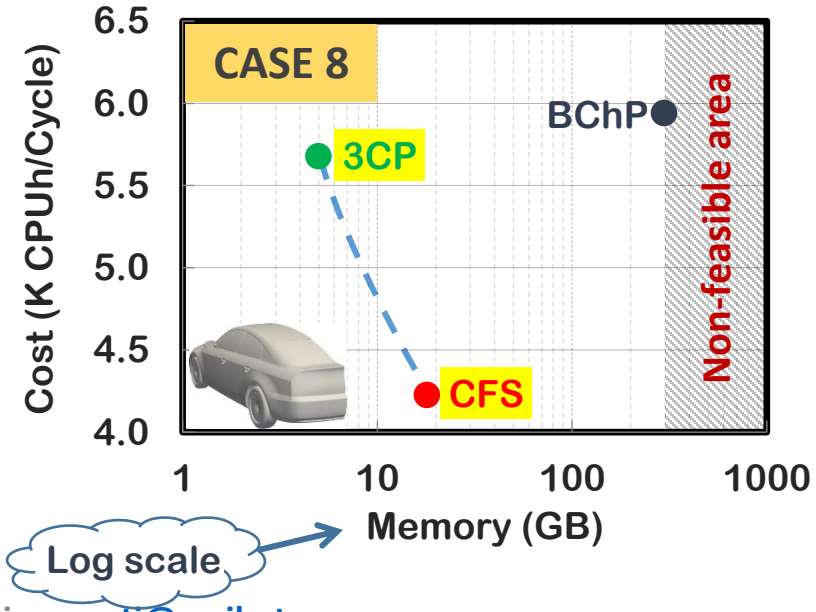
URANS Applications: CPU cost & Memory Saving (6/7)

Case	Flow Fields' Storage Strategy	CPU cost/cycle		Compr. Ratio	Mem (GB)
		CPUh	%		
Motorbike	BChP(882)	1.6K	100%	32	64*
	CFS(iPGDZ+)	1.2K	73%	452	4.5
	3CP(iPGDZ+)	1.5K	94%	1944	1.0
DrivAer	BChP(943)	6.0K	100%	42	298*
	CFS(iPGDZ+)	4.2K	72%	702	18
	3CP(iPGDZ+)	5.7K	95%	2508	5.0
ID.3	BChP(280)	7.8K	100%	36	298*
	CFS(iPGDZ+)	5.4K	69%	1221	8.7
CHR half-car	BChP(900)	22.8K	100%	31	1930*
	CFS(iPGDZ+)	16.9K	74%	350	171
	3CP(iPGDZ+)	22.2K	97%	1849	32.4
CHR full-car	BChP(450)	53.1K	100%	62	1930*
	CFS(iPGDZ+)	38.6K	73%	345	348
	3CP(iPGDZ+)	51.4K	97%	1825	65.6

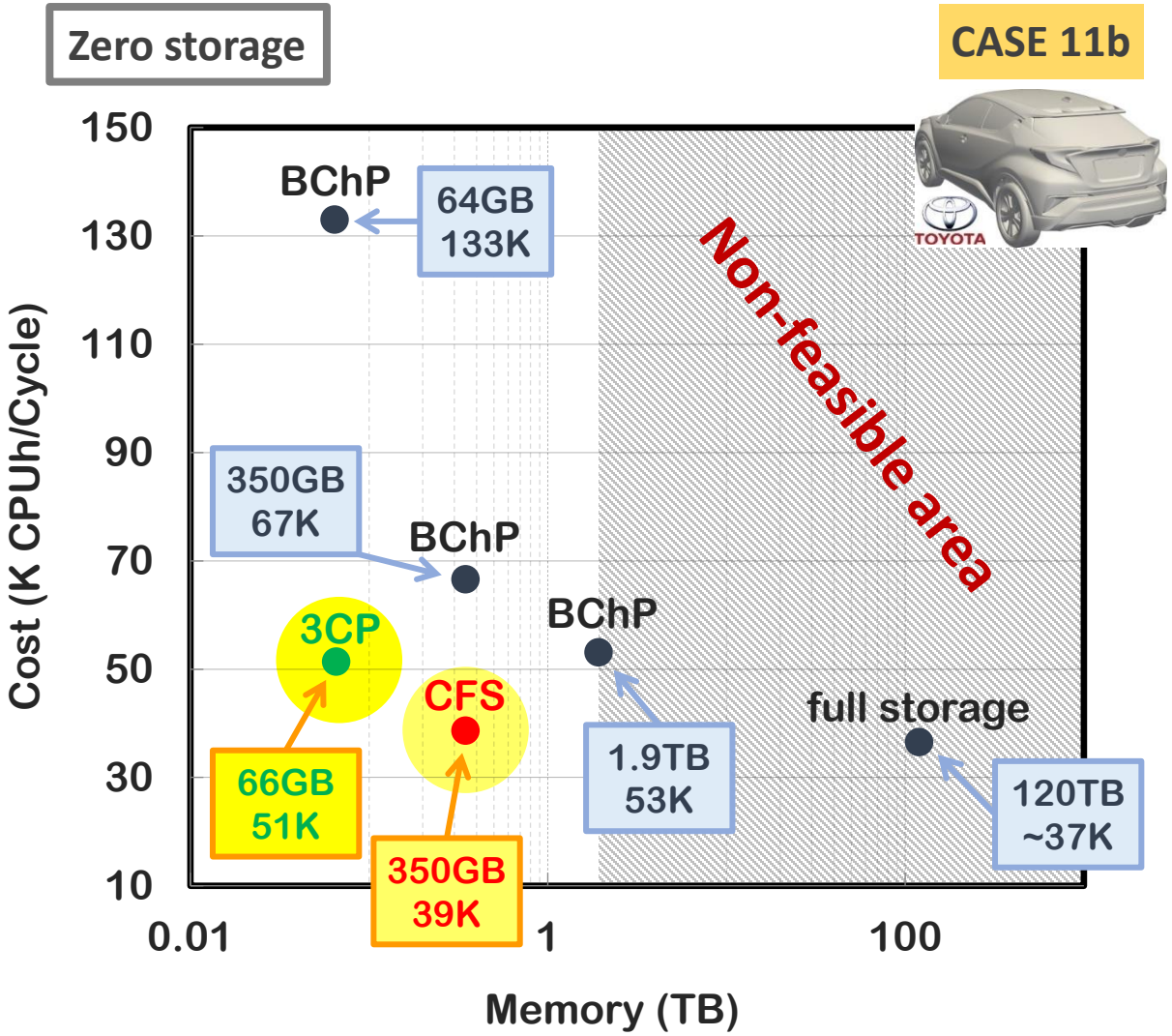
All cases	CFS vs. BChP	3CP vs. BChP
Memory	÷ 6 … 35	÷ 21 … 60
CPU cost	−30%	−4%

All cases	CFS vs. full storage	3CP vs. full storage
Memory	÷ 350 … 1250	÷ 1800 … 2500

Summary results across all cases



URANS Applications: CPU cost & Memory Saving (7/7)



All cases	CFS vs. BChP	3CP vs. BChP
Memory	÷ 6 … 35	÷ 21 … 60
CPU cost	−30%	−4%

All cases	CFS vs. full storage	3CP vs. full storage
Memory	÷ 350 … 1250	÷ 1800 … 2500

Summary results across all cases

CHR full-car

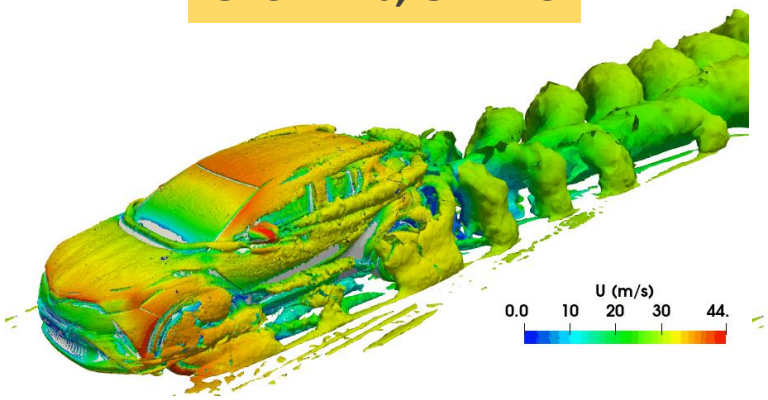
- full storage:
  - memory (RAM): 120TB
- BChP vs. CFS:
  - cost: 53K ⇒ 39K CPUh
  - wall-clock-time: 55h ⇒ 40h
  - memory: 1.9TB ⇒ 350GB
- BChP vs. 3CP:
  - cost: 53K ⇒ 51K CPUh
  - memory: 1.9TB ⇒ 66GB

DDES Applications: TME Concept & CHR Car Models (1/4)

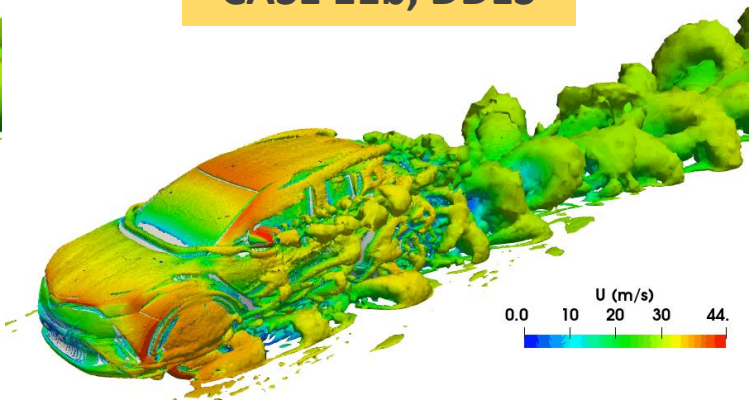
(D)DES is more demanding for CFS & 3CP, due to:

- ❑ **Inherently complex and fluctuating nature of the flow.**
- ❑ **Finer resolved spatial scales.**
- ❑ **Intense temporal gradients.**

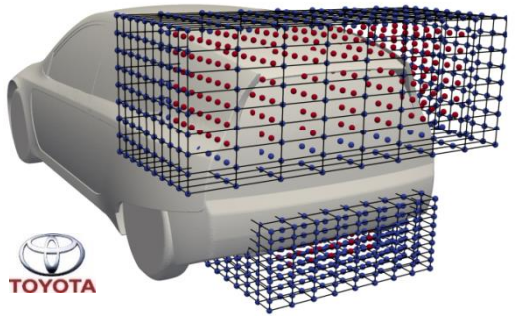
CASE 11b, URANS



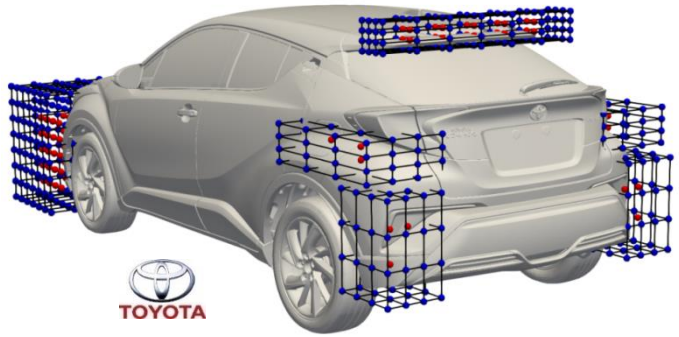
CASE 11b, DDES



	Case 10: Concept Car	Case 11: CHR	
		half-car	full-car
# Cells	29.3M	37.3M	74.7M
# Time-steps	15K	28K	
Δt(sec)	$2.3 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	
Flow Model	DDES Spalart-Allmaras		
Reynolds	$9 \cdot 10^{6}$		
# Cores	440	960	

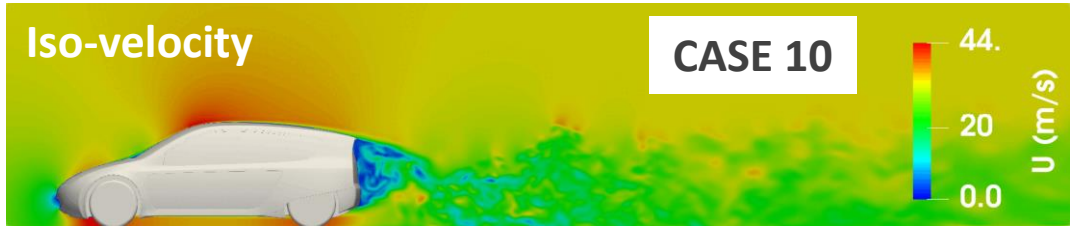


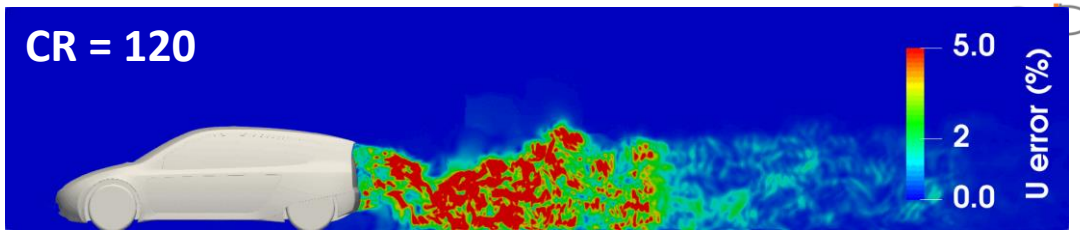
CASE 10

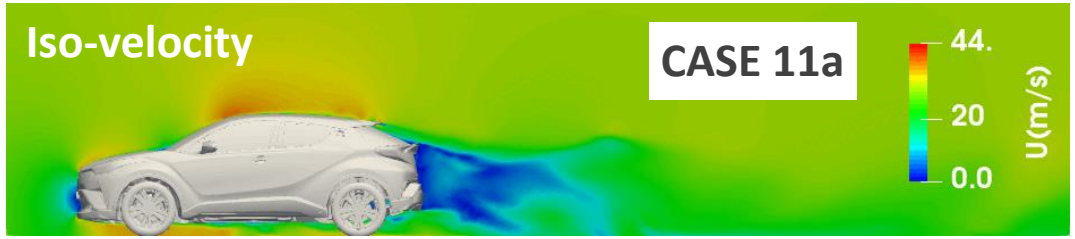


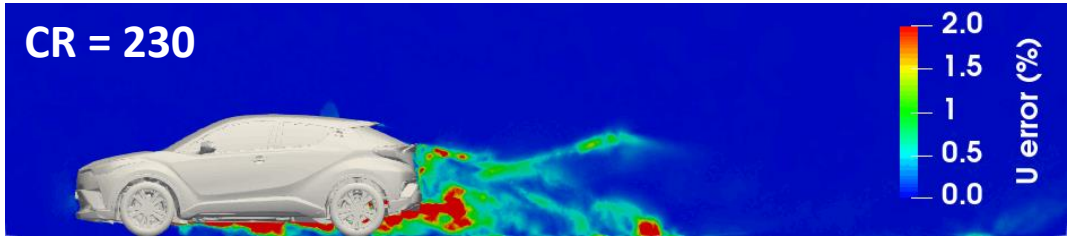
CASE 11b

DDES Applications: Effect of Compression on Velocity Field (2/4)



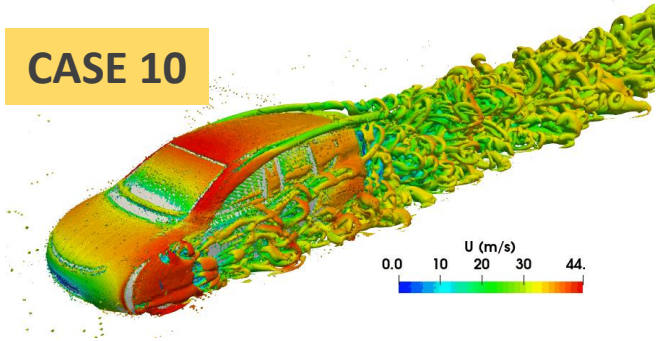


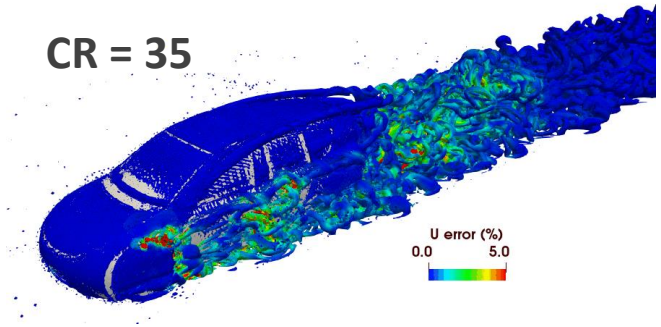




v magnitude

Relative error of v after decompression by CFS

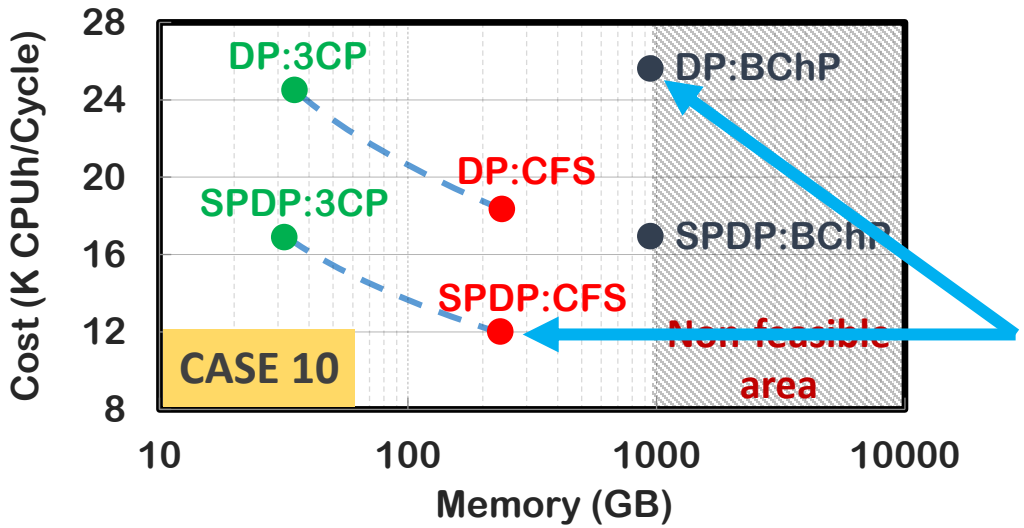




- $\|v\|$ & relative error after compression/decompression using **CFS**.
- Max. error in the wake and close to the underbody.
- SDs practically unaffected by lossy compression, with error < 0.6%!

DDES Applications: CPU cost & Memory Saving (3/4)

Concept Car	Flow Fields' Storage Strategy	CPU cost/cycle CPUh	CPU cost/cycle %	Compr. Ratio	Mem (GB)
Double Precision	BChP(500)	25.6K	100%	30	948*
	CFS(iPGDZ+)	18.3K	72%	121	239
	3CP(iPGDZ+)	24.5K	96%	827	35
Mixed Precision	BChP(1000)	16.9K	100%	15	948*
	CFS(iPGDZ+)	12.0K	71%	61	236
	3CP(iPGDZ+)	16.9K	99%	453	32



Mixed precision of OpenFOAM:

- ❑ **Double precision for linear solvers.**
- ❑ **Single precision elsewhere (incl. iPGD).**

Double vs. Mixed Precision:

- ❑ **Small effect on J_{C_D} & ∇J:**

	J_{C_D}	Difference in ∇J
Double Precision	0.2568	-
Mixed Precision	0.2587	2.4%

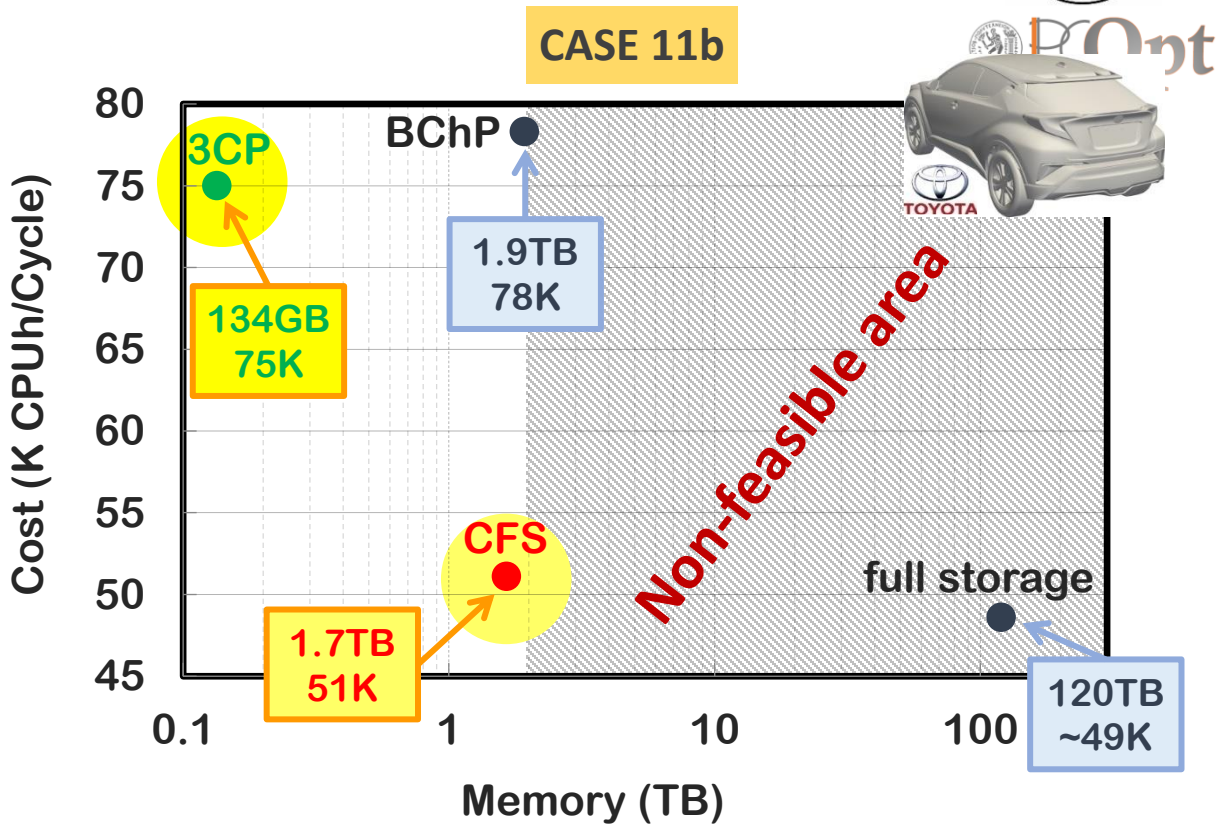
- ❑ **Cost ↓ 34%, for all storage strategies.**
- ❑ **Same memory footprint for CFS, 3CP.**

DP & BChP vs. SPDP & CFS:

- ❑ **Cost ↓ 55%, from 26K to 12K CPUh/cycle.**
- ❑ **Difference in ∇J = 2.2%.**

DDES Applications: CPU cost & Memory Saving (4/4)

CHR	Flow Fields' Storage Strategy	CPU cost/cycle		Compr. Ratio	Mem (GB)
		CPUh	%		
Half-car	BChP(900)	39.1K	100%	31	1930
	CFS(iPGDZ$^+$)	26.4K	68%	231	260
	3CP(iPGDZ$^+$)	38.9K	99%	1230	49
Full-car	BChP(450)	78.3K	100%	62	1930
	CFS(iPGDZ$^+$)	51.1K	65%	73	1650
	3CP(iPGDZ$^+$)	75.0K	96%	892	134



CHR full-car

- **BChP vs. CFS:**
  - cost: 78K ⇒ **51K** CPUh
  - wall-clock-time: 81h ⇒ **53h**
  - memory: 1.9TB ⇒ **1.7TB**
- **BChP vs. 3CP:**
  - cost: 78K ⇒ **75K** CPUh
  - memory: 1.9TB ⇒ **135GB**

Comments:

- Full- vs. half-car
- DDES vs. URANS

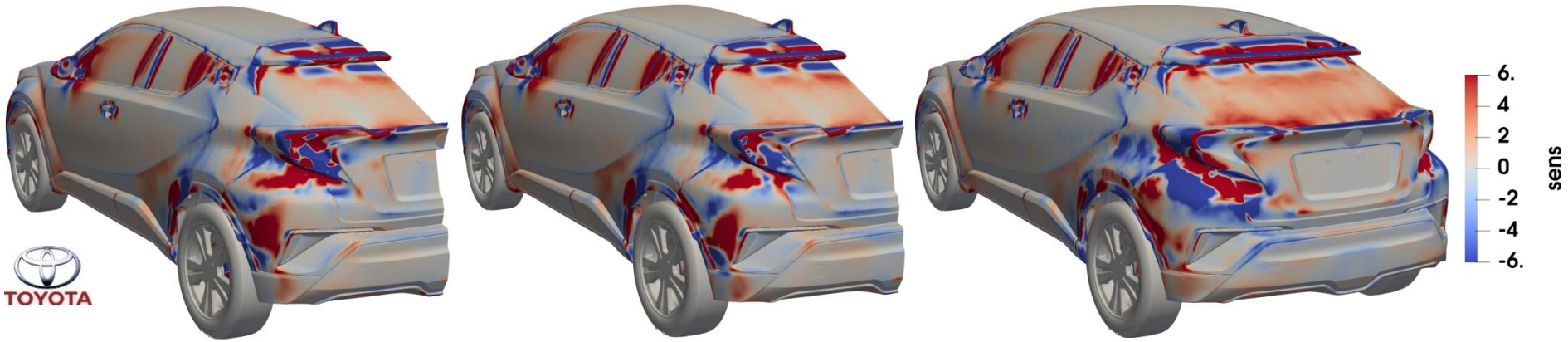
} ↓ compr. ratio for the same error in ∇J.

Compr. Ratio with URANS = **350**

CHR Model: Assessment of Optimisation Strategies (1/2)

Optimisation Strategies	Solver		Geometry	Cost per Cycle (CPUh)
	Flow	Adjoint		
RANS^2	RANS	RANS	half-car	1.0K
URANS^2	URANS	URANS	half-car	27.1K
DDES/RANS	DDES	RANS	full-car	40.5K

→ **assisted by CFS**

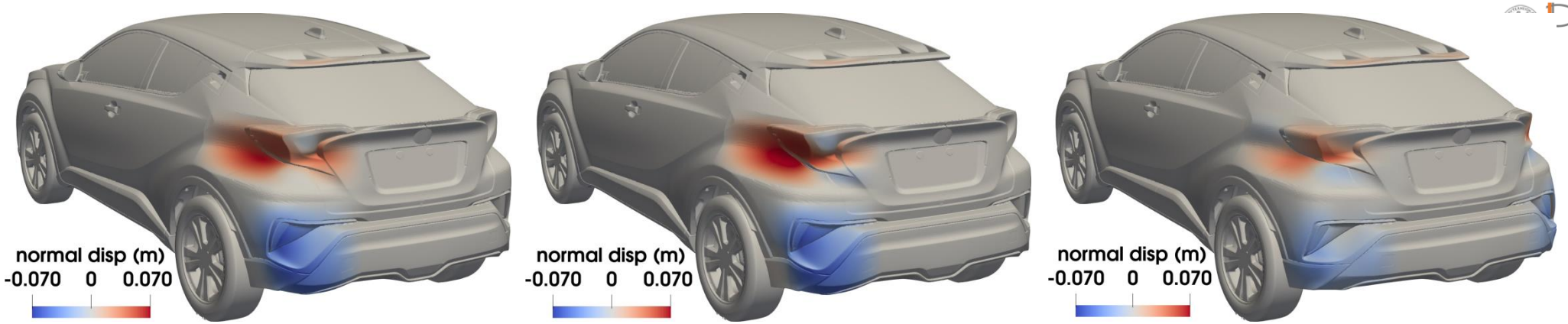


RANS^2 **URANS^2** **DDES/RANS**

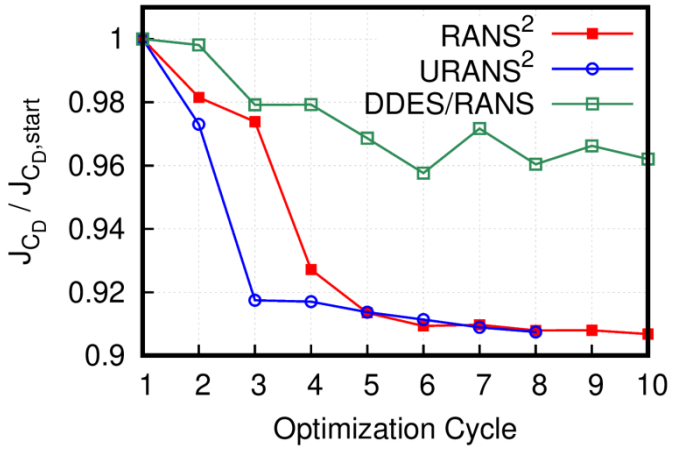
Sensitivity map of J_{C_D}

CHR Model: Assessment of Optimisation Strategies (1/2)

Optimised shapes



RANS²: G1 Shape | **URANS²: G2 Shape** | **DDES/RANS: G3 Shape**

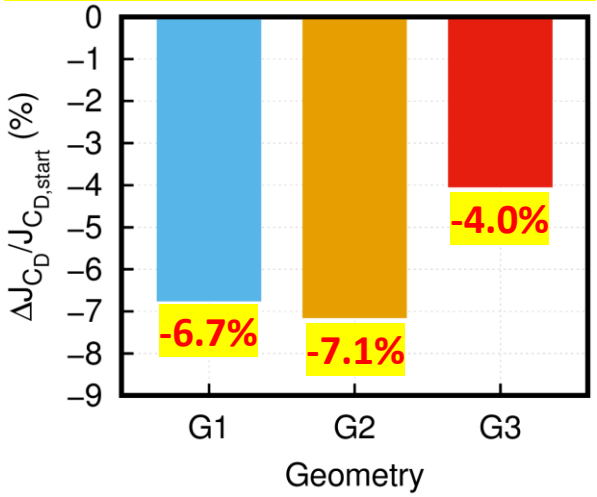


Results correspond to different analysis codes

Conclusions:

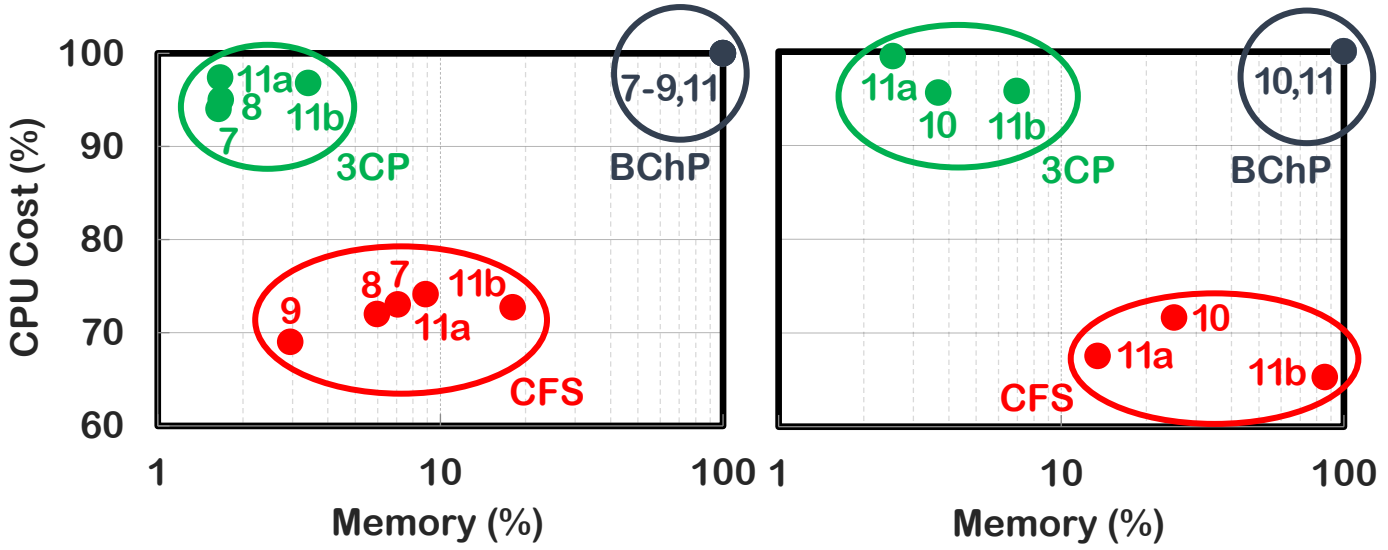
- ❑ **Reduction in J_{C_D} based on DDES full-car re-evaluations: G2 < G1 < G3.**
- ❑ **URANS² achieved the greatest reduction in J_{C_D} (-7.1%), but at a ×27 higher CPU cost compared to RANS².**

DDES full-car re-evaluations



Conclusions

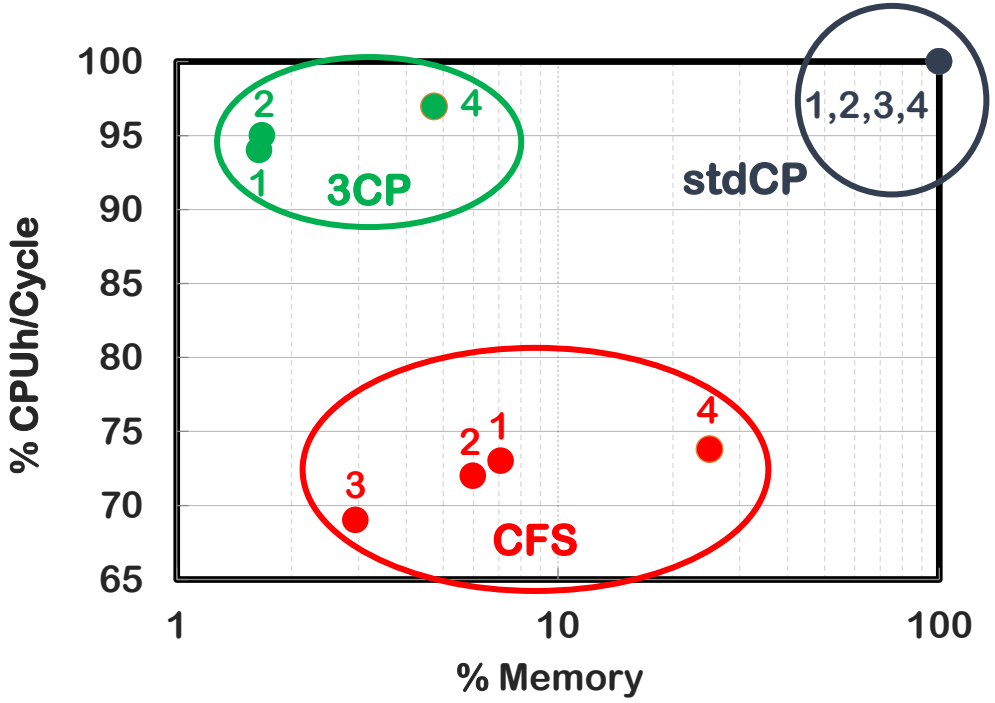
- **Lossy compression techniques:**
  - **Effectiveness: iPGDZ$^+$ > iPGDZ > iPGD > ZFP.**
  - **CFS & 3CP applicable to compressible & incompressible flows, continuous & discrete adjoint. Practically, any multi-dimensional adjoint-based optimisation problem can profit of them.**
  - **3CP is more efficient memory wise, CFS is more effective cost wise. Both outperform binomial check-pointing, in terms of both criteria.**
  - **∇J and evolution of J during the optimisation practically unaffected by lossy compression.**
  - **Enhanced both software developed by our group with efficient lossy compression capabilities, enabling the adjoint optimisation of larger cases at a reasonable cost.**
  - **Indicative automotive application (CHR model): CFS reduced cost/cycle from 78K to 51K CPUh (81h to 53h): Cost for 10 cycles on Amazon Web Services (AWS) reduced from 30K € to < 20K €.**
- **Adjoint to the (D)DES Spalart-Allmaras:**
  - **Differentiation of turbulence model PDE crucial to compute accurate SDs.**
  - **"Frozen turbulence" adjoint prone to inaccuracies, with SDs for some design variables, even exhibiting incorrect signs.**



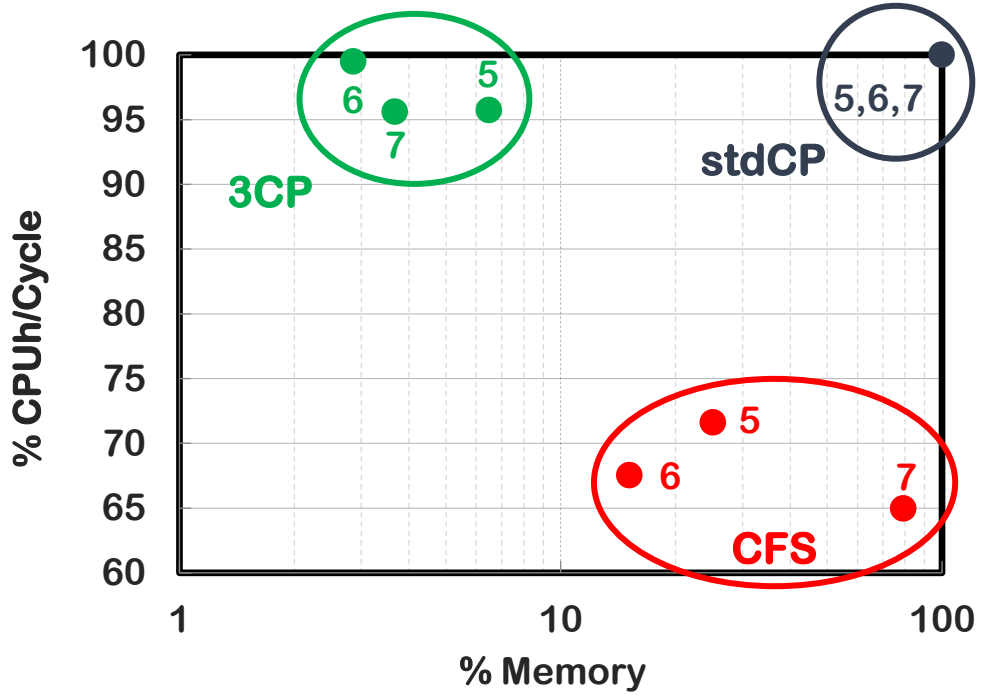
Results from the 5 URANS cases

Results from the 3 DDES cases

Unsteady Adjoint & Compression: CPU cost & Memory Savings of All Cases

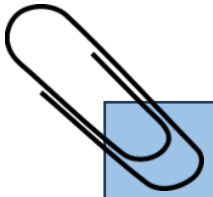


Results from the 4 U-RANS cases

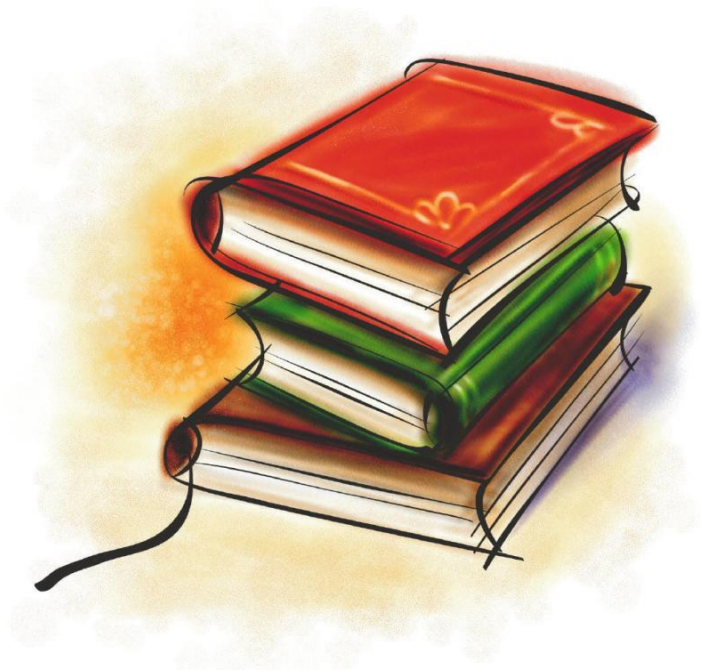


Results from the 3 DDES cases

Read more in:



- A.-S.I. Margetis, “Unsteady RANS & (D)DES Continuous Adjoint and CompressionTechniques in Aerodynamic Shape Optimization”, PhD Thesis, National Technical University of Athens, 2024.
- A.-S.I. Margetis, E.M. Papoutsis-Kiachagias and K.C. Giannakoglou. “Reducing Memory Requirements of Unsteady Adjoint by Synergistically using Check-pointing and Compression”. International Journal for Numerical Methods in Fluids 2023; 95(1):23-43. doi:10.1002/fld.5136.
- A.-S.I. Margetis, E.M. Papoutsis-Kiachagias and K.C. Giannakoglou. “Lossy Compression Techniques Supporting Unsteady Adjoint on 2D/3D Unstructured Grids”. Computer Methods in Applied Mechanics and Engineering 2021; 387:114152. doi:10.1016/j.cma.2021.114152.

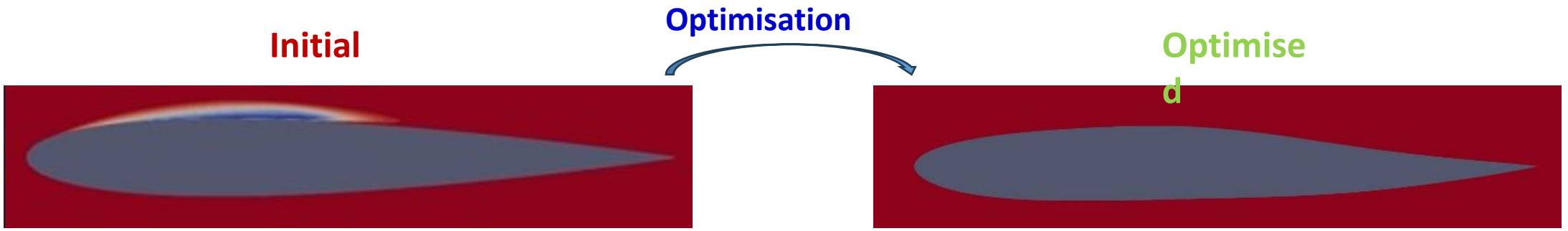


The (Continuous) Adjoint Method in ShpO for Two-Phase/Cavitated Flows

Why Cavitation is so Important?

Cavitation model should be implemented into a two-phase flow solver!

The adjoint to this new primal/flow system of equations must be formulated and programmed!



MFLOPS
Funded by the European Union

Two-Phase Flow Simulation Model, incl. Cavitation

$$R_n^{MF} = \frac{\partial f_{nk}^{inv}}{\partial x_k} - \frac{\partial f_{nk}^{vis}}{\partial x_k} - S_n^{cav} = 0,$$

$$R^{cav} = \frac{\partial(\alpha v_k)}{\partial x_k} - \frac{\dot{m}}{\rho_L} = 0$$

$$\vec{f}_k^{inv} = \begin{bmatrix} v_k \\ \rho v_1 v_k + p\delta_{1k} \\ \rho v_2 v_k + p\delta_{2k} \\ \rho v_3 v_k + p\delta_{3k} \end{bmatrix}, \quad \vec{f}_k^{vis} = \begin{bmatrix} 0 \\ \tau_{1k} \\ \tau_{2k} \\ \tau_{3k} \end{bmatrix}, \quad \vec{S}^{cav} = \begin{bmatrix} \left(\frac{1}{\rho_L} - \frac{1}{\rho_V}\right)\dot{m} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{cases} \tau_{mk} = (\mu + \mu_T)\left(\frac{\partial v_k}{\partial x_m} + \frac{\partial v_m}{\partial x_k}\right) \\ \rho = \alpha\rho_L + (1-\alpha)\rho_V \\ \mu = \alpha\mu_L + (1-\alpha)\mu_V \end{cases}$$

$$\dot{m} = \underbrace{\frac{C_{dest}\rho_V \alpha \min(0, p - p_v)}{\frac{1}{2}\rho_L U^2 t_\infty}}_{\text{Vaporisation}} + \underbrace{\frac{C_{prod}\rho_V \alpha^2(1-\alpha)}{t_\infty}}_{\text{Condensation}}$$

→ **Mass transfer between liquid and vapor phase.**

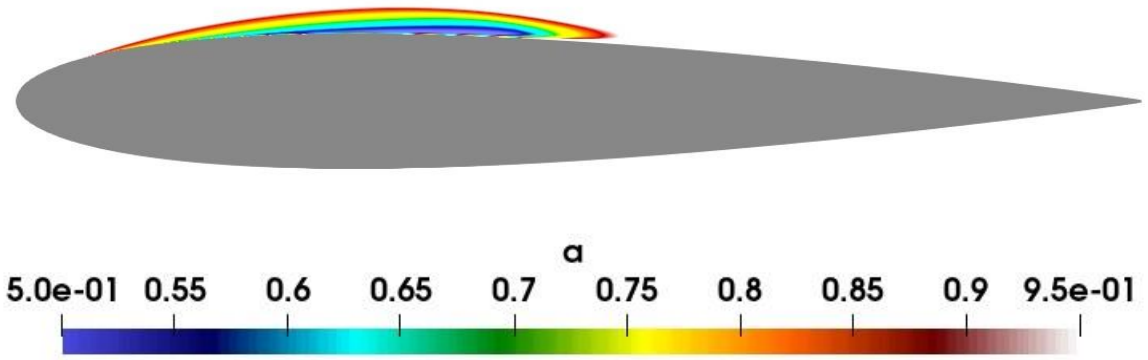
MFLOPS Funded by the European Union

Kunz et al. *A preconditioned Navier–Stokes method for two-phase flows with application to cavitation prediction*. Comp. & Fluids (2000).

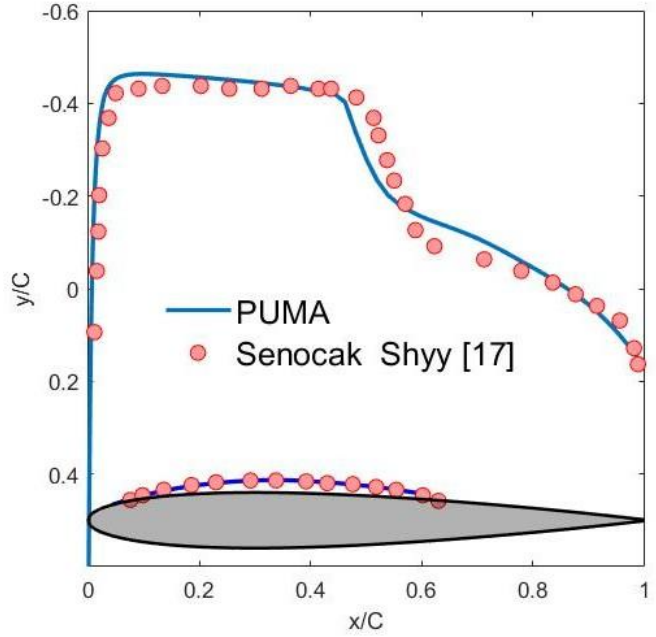
Assessment of the Kunz Cavitation model (Primal Solver) (1/2)

Turbulent cavitating flow around a NACA0012 hydrofoil.

- $\sigma = \frac{p_\infty - p_{vap}}{1/2 \rho U^2} = 0.42.$
- $Re = 2 \times 10^6$
- $\frac{\rho_L}{\rho_{vap}} = 1000$



Liquid volume fraction field



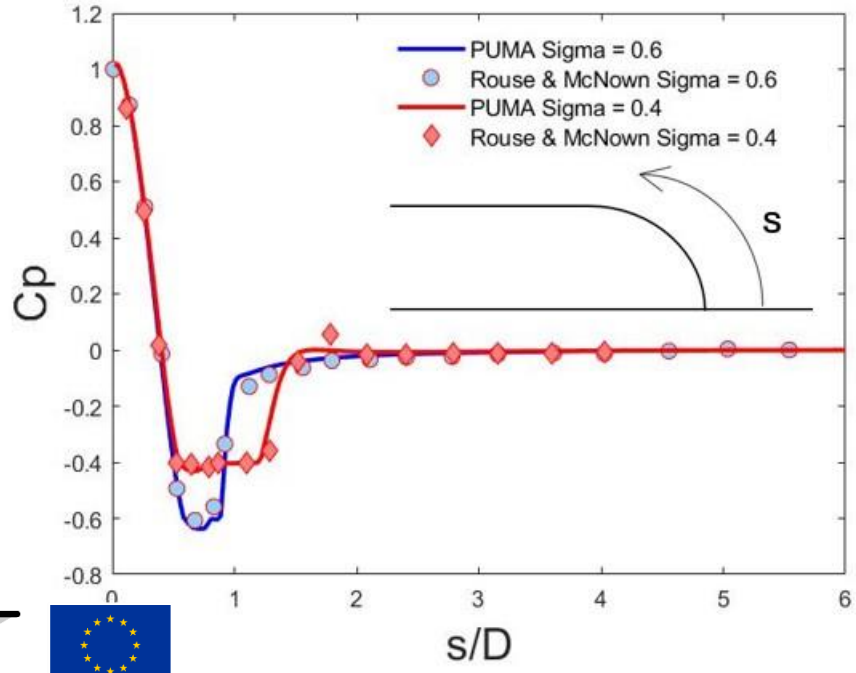
Pressure coefficient distribution over the suction side, and $\alpha = 0.99$ contour

MFLOPS
Funded by the European Union

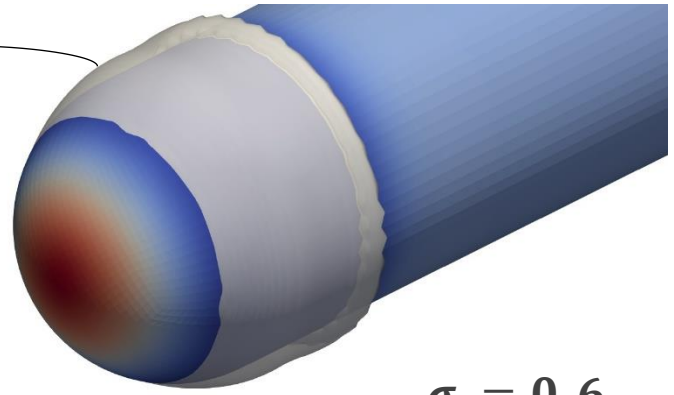
Assessment of the Kunz Cavitation model (Primal Solver) (2/2)

Turbulent cavitating flow around a hemispherical head body.

- $\sigma = 0.4, \sigma = 0.6.$
- $Re = 2 \times 10^5$, $\frac{\rho_L}{\rho_{vap}} = 1000$

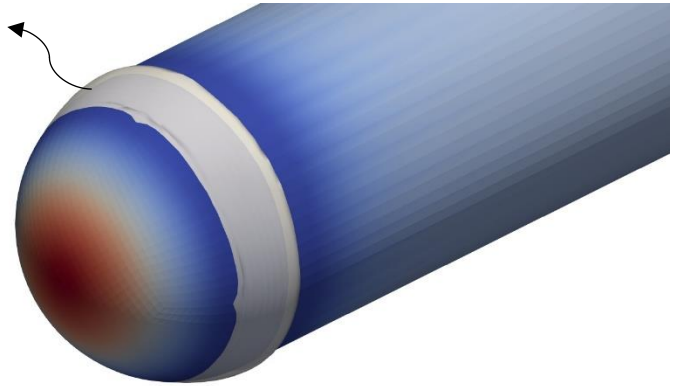


$\sigma = 0.4$



Cavitation bubble

$\sigma = 0.6$



Pressure & liquid fraction iso-surfaces, at $\alpha = 0.9$

MFLOPS
Funded by the European Union

Continuous Adjoint Method for Two-Phase Flows

$$L = F + \int \Psi_m R_m d\Omega + \int \Psi^\alpha R^{cav} d\Omega$$

$R_{\Psi^p} = 0$, adjoint continuity

$R_{\Psi^{\vec{u}}} = 0$, adjoint momentum

$R_{\Psi^\alpha} = 0$, adjoint liquid volume fraction

$$\frac{\delta L}{\delta b_i} = \int R_{\Psi^p} \frac{\delta p}{\delta b_i} d\Omega + \int R_{\Psi^{\vec{u}}} \frac{\delta \vec{u}}{\delta b_i} d\Omega + \int R_{\Psi^\alpha} \frac{\delta \alpha}{\delta b_i} d\Omega + \cdots$$

$$\ldots + \int ABC^p \frac{\delta p}{\delta b_i} dS + \int ABC^{\vec{u}} \frac{\delta \vec{u}}{\delta b_i} dS + \int ABC^\alpha \frac{\delta \alpha}{\delta b_i} dS + \int SD_1 \frac{\delta x_l}{\delta b_i} dS + \int SD_2 \frac{\delta n_k}{\delta b_i} dS$$

$ABC^{p,\vec{u},\alpha} = 0$
Adjoint boundary conditions

Sensitivity derivatives

MFLOPS
Funded by the European Union

Continuous Adjoint Method for Two-Phase Flows

Primal Mean-flow equations

$$R_n^{MF} = \frac{\partial f_{nk}^{inv}}{\partial x_k} - \frac{\partial f_{nk}^{vis}}{\partial x_k} - S_n^{cav} = 0$$

Adjoint Mean-flow equations

$$R_{\psi_n} = -A_{nmk}\frac{\partial \psi_n}{\partial x_k} - \frac{\partial \phi_{nk}^{vis}}{\partial x_k} + T_n^a - \psi_1\left(\frac{1}{\rho_L} - \frac{1}{\rho_V}\right)\frac{\partial S_n^{cav}}{\partial p} - \psi^\alpha \frac{1}{\rho_L}\frac{\partial \dot{m}}{\partial p}\delta_{1,n} - \alpha\frac{\partial \psi^\alpha}{\partial x_k}\delta_{k+1,n} = 0$$

Primal Liquid volume fraction equation

$$R^{cav} = \frac{\partial(\alpha v_k)}{\partial x_k} - \frac{\dot{m}}{\rho_L} = 0$$

Adjoint Liquid volume fraction equation

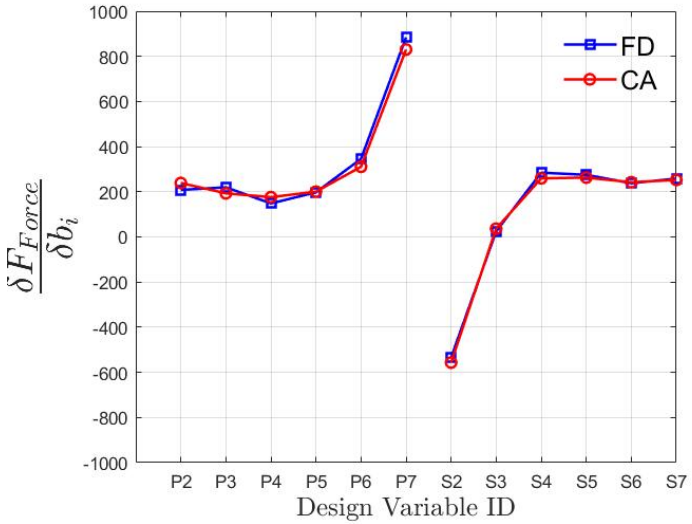
$$R_{\psi_\alpha} = -v_k\frac{\partial \psi^\alpha}{\partial x_k} - \frac{\partial \psi_{m+1}}{\partial x_k}v_m v_k(\rho_L - \rho_V) - \psi^\alpha\frac{1}{\rho_L}\frac{\partial \dot{m}}{\partial \alpha} - \frac{\partial \psi_{m+1}}{\partial x_k}\frac{\tau_{mk}}{\mu+\mu_t}(\mu_L - \mu_V) - \psi_1\left(\frac{1}{\rho_L} - \frac{1}{\rho_V}\right)\frac{\partial \dot{m}}{\partial \alpha} + S_{SA} = 0$$

MFLOPS
Funded by the European Union

ShpO cases: NACA0012 Hydrofoil (1/2)

$$FForce = \int [(p\delta_{ij} - \tau_{ij}) r_i] n_j dS$$

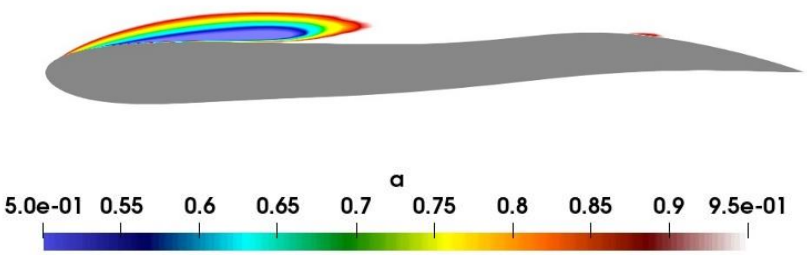
- $\sigma = \frac{p_\infty - p_{vap}}{1/2 \rho U^2} = 0.42.$
- $Re = 2 \times 10^6$
- $\frac{\rho_L}{\rho_{vap}} = 1000$

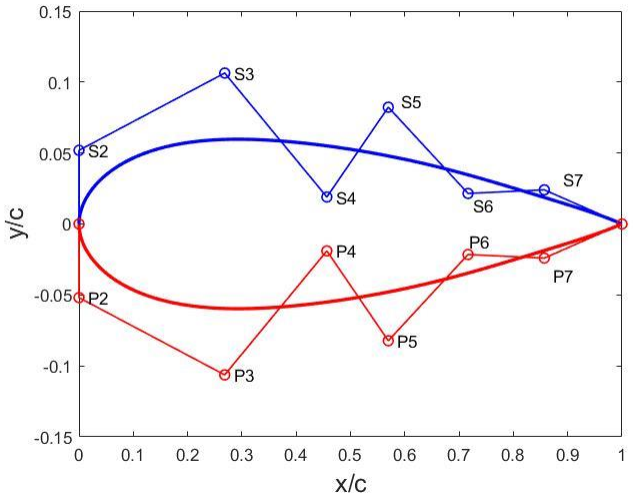


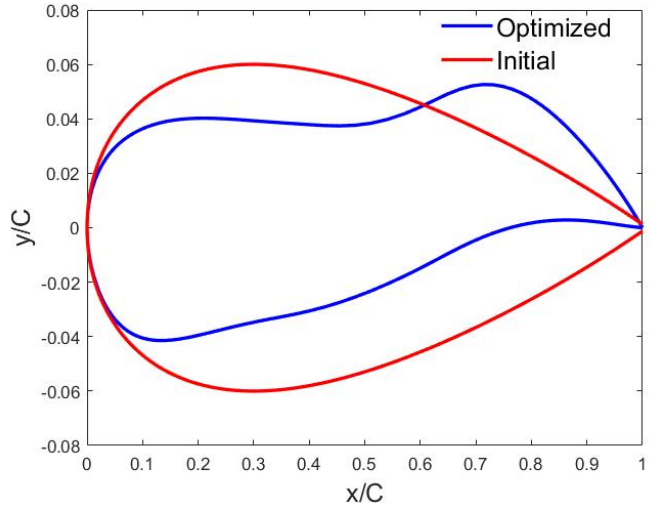
MFLOPS

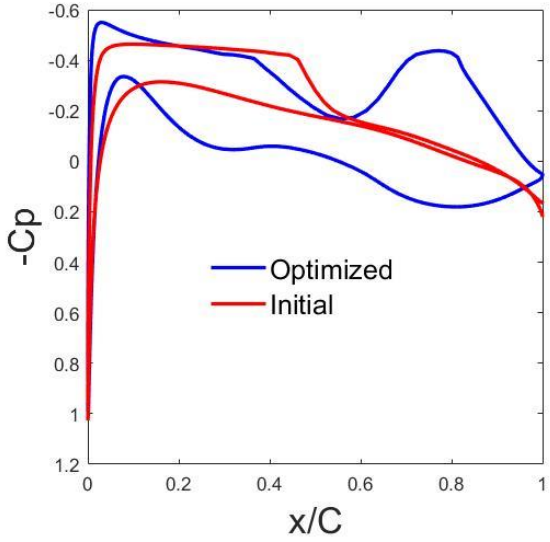
Funded by the European Union

PCOpt





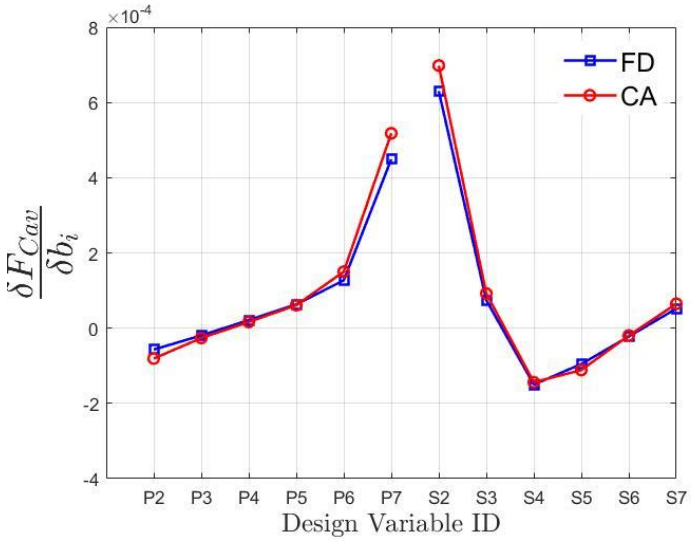


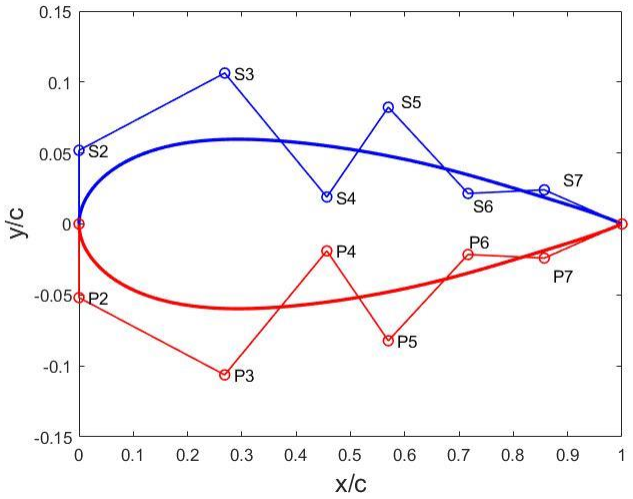


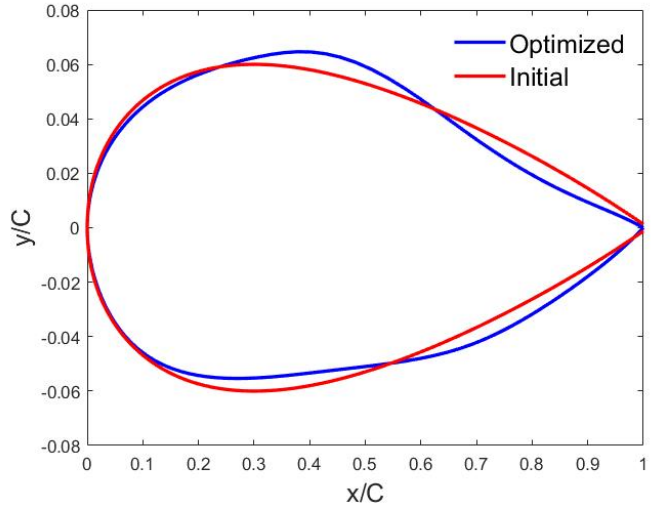
ShpO cases: NACA0012 Hydrofoil (2/2)

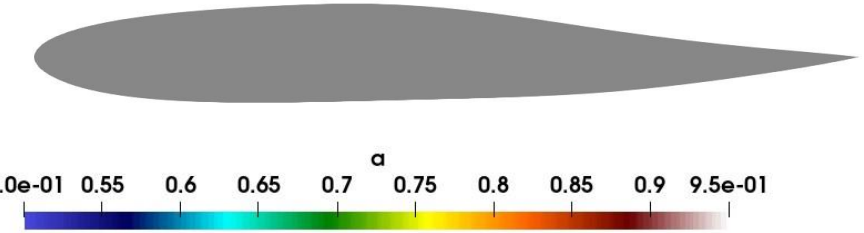
$$FCav = \frac{1}{2}\int (1 - \alpha)^2 d\Omega$$

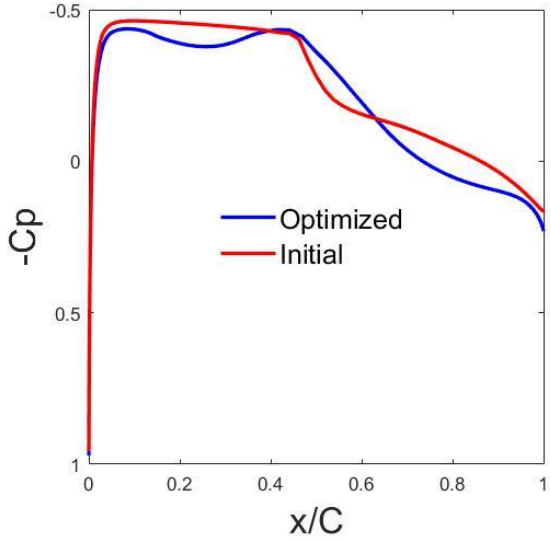
- $\sigma = \frac{p_\infty - p_{vap}}{1/2 \rho U^2} = 0.42.$
- $Re = 2 \times 10^6$
- $\frac{\rho_L}{\rho_{vap}} = 1000$











ShpO cases: Hemispherical Head body (1/2)

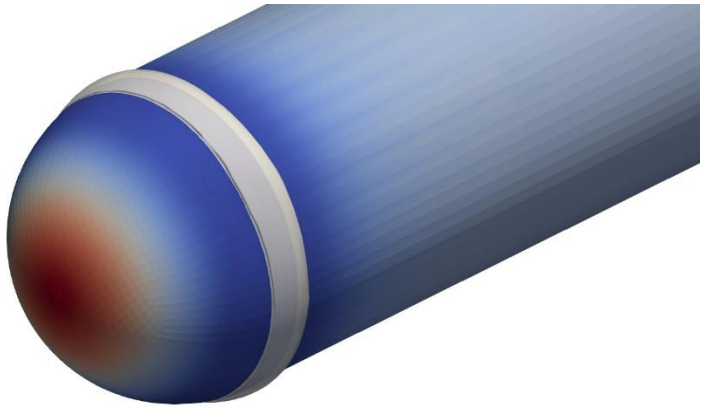
$$F_{Cav} = \frac{1}{2}\int (1 - \alpha)^2 d\Omega$$

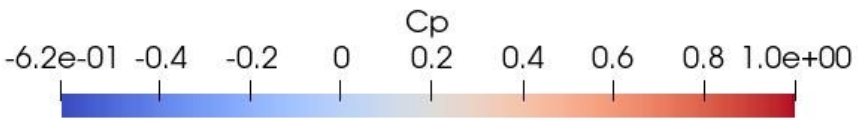
- $Re = 2 \times 10^6$
- $\frac{\rho_L}{\rho_{vap}} = 1000$

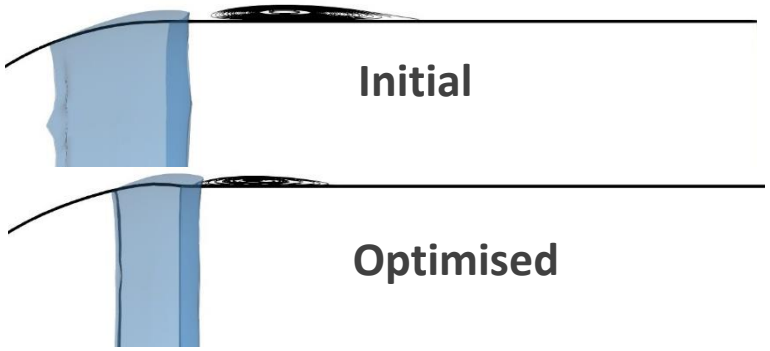
MFLOPS

Funded by the European Union

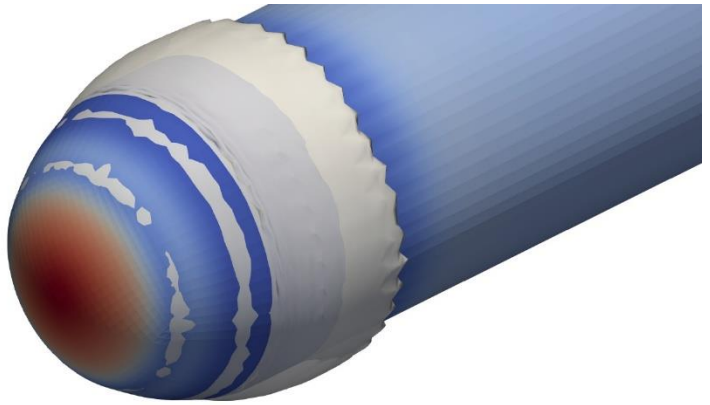
$\sigma = 0.6$

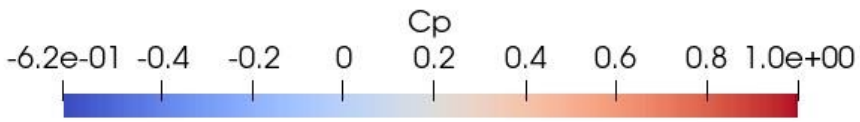


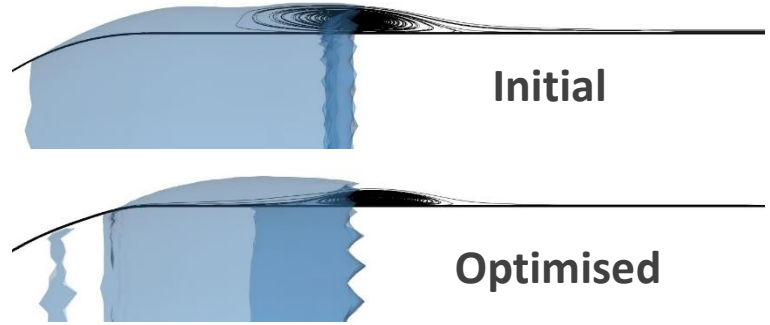




$\sigma = 0.4$



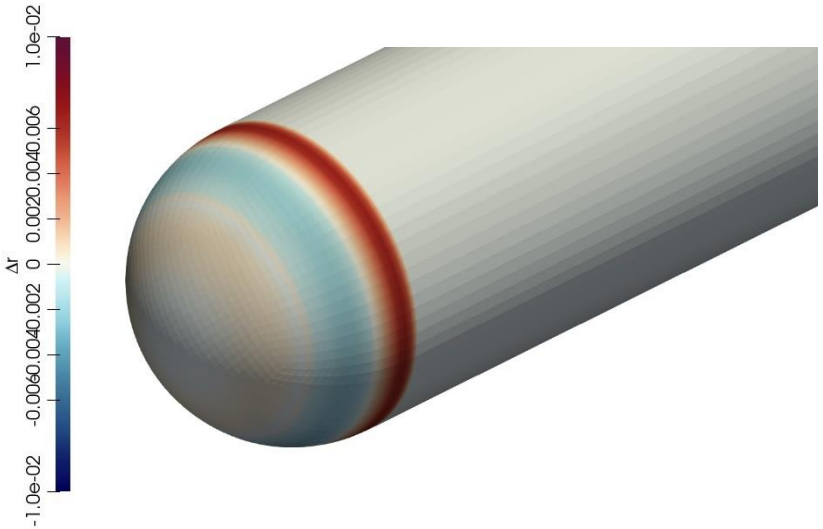


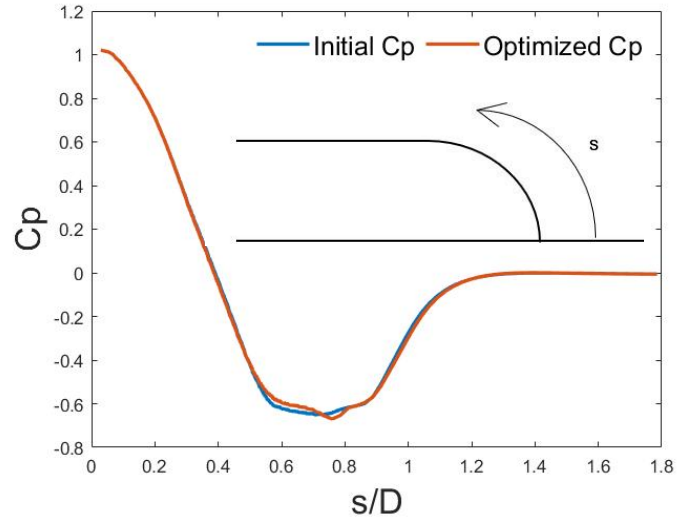


~30% reduction in F_{cav} for both cavitation numbers

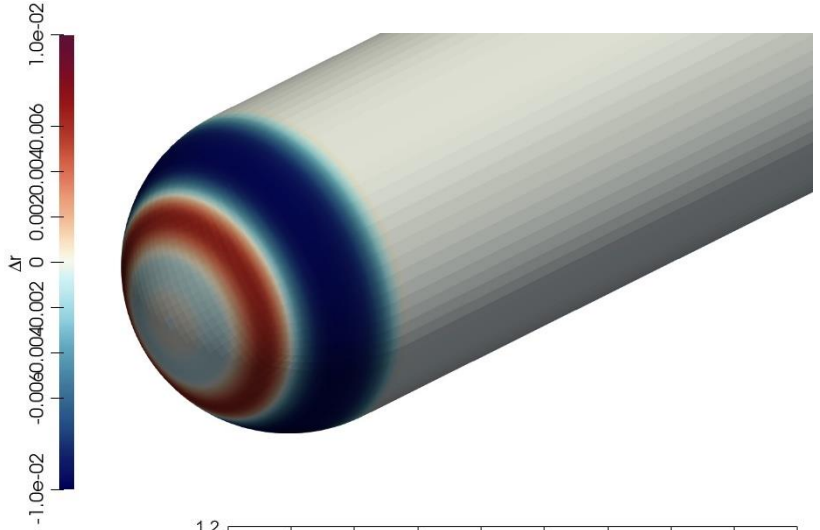
ShpO cases: Hemispherical Head body (2/2)

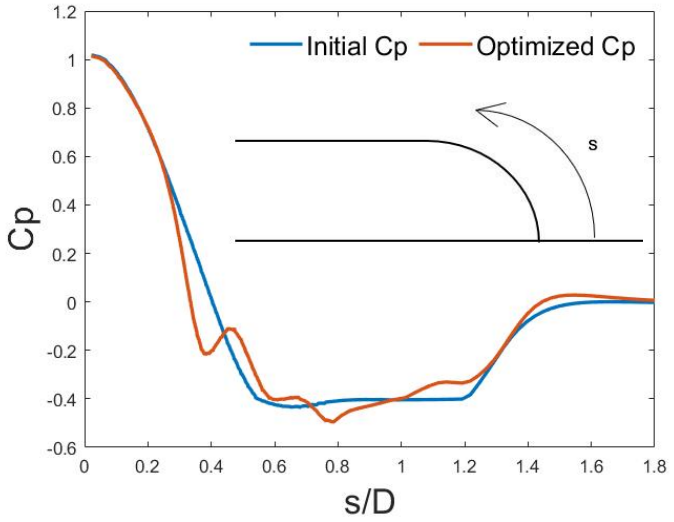
$\sigma = 0.6$



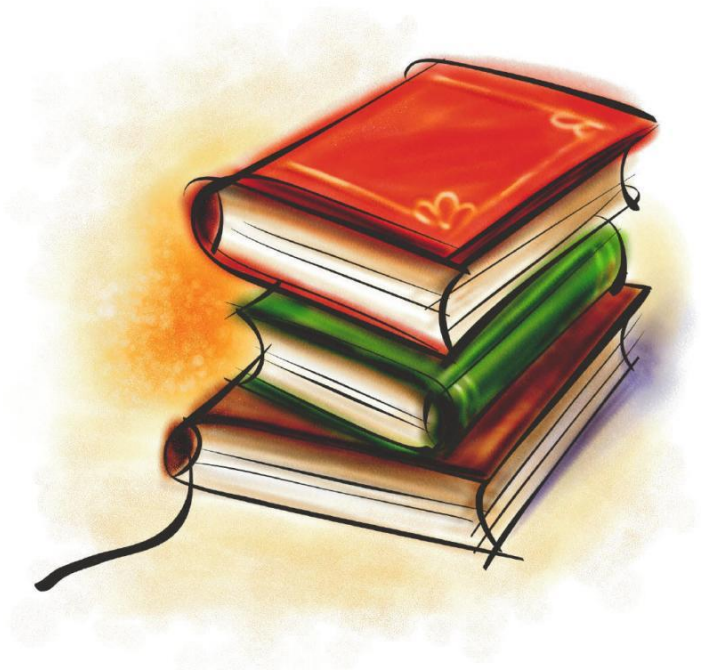


$\sigma = 0.4$





MFLOPS
Funded by the European Union

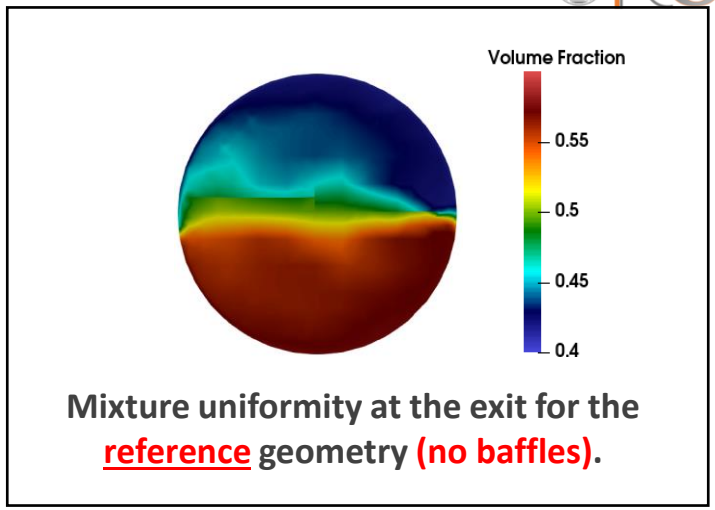


The (Continuous) Adjoint Method in ShpO for Static Mixers

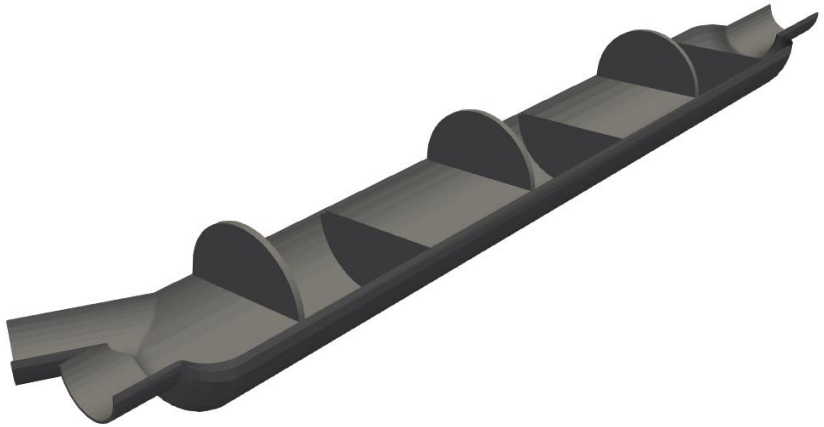
Prof. K.C. Giannakoglou, kgianna@mail.ntua.gr, Dr. V.G. Asouti, vasouti@mail.ntua.gr

Adjoint-based ShpO of Mixing Devices

- **ShpO of static mixing devices with motionless compartments (e.g. baffles) used for the continuous blending of two miscible fluids.**
- **Optimise the 6 baffles' shapes, targeting:**
  - **Max. Mixture Uniformity at the exit J_U**
  - **Min. Total Pressure Losses J_{Pt}**
- **ShpO performed by extending the *adjointOptimisationFoam* s/w, developed by the PCOpt/NTUA.**
- **The starting (reference) geometry was manually designed.**



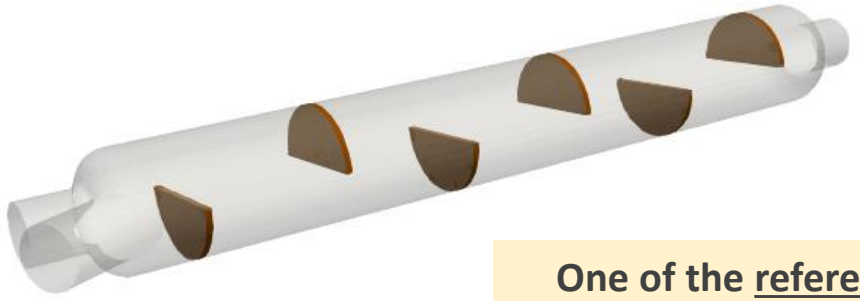
Mixture uniformity at the exit for the reference geometry (no baffles).



MSc Theses by N. Galanos, P. Alexias, D. Katmeridis NTUA.

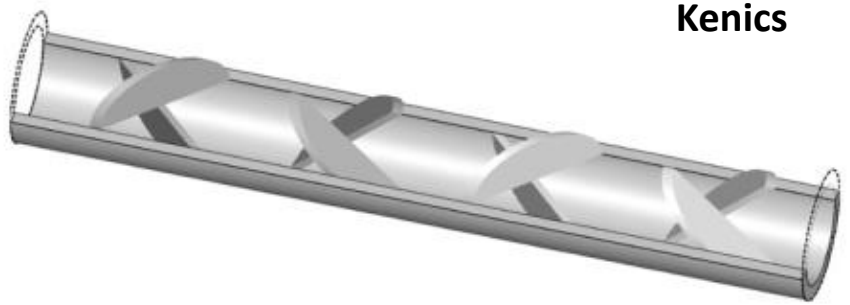
History of Static Mixers

- **First application as air-fuel mixture in 1874.**
- **Widely used in industry as chemical reactors from 1970.**
- **Static mixtures: no power consumption.**

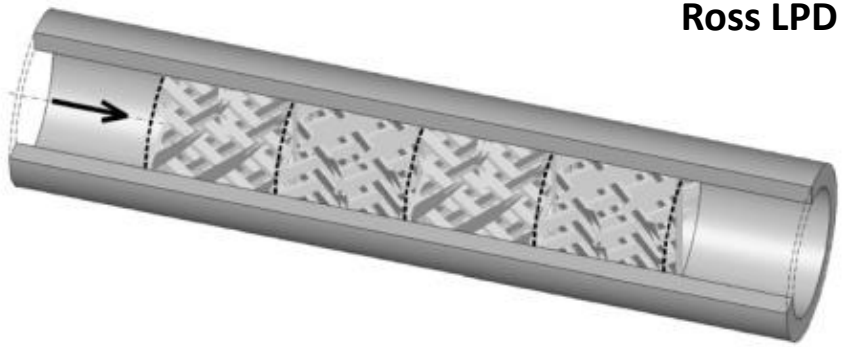


One of the reference designs proposed by the PCOpt/NTUA

Kenics



Ross LPD



Sulzer SMX

Miscible Fluid Flow Model - Primal Equations

- **The two incompressible, miscible fluids are modeled (laminar flow) using the Volume of Fluid (VoF) method. α denotes the volume fraction of phase 1 inside the mixture.**

$$R^p = -\frac{\partial(\rho v_j)}{\partial x_j} = 0$$

$$R_i^v = \rho v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial(\mu \epsilon_{ij})}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0, i = 1,2,3$$

$$R^\alpha = v_i \frac{\partial \alpha}{\partial x_i} - \frac{\partial}{\partial x_j}\left(D \frac{\partial \alpha}{\partial x_j}\right) = 0$$

where $\epsilon_{ij} = \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}$ is the strain tensor.

- **Mixture properties**

$$\rho = \alpha\rho_1 + (1-\alpha)\rho_2, \qquad \mu = \alpha\mu_1 + (1-\alpha)\mu_2$$

- **Laplacian Grid-displacement PDEs (as in the E-SI adjoint!)**

$$R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0, i = 1,2,3$$

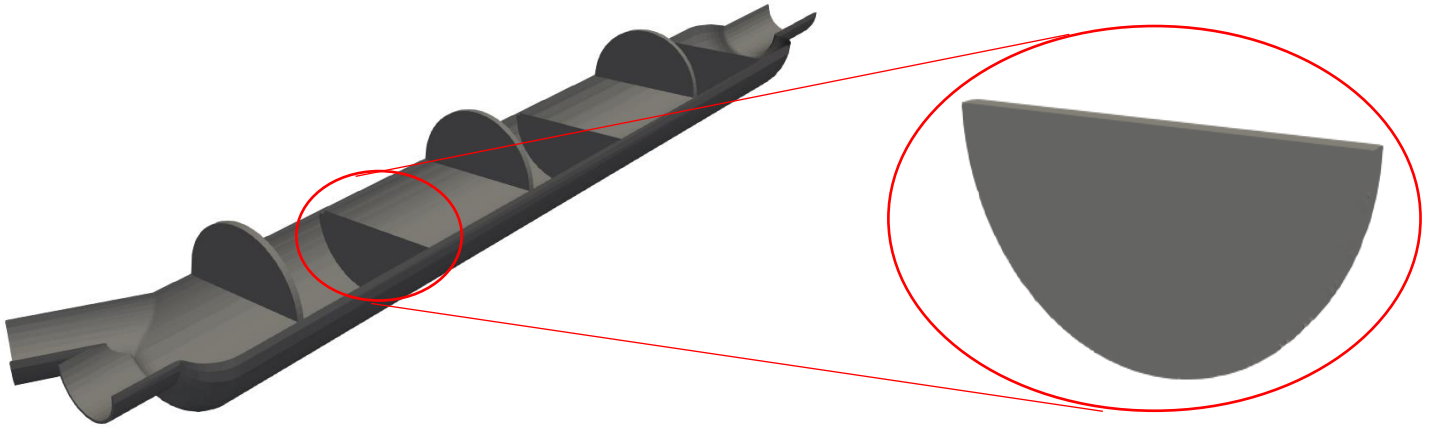
Objective Functions

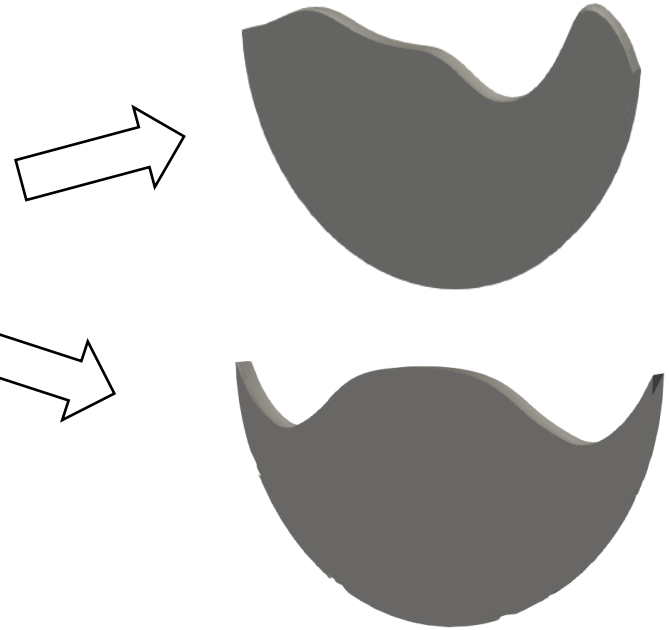
- **Two objective functions :**
  - **Mixture uniformity at the exit:**

$$J_U = \frac{1}{2}\int_{S_O} v_i n_i \left(\alpha - \frac{1}{|S_O|}\int_{S_O} \alpha dS\right)^2 dS$$

  - **Total pressure losses:** $J_{Pt} = -\int_{S_{I,O}} v_i n_i \left(p + \frac{1}{2}\rho v_j^2\right) dS$
- **The two performance-metrics (min.) are combined into a SOO problem by minimizing their weighted sum**

$$J = w_1 J_U + w_2 J_{Pt}$$





The Adjoint System of Equations

- The development of the E-SI adjoint method starts by defining the Lagrangian

$$L = J + \int_\Omega q R^p d\Omega + \int_\Omega u_i R_i^v d\Omega + \int_\Omega \psi R^\alpha d\Omega + \int_\Omega m_i^\alpha R_i^m d\Omega$$

and differentiating it w.r.t. b_n using the Leibniz theorem

$$\frac{\delta L}{\delta b_n} = \frac{\delta J}{\delta b_n} + \int_\Omega q \frac{\partial R^p}{\partial b_n} d\Omega + \int_\Omega u_i \frac{\partial R_i^v}{\partial b_n} d\Omega + \int_\Omega \psi \frac{\partial R^\alpha}{\partial b_n} d\Omega + \int_\Omega m_i^\alpha \frac{\partial R_i^m}{\partial b_n} d\Omega + \int_S (q R^p + u_i R_i^v + \psi R^\alpha + m_i^\alpha R_i^m) \frac{\delta x_k}{\delta b_n} n_k dS$$

Grid Sensitivities

Leibniz Term

- Adjoint two-phase flow equations

$$R^q = -\frac{\partial u_i}{\partial x_i} = 0$$

$$R_i^u = \rho u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial (\rho v_j u_i)}{\partial x_j} - \frac{\partial (\mu \epsilon_{ij}^\alpha)}{\partial x_j} + \rho \frac{\partial q}{\partial x_i} + \psi \frac{\partial \alpha}{\partial x_i} = 0, i = 1,2,3$$

$$R^\psi = -\frac{\partial (v_i \psi)}{\partial x_i} - \frac{\partial}{\partial x_j}\left(D \frac{\partial \psi}{\partial x_j}\right) + \rho_\Delta \left(v_i \frac{\partial q}{\partial x_i} + u_i v_j \frac{\partial v_i}{\partial x_j}\right) + \mu_\Delta \frac{\partial u_i}{\partial x_j} \epsilon_{ij} = 0$$

where $\rho_\Delta = \rho_1 - \rho_2,\ \mu_\Delta = \mu_1 - \mu_2$.

- Adjoint grid-displacement PDEs in order to eliminate grid sensitivities…

$$R_k^{m^\alpha} = \frac{\partial^2 m_k^\alpha}{\partial x_j^2} - \frac{\partial}{\partial x_j}\left\{ q \frac{\partial (\rho v_j)}{\partial x_k} - \rho u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} + u_i \frac{\partial (\mu \epsilon_{ij})}{\partial x_k} - \mu \epsilon_{ij}^\alpha \frac{\partial v_i}{\partial x_k} - \psi v_j \frac{\partial \alpha}{\partial x_k} - D \frac{\partial \psi}{\partial x_j} \frac{\partial \alpha}{\partial x_k} + D\psi \frac{\partial^2 \alpha}{\partial x_k \partial x_j} \right\} = 0, k = 1,2,3$$

E-SI Adjoint Sensitivities

- **Computation of sensitivity derivatives (SDs) including only surface integrals (SI Adjoint) (reduced computational cost)!**

$$\frac{\delta J}{\delta b_n} = -\int_{S_W} \left[\left(-q n_i + \mu \epsilon_{ij}^{\alpha} n_j \right) \frac{\partial v_i}{\partial x_j} n_k + \frac{\partial m_k^{\alpha}}{\partial x_j} - D\psi \frac{\partial^2 \alpha}{\partial x_j \partial x_k} \right] \frac{\delta x_k}{\delta b_n} n_j dS + \int_{S_W} D\psi \frac{\partial \alpha}{\partial x_j} \frac{\delta n_j}{\delta b_n} dS$$

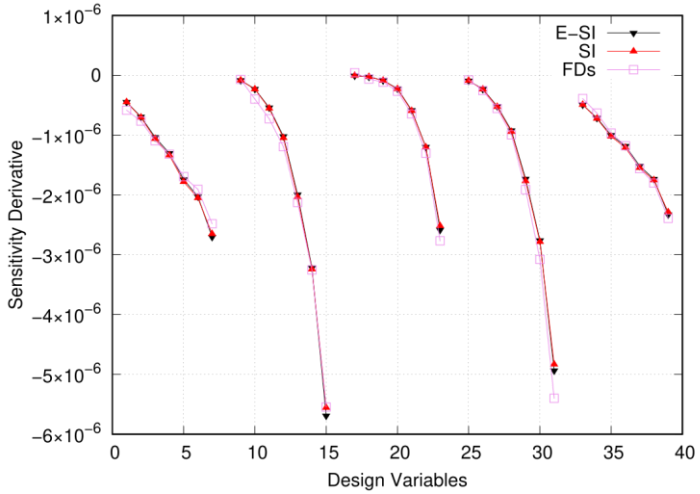
- **Extra terms, arising from the differentiation of the grid-displacement PDEs, to account for the effect of grid-sensitivities!**

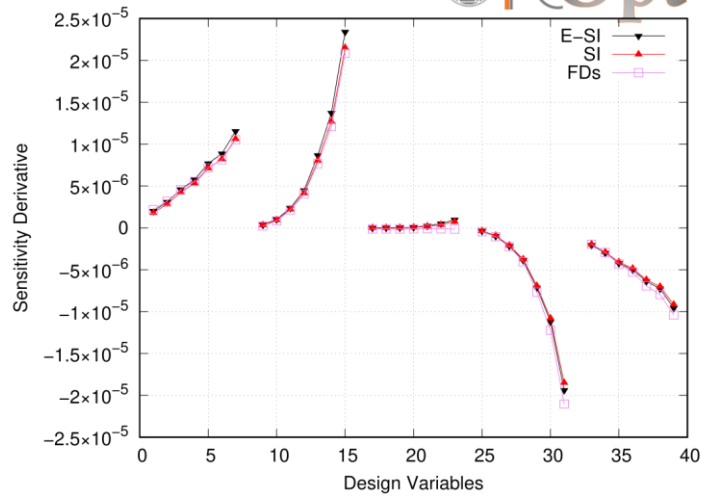
Comparison of Adjoint with FDs - Two-inlet Duct (Demo Case)

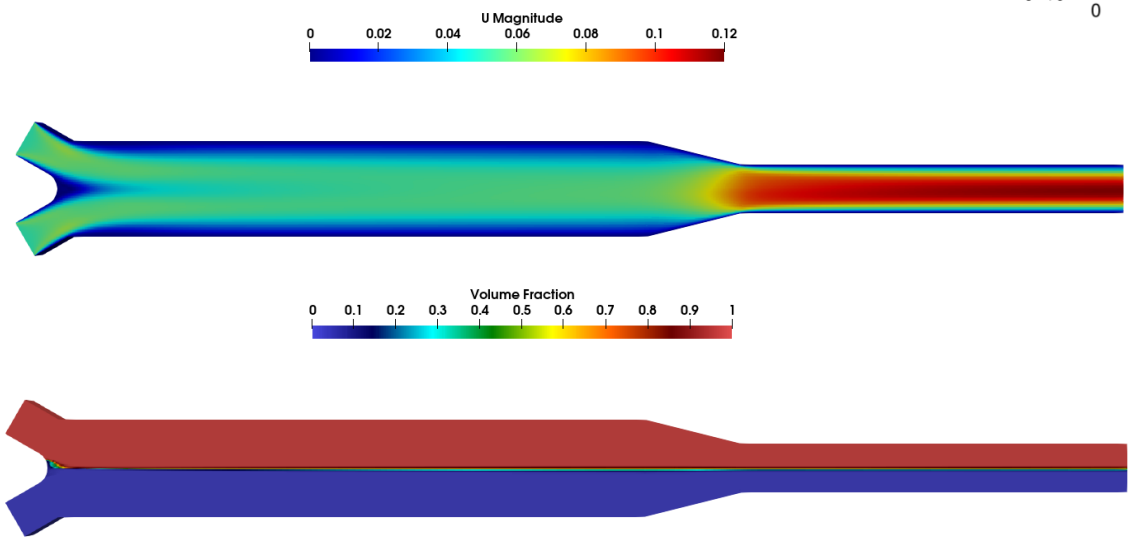
- **The only target: min. total pressure losses:**

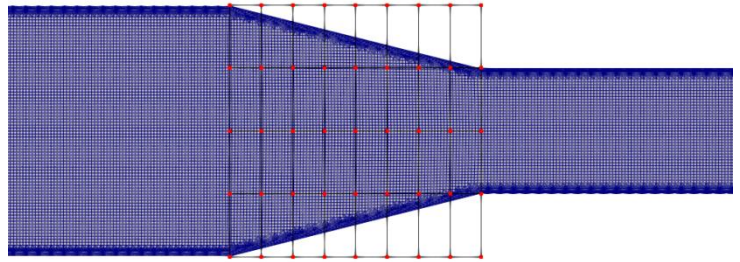
$$J_{Pt} = -\int_{S_{I,O}} v_i n_i \left(p + \frac{1}{2}\rho v_j^2\right) dS$$

- **Adjoint-based SDs compared with FDs!**
- **Mixing of the two fluids is no interest here!**









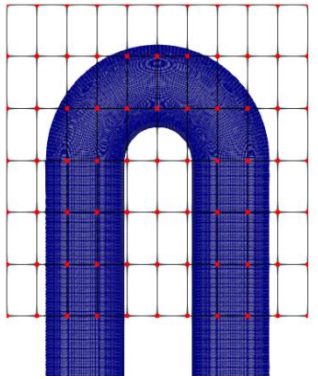
Parametrisation of the convergent part using volumetric NURBS.

Comparison of Adjoint with FDs - U-bend duct (Demo Case)

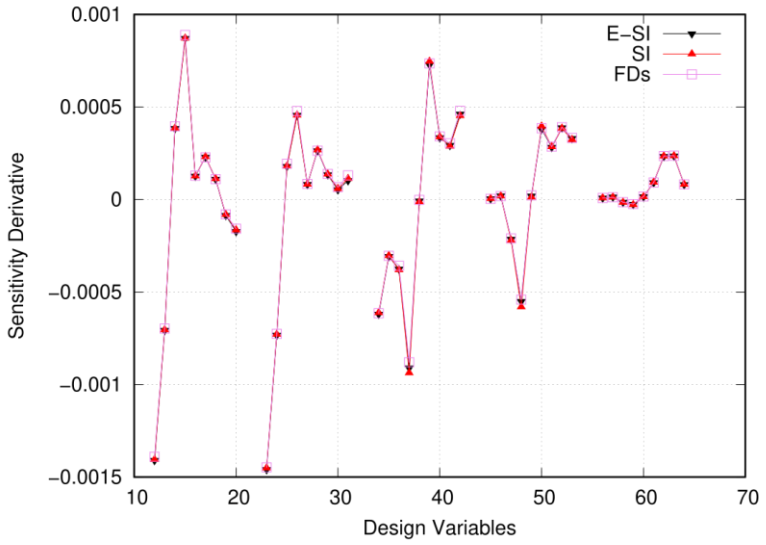
- **The only target: min. total pressure losses:**

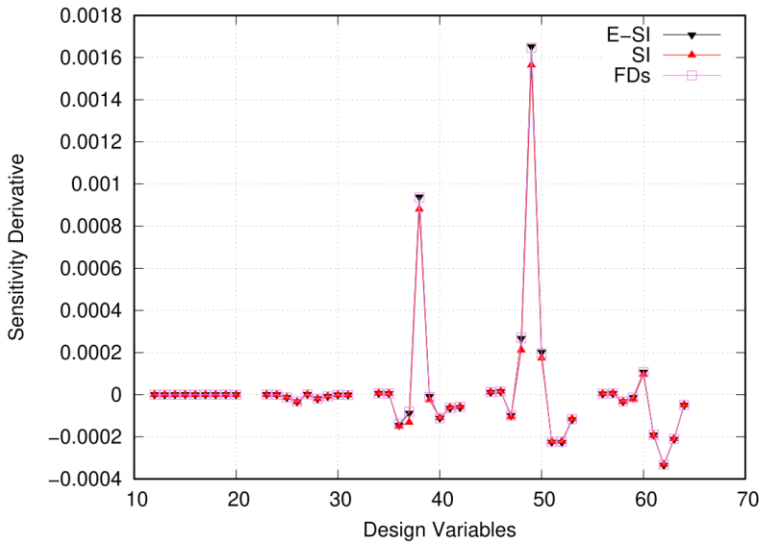
$$J_{Pt} = -\int_{S_{I,O}} v_i n_i \left(p + \frac{1}{2}\rho v_j^2 \right) dS$$

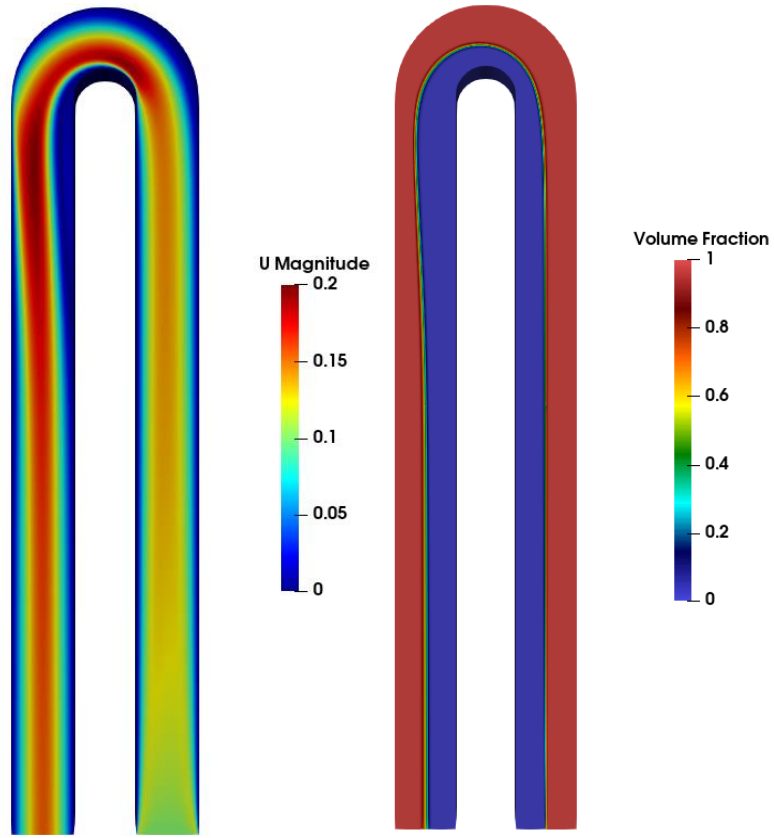
- **Adjoint-based SDs compared with FDs!**
- **Mixing of the two fluids is of no interest here!**



Parametrisation of the curved part using volumetric NURBS.





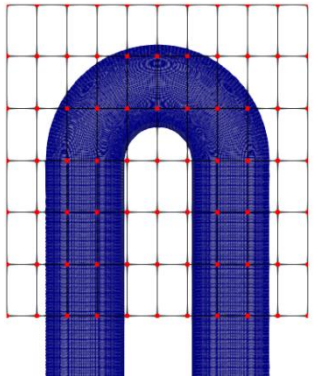


Same Demo Case, different target: Severed SI vs. E-SI Adjoint

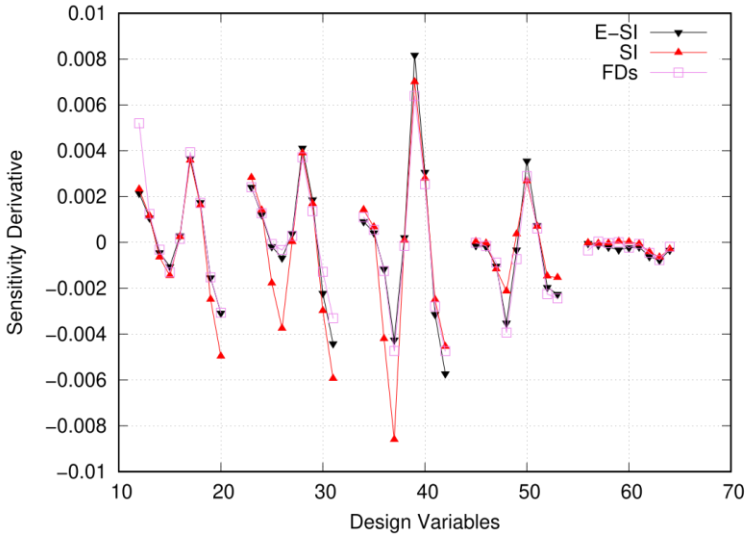
The only target: min. the mean value of α at the exit:

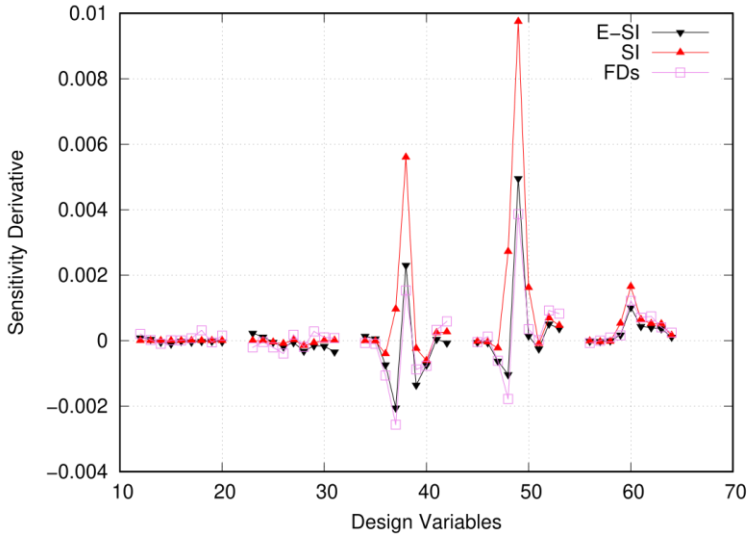
$$J_\alpha = \frac{1}{|S_O|}\int_{S_O} \alpha dS$$

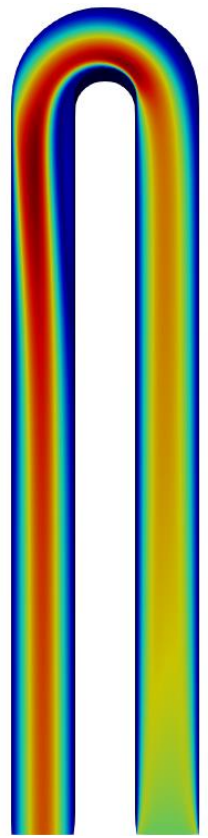
Inaccuracies in the computed if the "severed" SI adjoint was used.

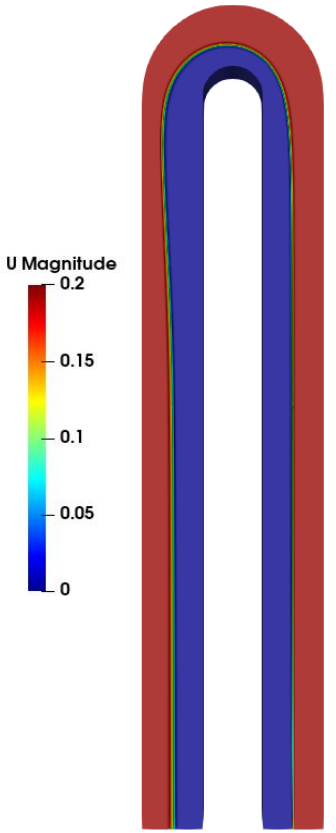


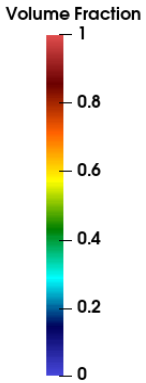
Parametrisation of the curved part using volumetric NURBS.



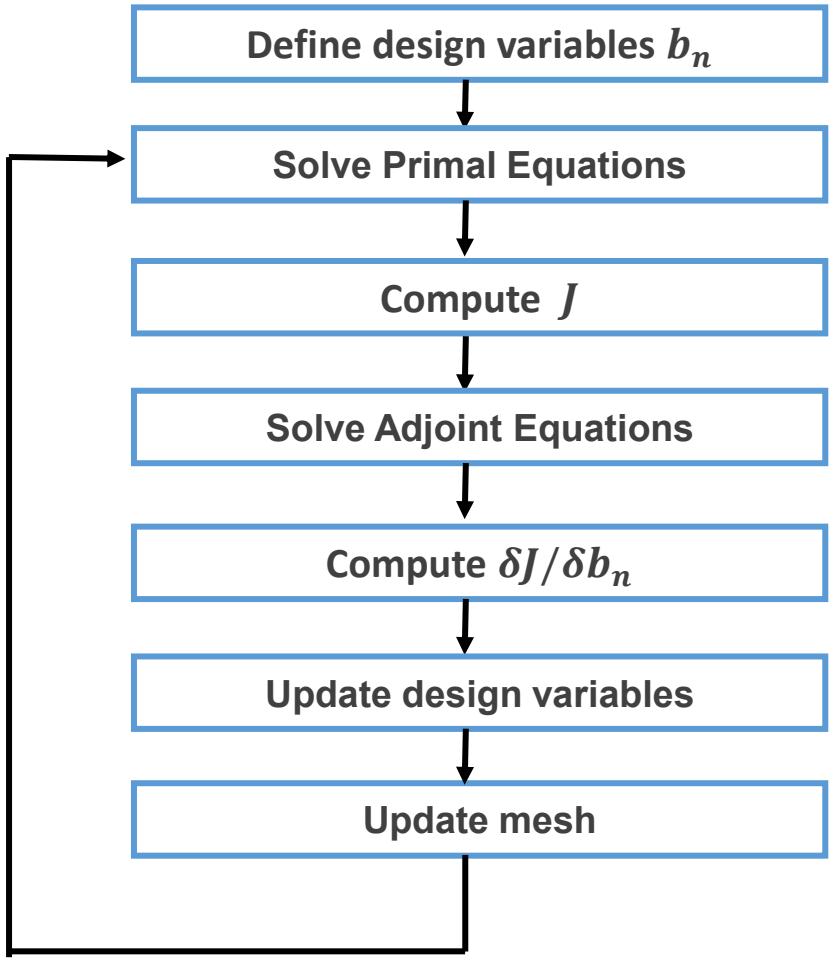








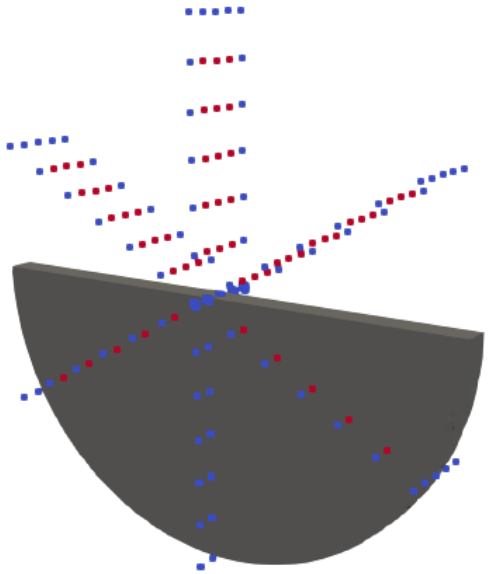
Parametrisation Scheme & Adjoint-based ShpO Workflow

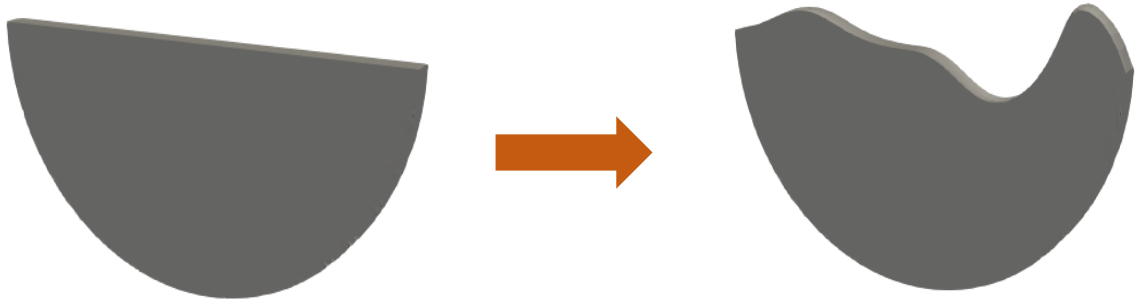


- Baffle Shape Parametrisation and Grid Adaptation based on a volumetric b-splines morphing technique.

$$x_m = U_{i,pu}(u)V_{j,pv}(v)W_{k,pw}(w)b_m^{ijk}$$

- 245 NURBS CPs control the shape of each baffle (cylindrical arrangement).
- For each baffle 75 active CPs are allowed to move only in the peripheral direction (75 design variables per baffle).
- Manufacturability constraints: retain the baffles' thickness and flatness at their tips.



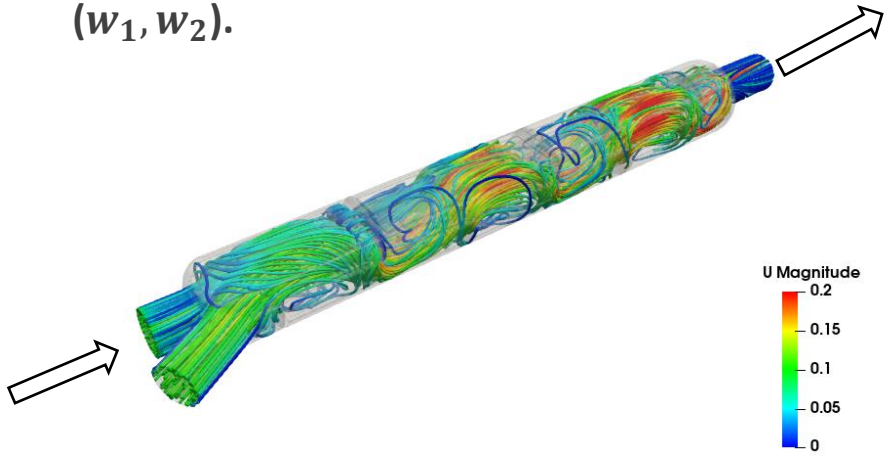


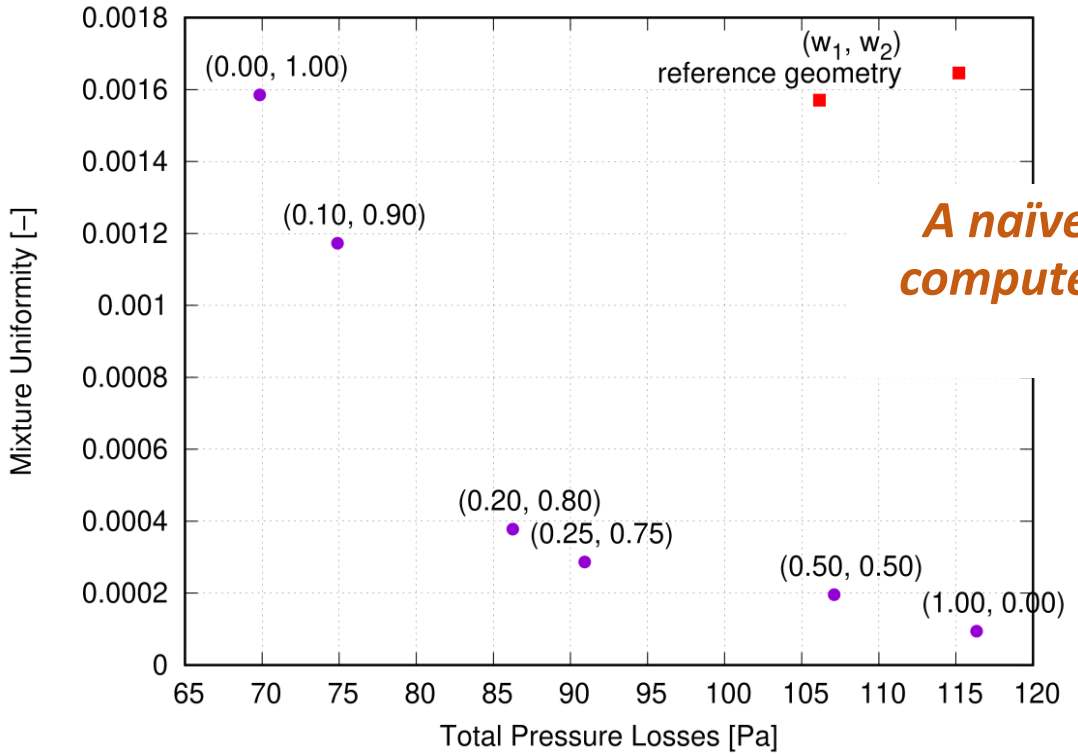
Results – ShpO of a Mixing Device with Baffles

- **Device length:** $L = 0.7m$
- **Device inner diameter:** $d_i = 0.1m$
- **Re=750**
- **Diffusion coefficient:** $D = 1.5 \cdot 10^{-8} \frac{m^2}{s}$
- **Objective function:**

$$J = w_1 J_U + w_2 J_{Pt}$$

- **The front of non-dominated solutions is computed by minimising *J* with several pairs of values for (w_1, w_2).**

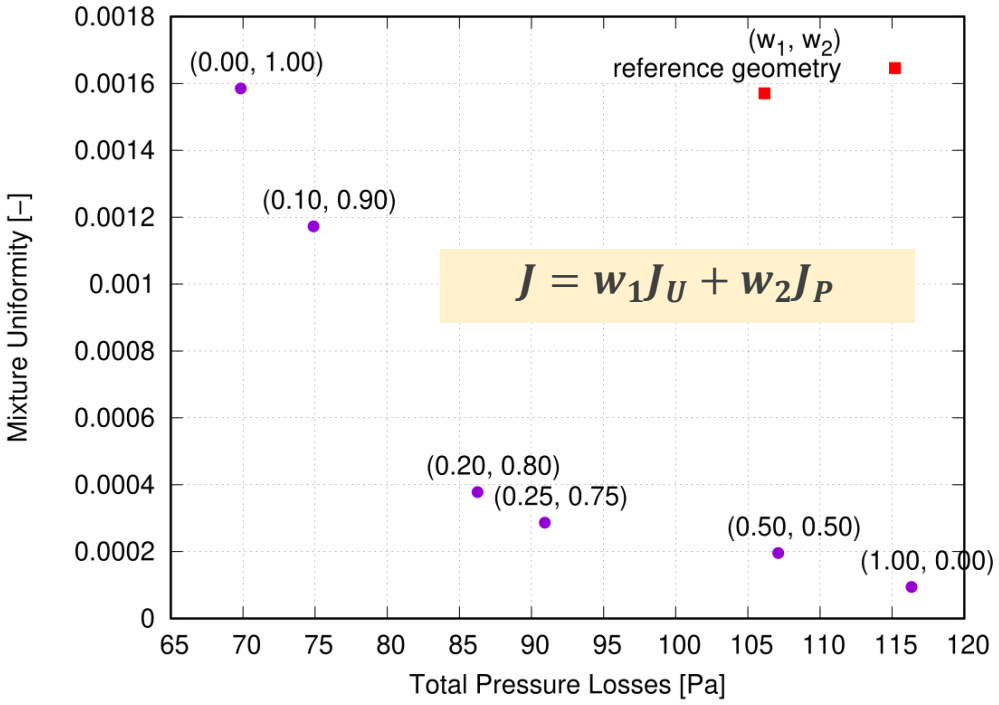


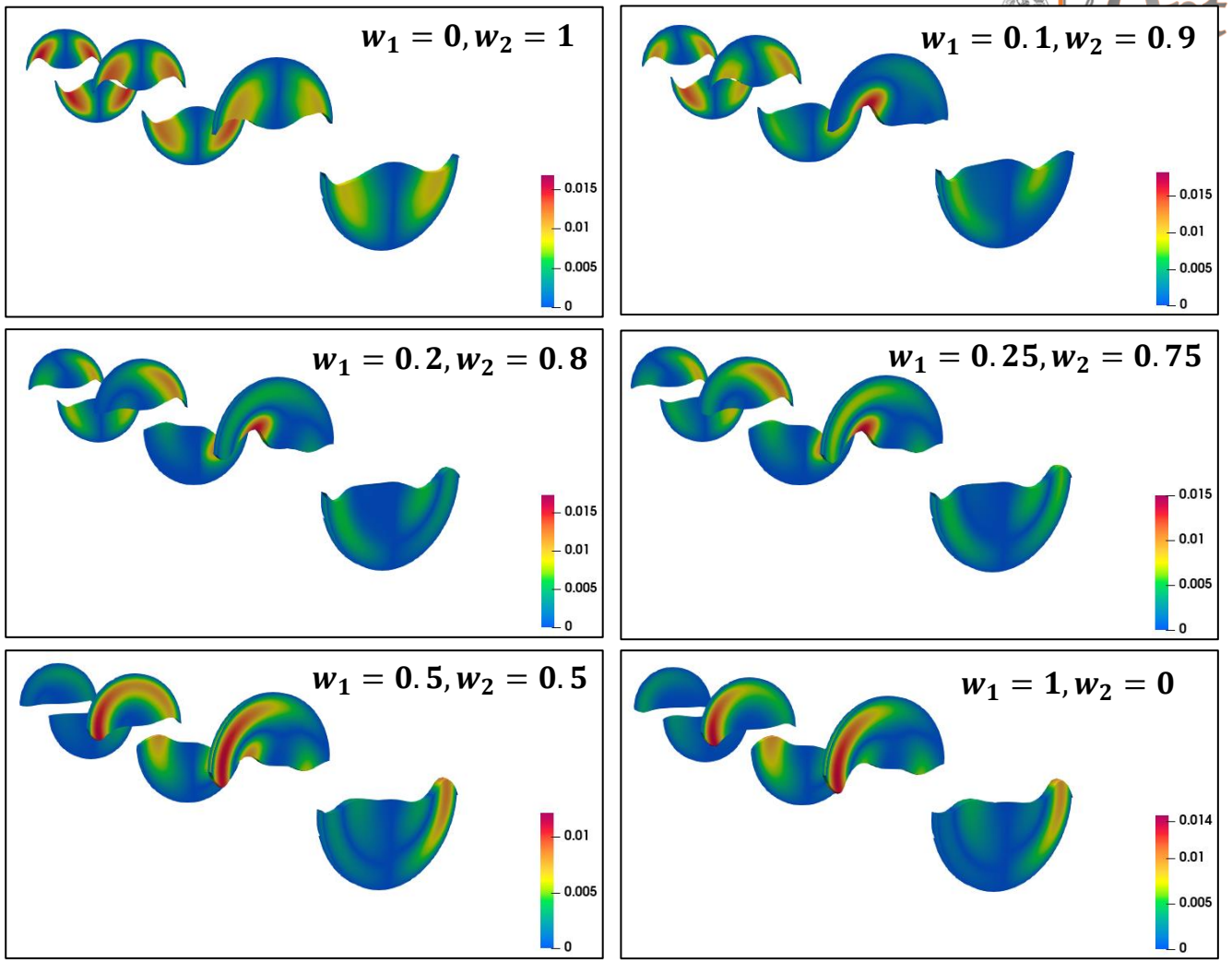


A naïve way to compute Pareto fronts

	Fluid 1	Fluid 2
Density (kg/m^3)	1500	1300
Kinematic viscosity (m^2/s)	$1.5 \cdot 10^{-5}$	$1.3 \cdot 10^{-5}$

Results – Optimal Baffle Shapes (two objectives)



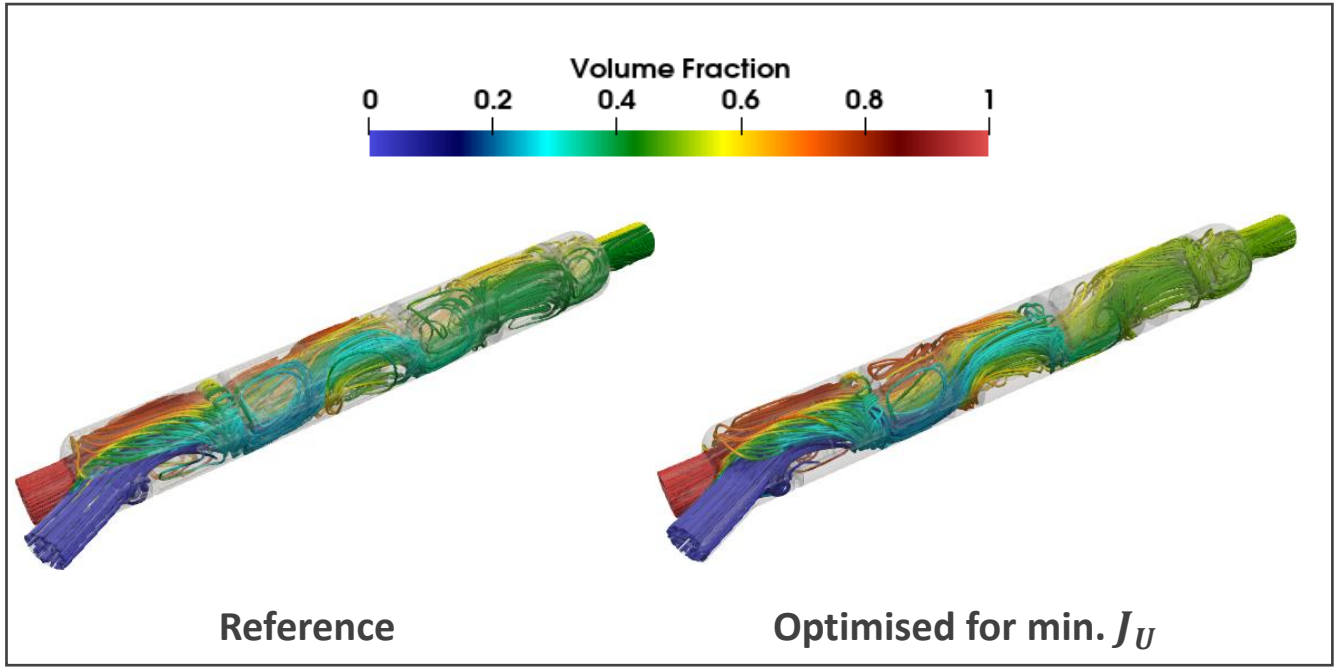


- Minimising J_{Pt} leads to reduced baffles' cross-sectional area.
- Minimising J_U leads to baffles of rippling shapes.

… as expected!

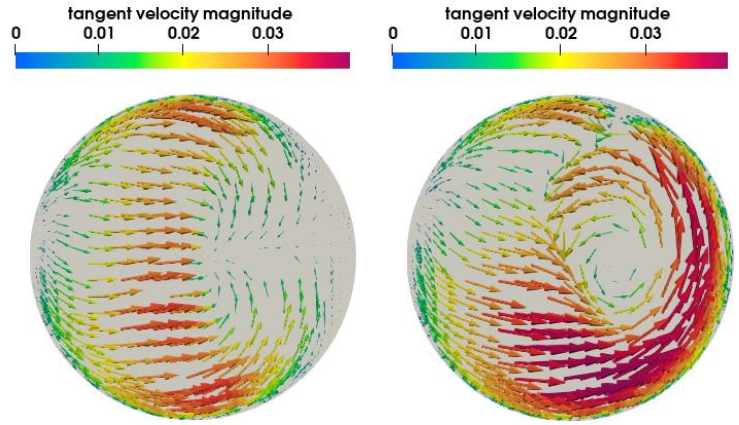
Results – Maximising (only) Mixture Uniformity

- **Minimising J_U leads to a 94% increase in uniformity, whereas J_P has increased by 9.6%.**
- **Gain achieved due to increased flow vortical structure!**



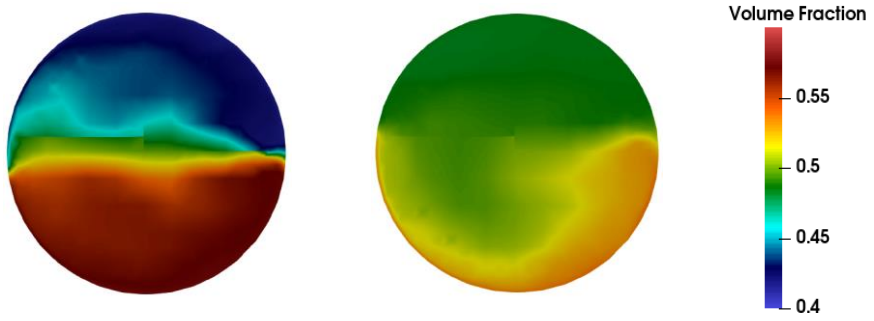
Reference **Optimised for min. J_U**

Tangent velocity vectors at a cross section in the middle of the mixer's length.



Reference **Optimized for min. J_U**

Volume fraction distribution at the exit

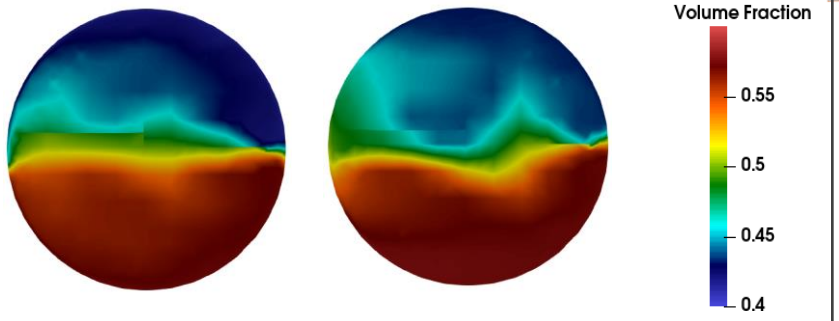


Reference **Optimized for min. J_U**

Results – Minimising (only) Total Pressure Losses

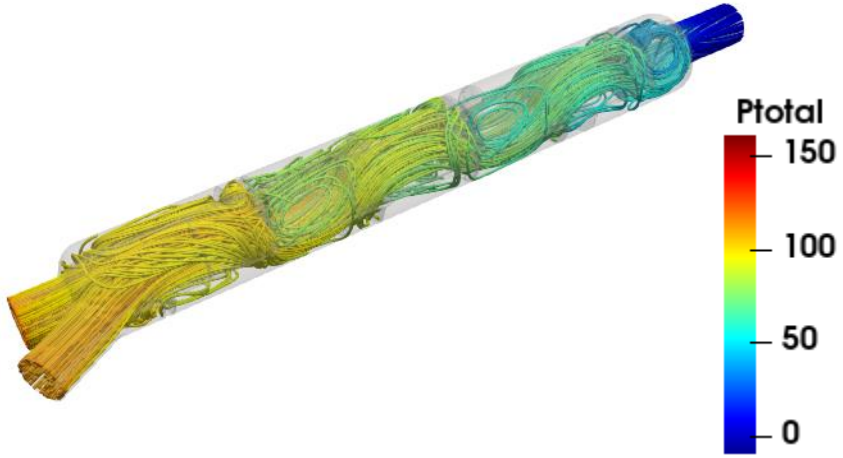
- **Minimising J_{Pt} leads to a 34% reduction in total pressure losses, whereas (non)uniformity remains practically the same.**
- **Gain achieved due to reduced flow separation behind the baffles!**

Volume fraction distribution at the exit



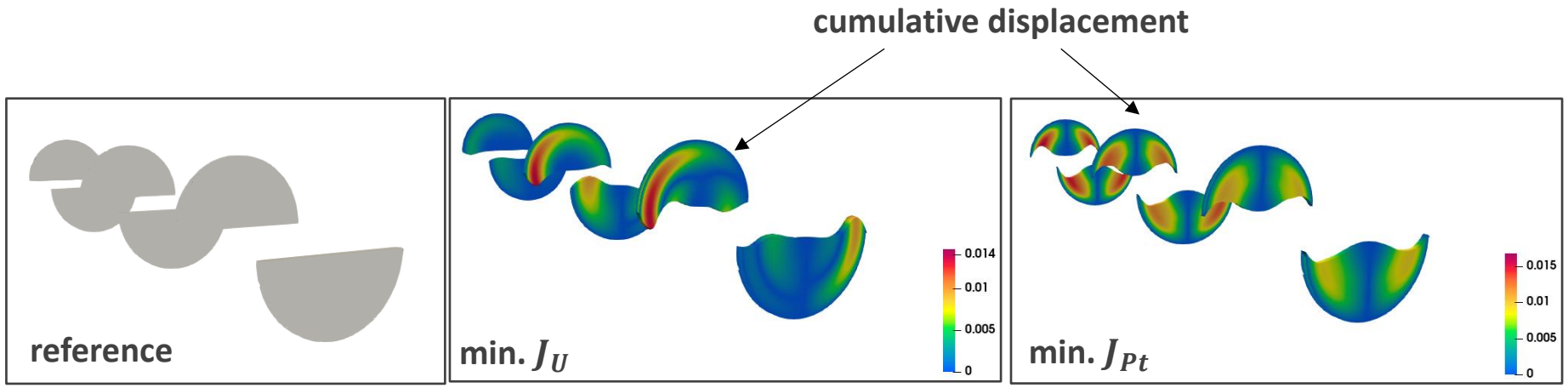
Reference **Optimised for min. J_P**

Reference



Optimised for min. J_P

The two Extreme Points on the Pareto Front



min. J_U	min. J_{Pt}
-94%	-34%

Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

- **Same physical model with two miscible fluids**
- **Min. Total Pressure Losses:**

$$F_{Pt} = -\int_{S_{I,O}} \left(p + \frac{1}{2}\rho v^2\right) v_i n_i dS$$

- **Max. Mixture Uniformity:**

$$F_u = \int_{S_O} \frac{1}{2}(a - \bar{a})^2 v_i n_i dS$$

- **Single Objective:**

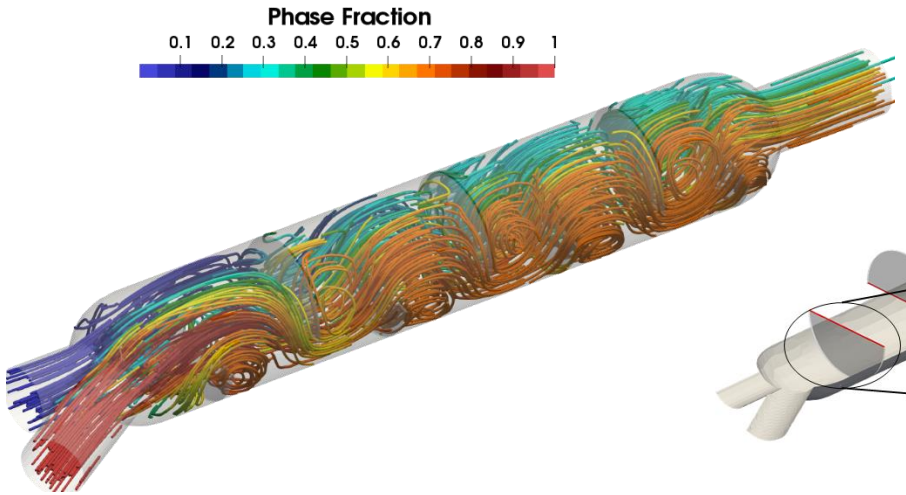
$$F = w_1 F_U + w_2 F_P$$

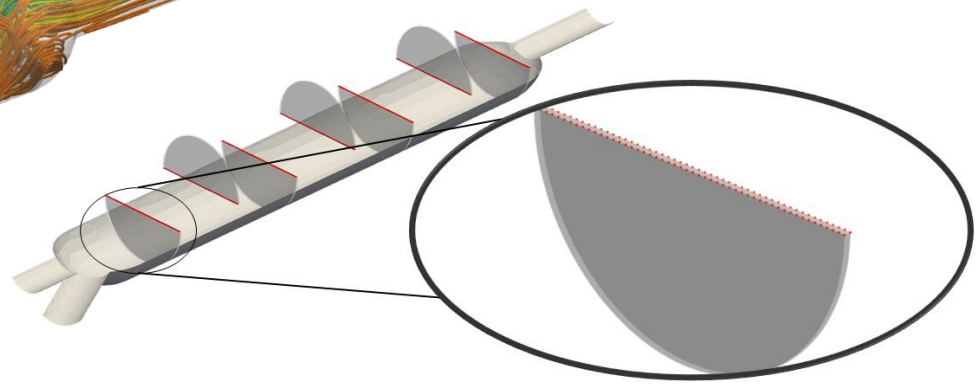
- **6 Sets of weights :**

w1	0	0.25	0.5	0.75	0.9	1
w2	1	0.75	0.5	0.25	0.1	0

Why Nodal Parametrisation?

	Fluid n.1	Fluid n.2
Density (Kg/m^3)	1500	1300
Kinematic Viscosity (m^2/s)	1.5^{-5}	1.3^{-5}
Mass Flow Rate (Kg/s)	0.29	0.26



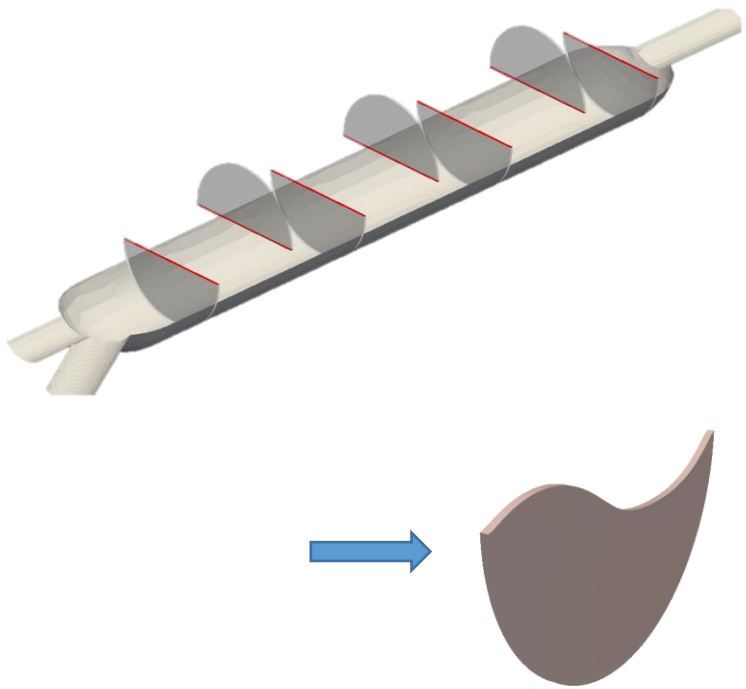


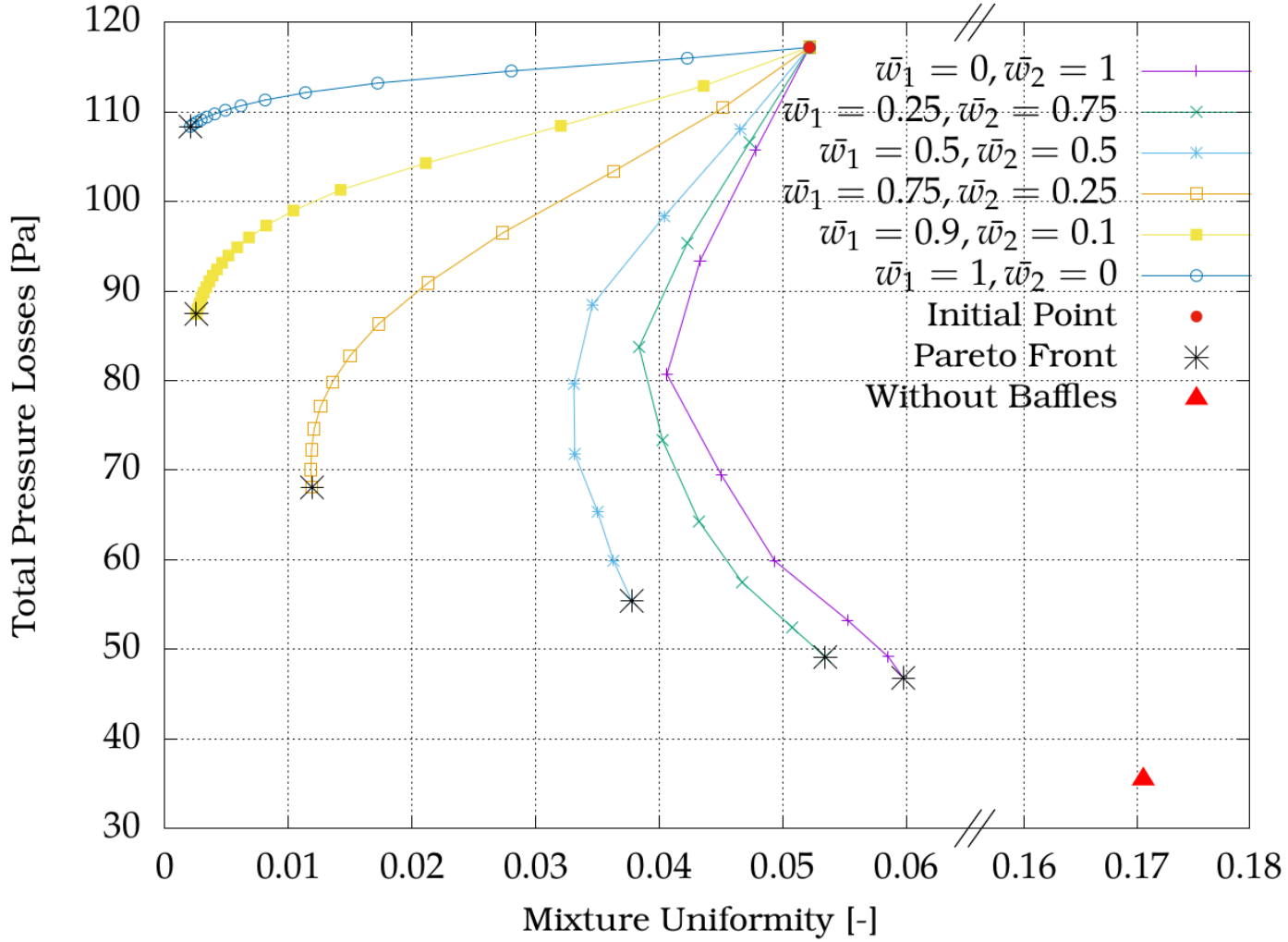
Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

Attempt 1: Node-Based Approach

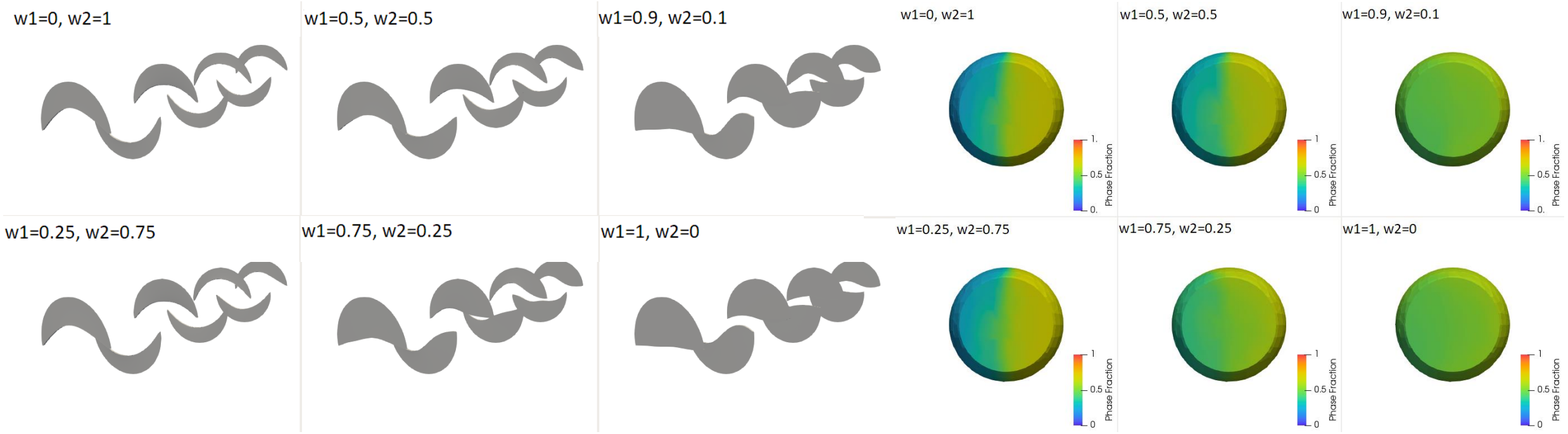
Only the points from the top part act as design variables.

Maintain the flatness and thickness (“In plane”)





Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

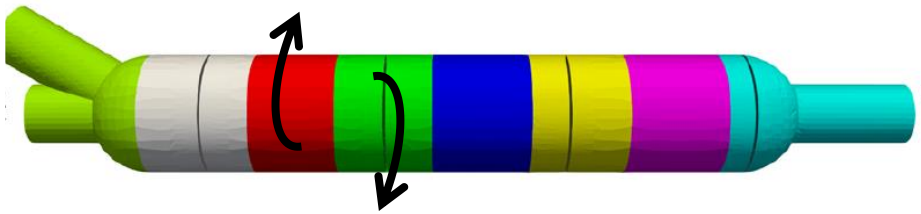


Same conclusions as before!

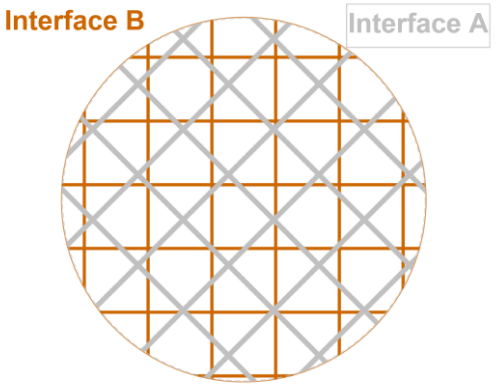
Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

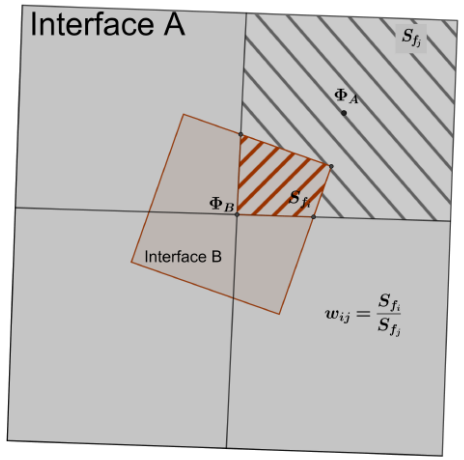
Attempt 2: Positional Angle Optimisation

- The angular positions of the baffles across the mixer are used as design variables.
- Baffles can be placed at different angles inside the mixer without changing either their shapes or longitudinal positions.
- Split into different regions



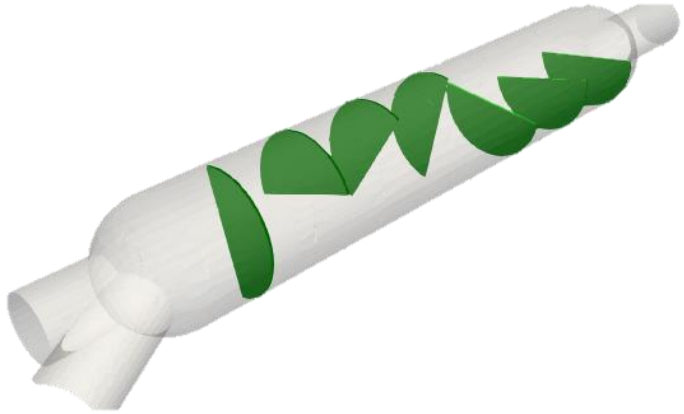
- For each baffle
  - $\tau_b = \sum_j \left(r_j \times G_j n_j \right) \cdot z$
  - $\delta\theta_b = H^{-1} \cdot \tau_b$
  - Extra mathematical development is needed!



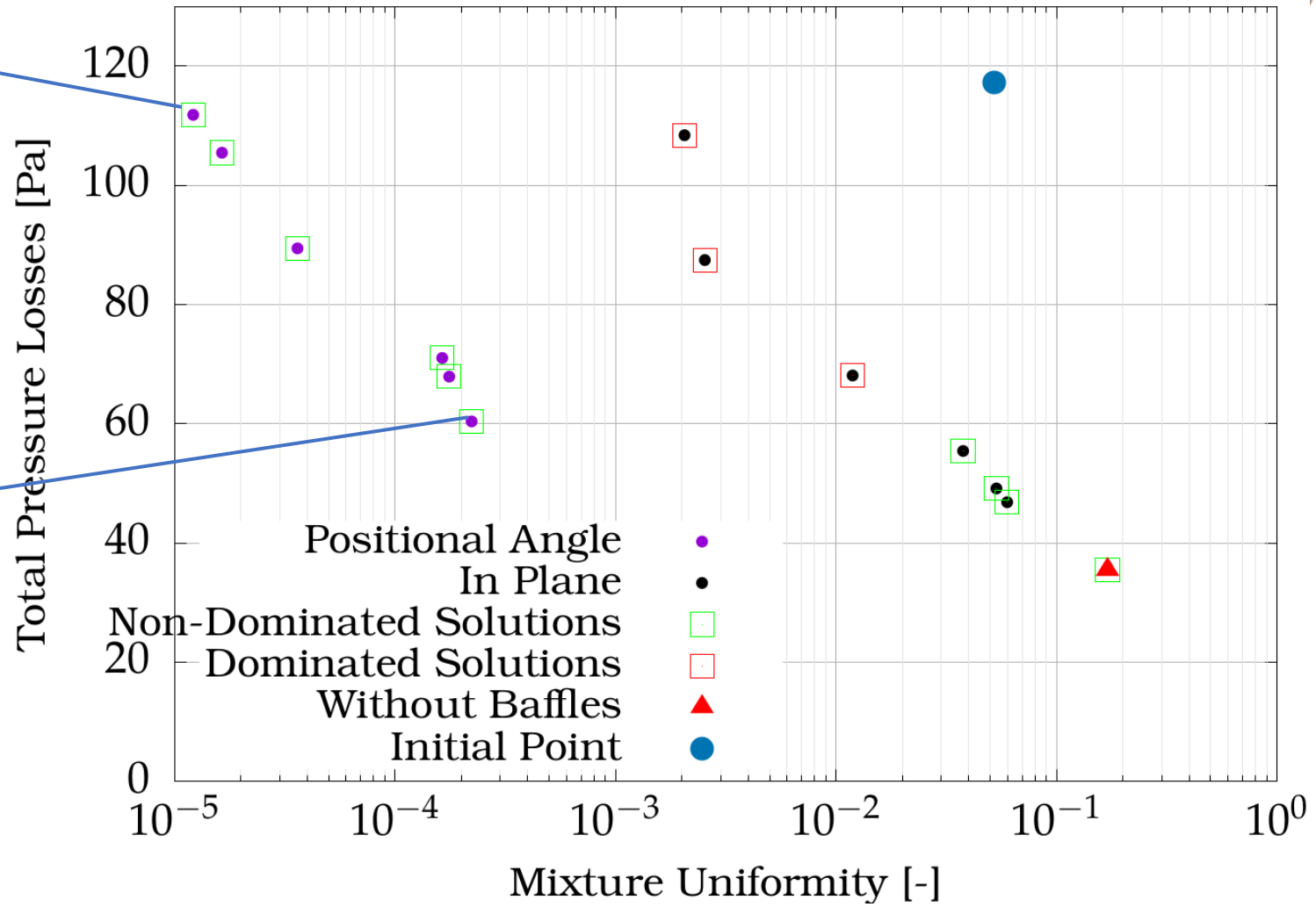


Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

“Best” Mixture Uniformity



“Best” Total Pressure Losses

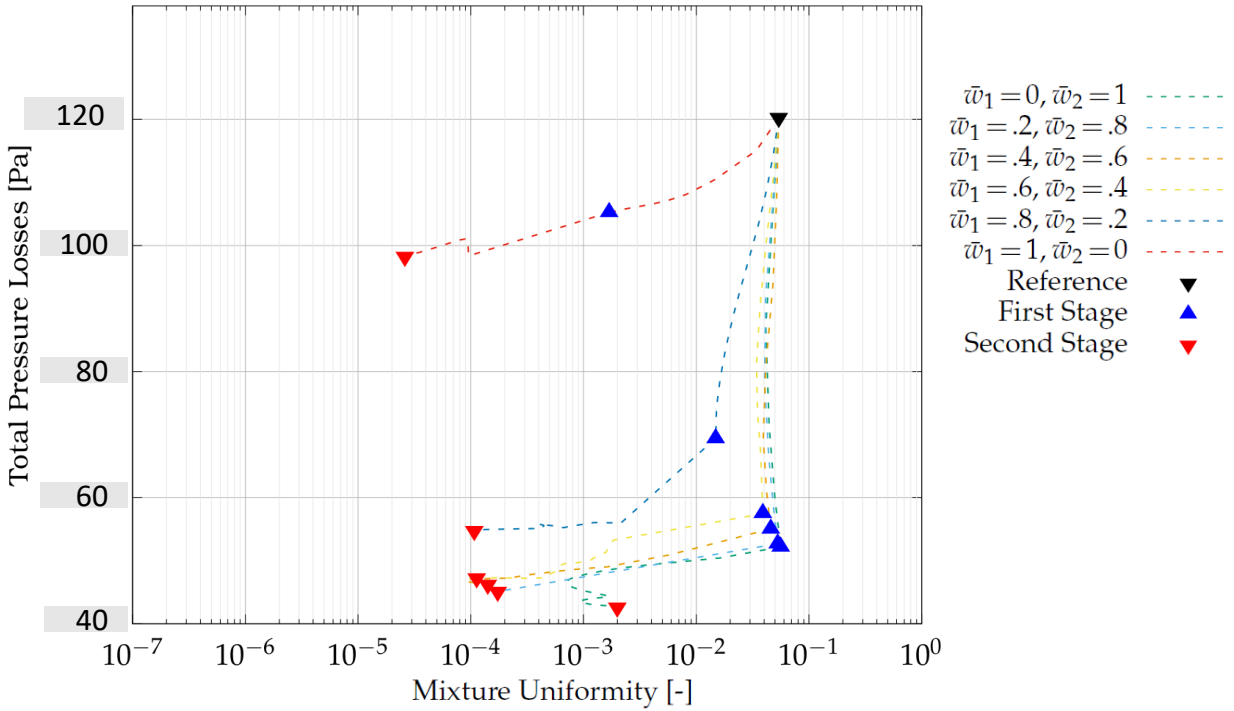


Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

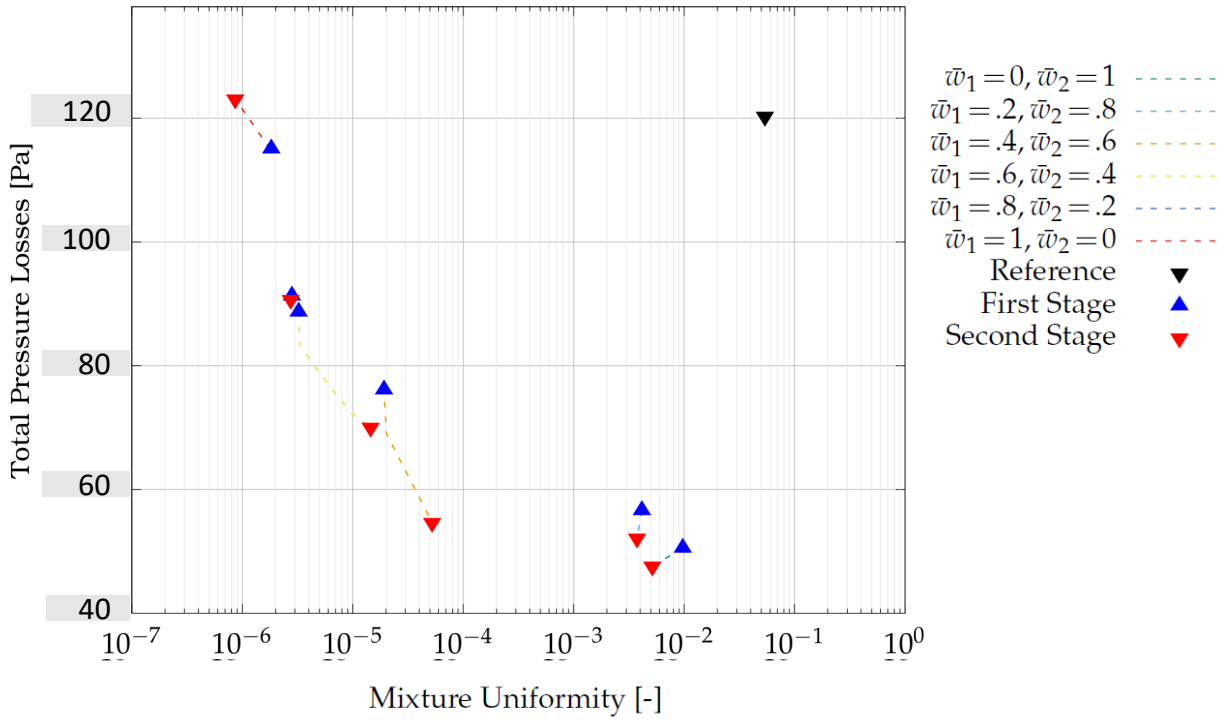
Attempt 3: Combine Node-Based with Positional-Angle Optimisation

In which sequence?

- **Node-based -> Positional-Angle (Scenario 1)**



- **Positional-Angle -> Node-Based (Scenario 2)**

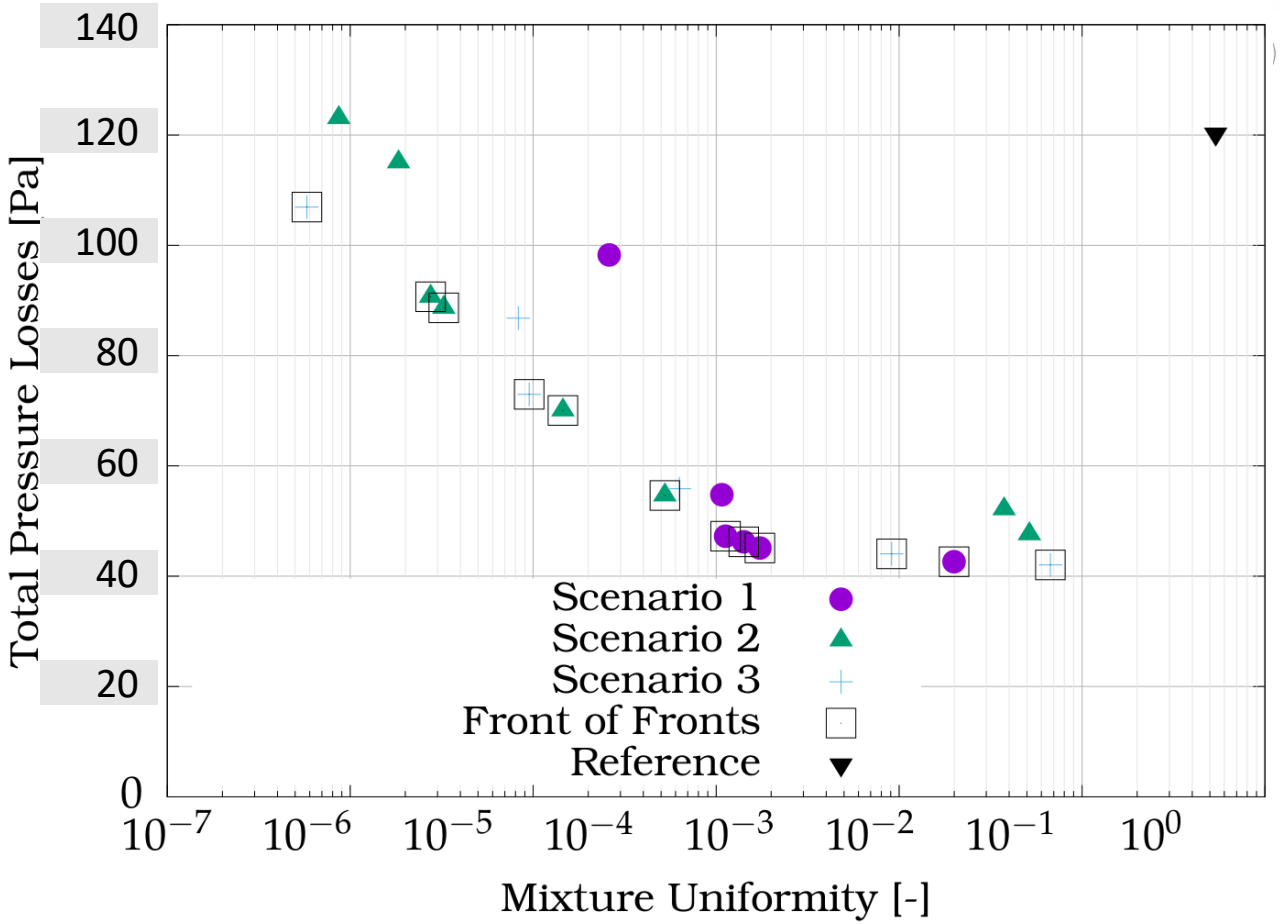


Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

- **Simultaneously changing Positional-Angle + Node-based (Scenario 3)**

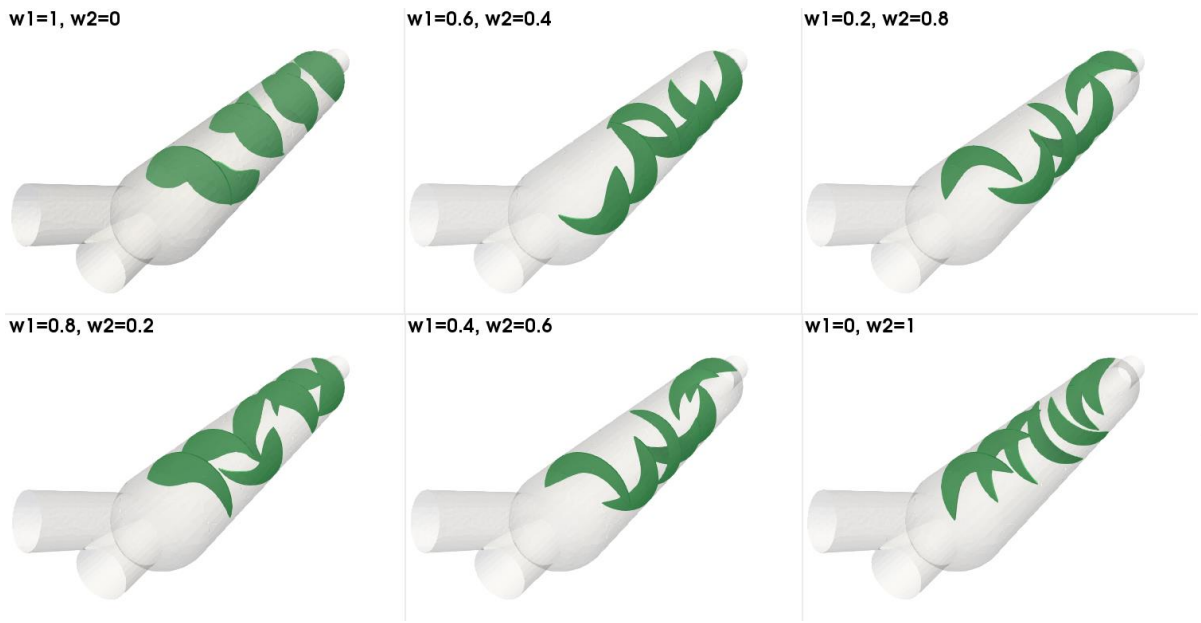
- **Scenario 1: Better results for total pressure losses**
- **Scenario 2: Better results for mixture uniformity**
- **Scenario 3: Well-spread front of non-dominated solutions**

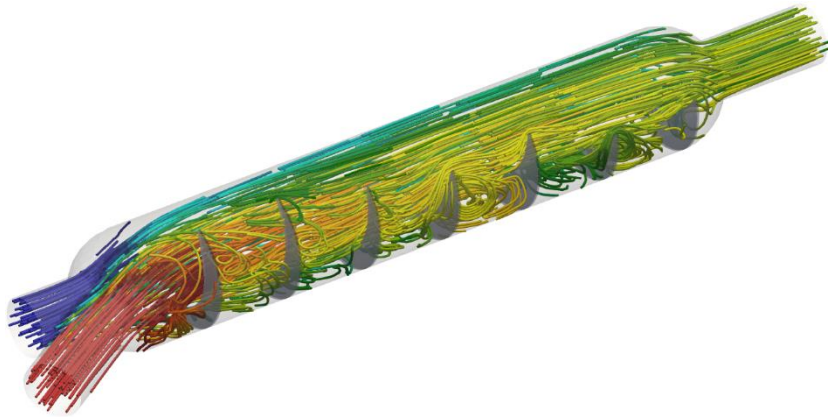
Optimising by controlling all design variables at the same time is usually the most effective way to solve the problem. However, this might need extra development!

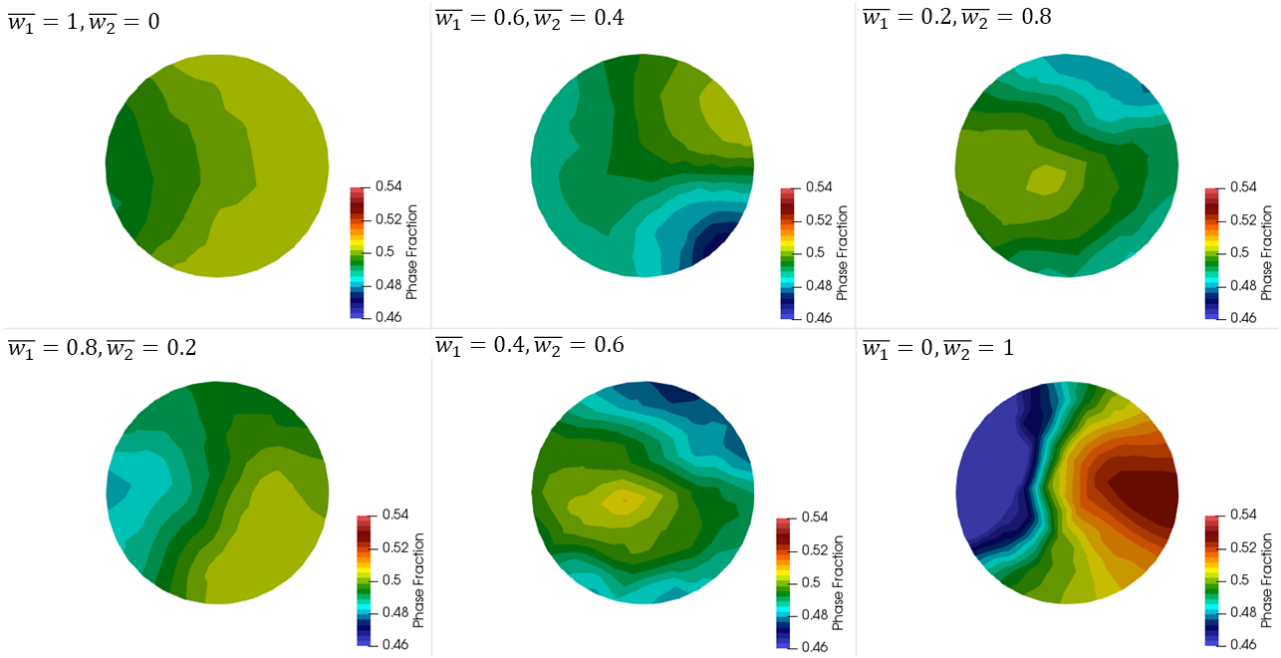


Fronts of non-dominated solutions for all the optimisation scenarios. The final front of non-dominated solutions (empty squares) from all optimisations ("Front of Fronts") as well as the reference configuration are included.

Adjoint-based ShpO of a Mixer using Nodal Parametrisation - Revisited

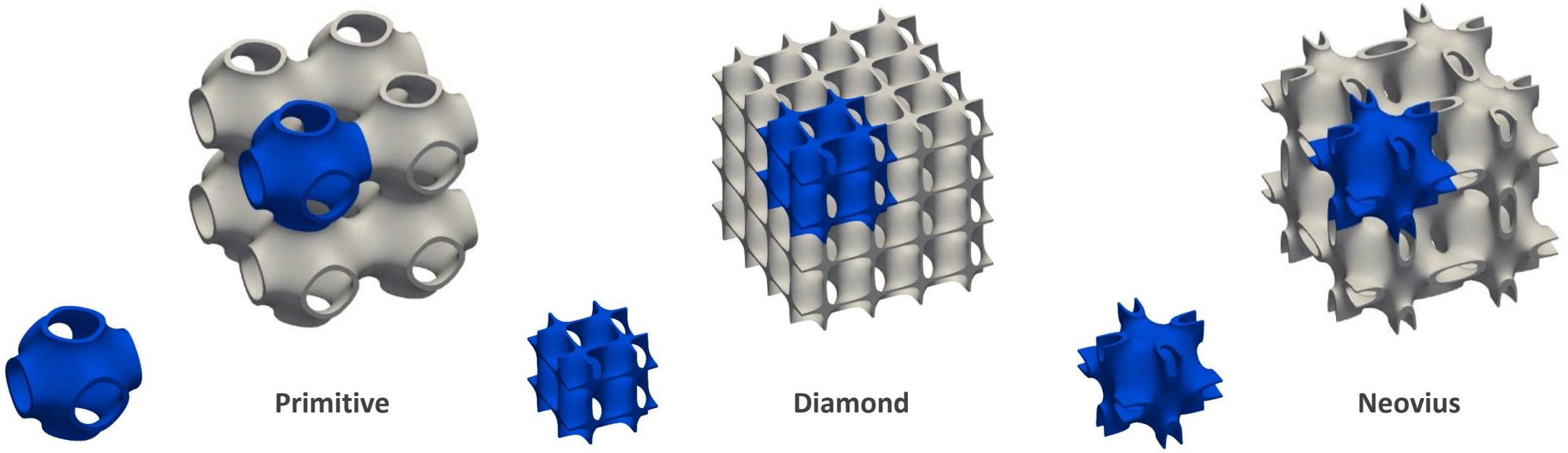






Recent Relevant Research….

Triple Periodic Minimal Surfaces - Gyroids



Primitive

Diamond

Neovius

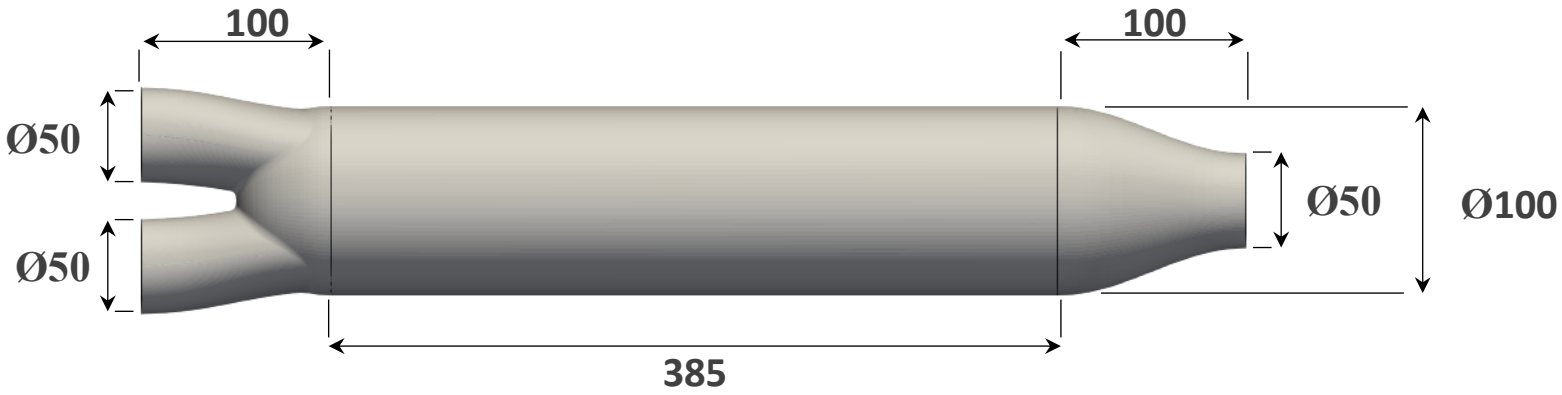
Complex structured, repeated in space. Labyrinthine geometry. High surface/volume ratio.

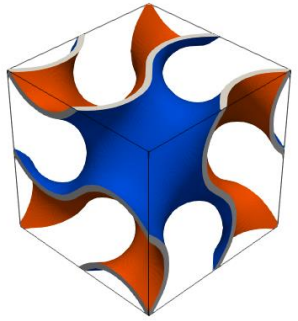
$$\sin(X)\cos(Y) + \sin(Y)\cos(Z) + \sin(Z)\cos(X) = 0$$

where $X = 2\alpha\pi x, Y = 2\beta\pi y, Z = 2\gamma\pi z$ με $x, y, z \in [0, 1]$.

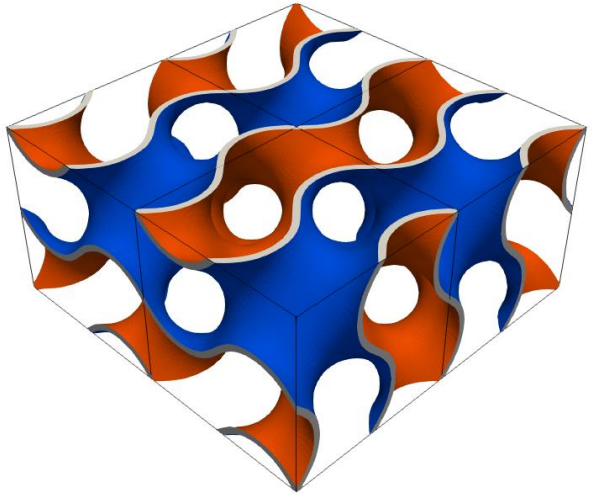
$\alpha, \beta, \gamma > 0$

Recent Relevant Research….

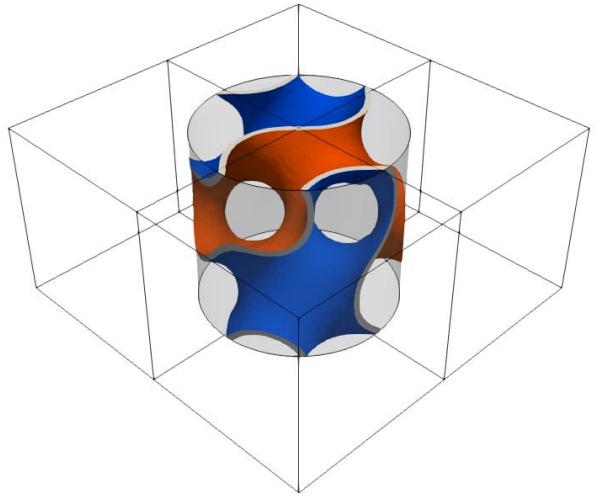




Basic Building Block

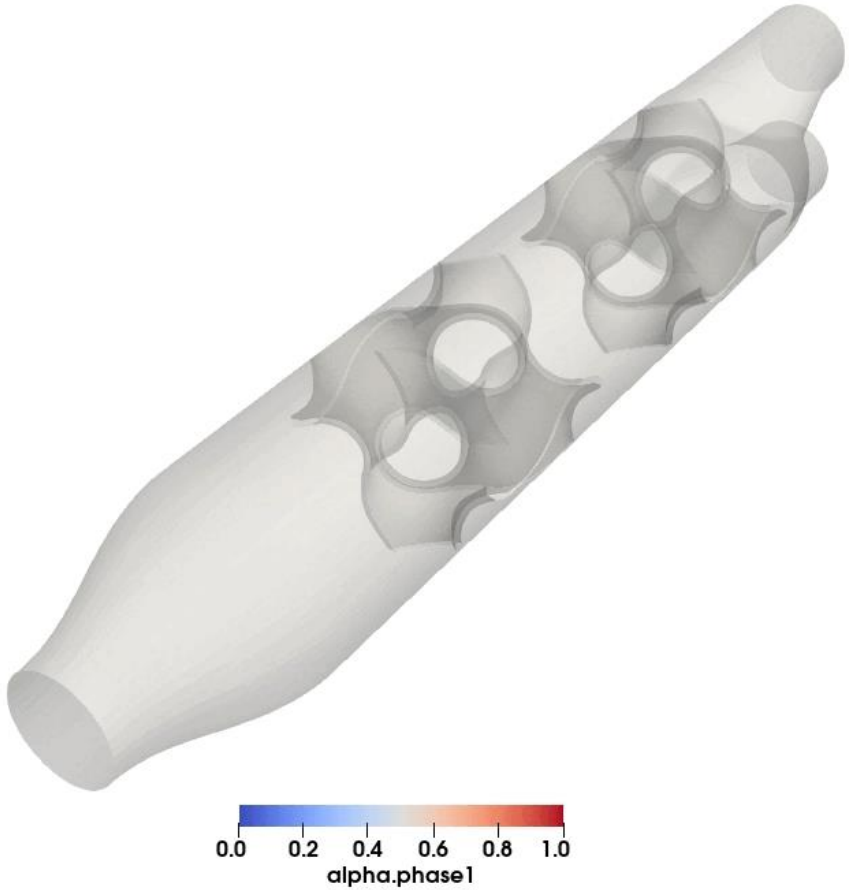


4 Building Blocks

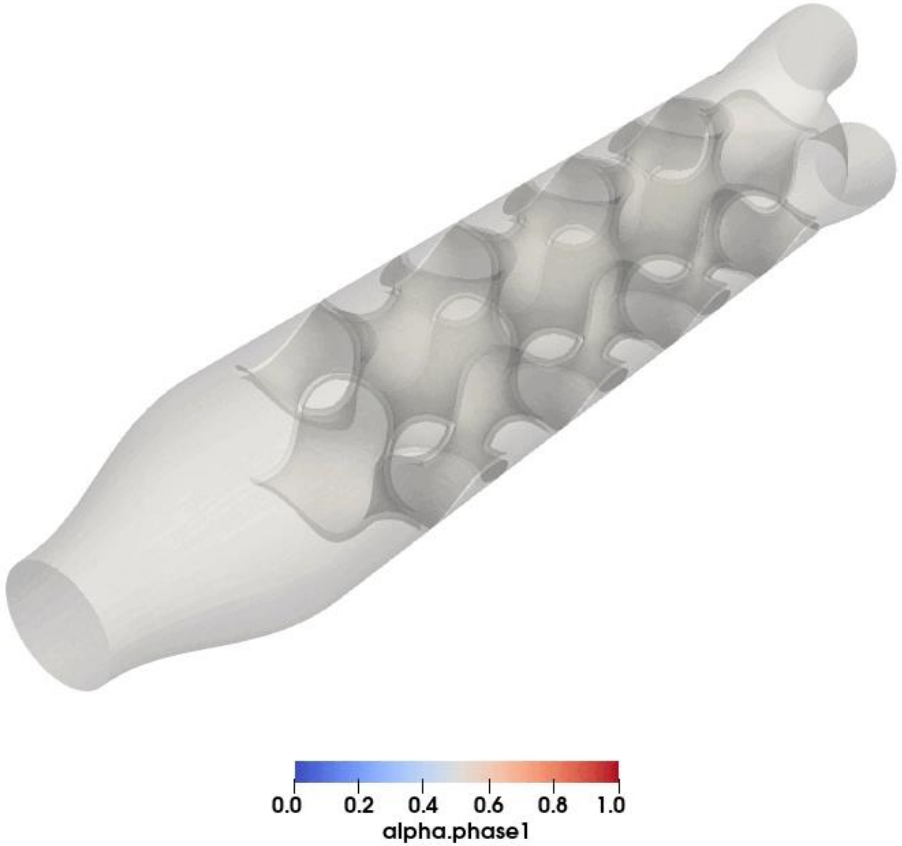


Gyroid into the Mixer

Recent Relevant Research….

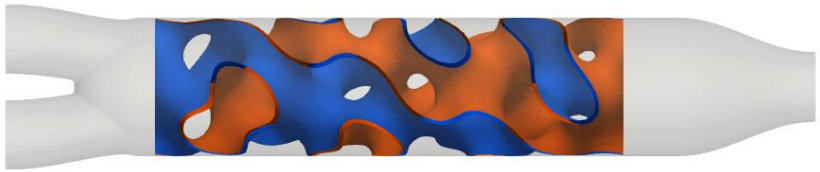


Recent Relevant Research….



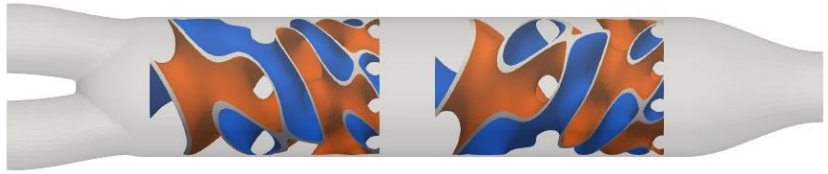
Recent Relevant Research….

- Case 2, ~66Pa, CoV=0.114

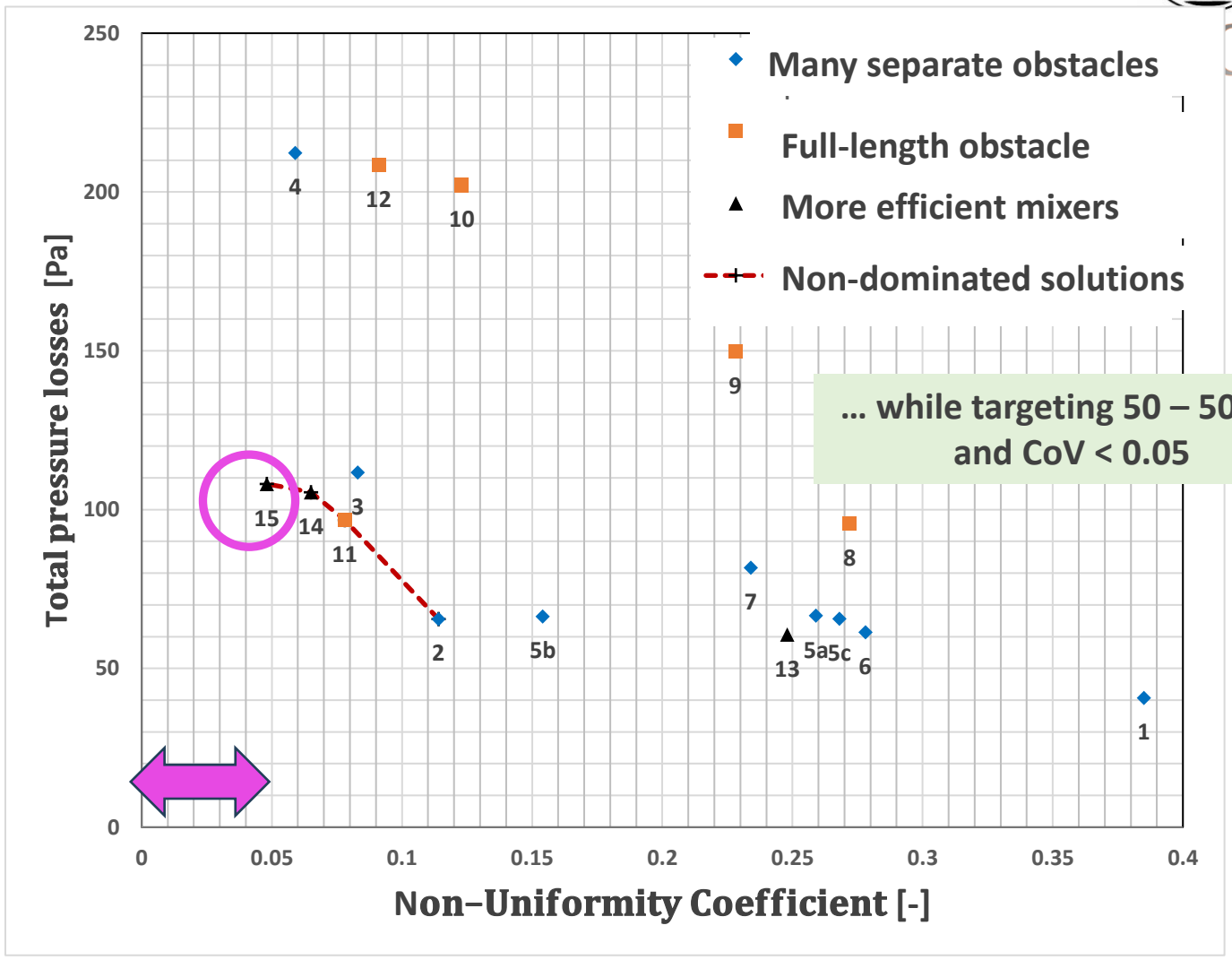


- Case 11, ~97 Pa, Cov=0.078

- Case 14, ~106 Pa, CoV=0.065



- Case 15, ~108 Pa, CoV=0.048



… while targeting 50 – 500 Pa and CoV < 0.05