Industrial partners:

University partners:

National Technical University of Athens

Node-based vs CAD-based Approach in CFD Adjoint-based Shape Optimisation

Mateusz Gugala

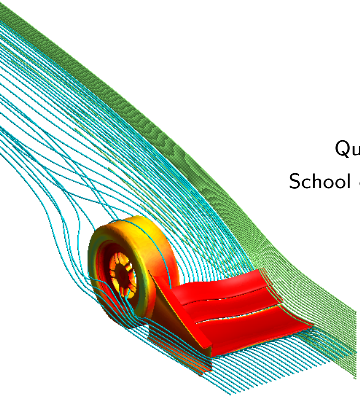
Shenren Xu

Jens-Dominik Müller

Queen Mary University of London

School of Engineering and Material Science

June 4, 2014



Research funded by the European Commission

Contents

Motivation

Parameterisation method is an important part of successful optimisation. There are several techniques that can be used:

- Node-based parameterisation

  Mesh node coordinates as a design variables, reach design space, smoothing required.

- CAD-based parameterisation

  CAD parameters as design variables e.g. coordinates of NURBS control points, access to CAD system required in order to get partial derivatives related to parameterisation.

- FFD - Free Form Deformation
- RBF morphing

This presentation is focused on comparing two methods:
Cad-based and Node-based parameterisation approaches.

The flow (Primal) and adjoint (Dual) solver

Flow system solve:

$$R(U, X) = 0 \qquad (1)$$

Adjoint system solve:

$$A^T v = g \;\Leftrightarrow\; \left(\frac{\partial R}{\partial U}\right)^T \left(\frac{\partial L}{\partial R}\right)^T = \left(\frac{\partial L}{\partial U}\right)^T \qquad (2)$$

1. Discrete Adjoint solver, semi-automatic
2. Derived using pseudo-time-stepping framework adopted from flow solver and its selectively differentiated parts/subroutines (with the use of *Tapenade*[1])

A - system Jacobian, v - adjoint variable, g - adjoint source term,
L - cost/objective function, U - vector of flow variables, X - mesh coordinates (x,y,z),
R - vector of residuals for each equation out of total 5(inviscid case) / 6(viscous case)

[1] AD tool developed at Inria http://www-sop.inria.fr/tropics/

Sensitivity evaluation

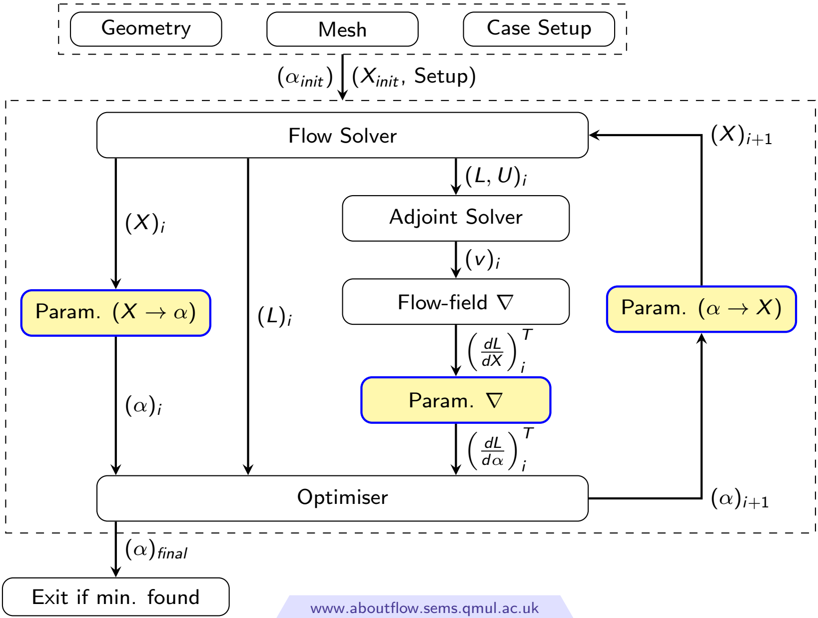
1. Derive the formula for sensitivity of cost function 'L' wrt. design variable 'α' (3)
2. Differentiate parts of code to get each partial derivative present in chain rule (3) (use *Tapenade* AD tool)
3. Assemble sensitivity based on chain rule using hand written code and computed partial derivatives

$$\overset{Sensitivity\ \ accumulation\ \ order}{\longleftarrow} \qquad \frac{dL}{d\alpha}^{T} = \underbrace{\left(\frac{\partial X}{\partial \alpha}^{T}\right)}_{Parameterisation\ \ gradient} \underbrace{\left(\frac{\partial L}{\partial X}^{T} - \frac{\partial R}{\partial X}^{T}\frac{\partial L}{\partial R}^{T}\right)}_{Flow-field\ \ gradient} \tag{3}$$

Parameterisation gradient - partial derivative obtained by differentiating parameterisation module

Flow-field gradient - the derivative obtained by differentiating flow solver.

Optimisation flowchart



Parameterisation Methods

Shape parametrisation techniques

1. Node-based

   Good for initial investigations, easy to set up, not much user input required / constraint limitations, requires transferring the result back to CAD.

2. CAD-based

   Good for industrial design practice, more sophisticated constraints, no back transfer to CAD needed / more tedious to set up, time required to develop CAD parameterisation tools.

3. Free-form deformation (FFD)
4. RBF morphing

Is any of these methods best... ?

CAD-based parametrisation

1. The design variable α is e.g. control point CP of B-spline/NURBS
2. Only boundary representation of design surface is required
3. CAD module developed at QMUL[2] is used to match mesh nodes of design surface with CP parameter, i.e. $X_\Delta \rightarrow CP$. Several constraint approaches available including G1 and G0 continuity.
4. The spring-based algorithm is used to transfer from Volume mesh nodes to design surface mesh nodes, i.e. $X \rightarrow X_\Delta$

The final Sensitivity for CAD-based parameterisation:

$$\frac{dL}{d\alpha}^T = \underbrace{\left(\frac{\partial X_\Delta}{\partial CP}^T \frac{\partial X}{\partial X_\Delta}^T \right)}_{Parameterisation\ \ gradient} \underbrace{\left(\frac{\partial L}{\partial X}^T - \frac{\partial R}{\partial X}^T \frac{\partial L}{\partial R}^T \right)}_{Flow-field\ \ gradient} \quad (4)$$

[2] Xu, S., Jahn, W. and Müller, J.-D. (2014), CAD-based shape optimisation with CFD using a discrete adjoint. Int. J. Numer. Meth. Fluids, 74: 153â

Node-based parametrisation

1. The design variable α is coordinate of each design surface node i.e. X_Δ
2. Explicit Laplacian smoothing is used to filter high frequency shape modes - transfers from design surface mesh nodes to smoothed design surface mesh nodes, i.e. $X_\Delta \rightarrow X_S$. Remark: the smoothed quantity is δX_Δ (perturbation field) not X_Δ.
3. The spring-based algorithm is used to transfer from smoothed design surf. nodes to volume nodes, i.e. $X_S \rightarrow X$

The final Sensitivity for Node-based parameterisation:

$$\frac{dL}{dX_\Delta}^T = \underbrace{\left(\frac{\partial X_S}{\partial X_\Delta}^T \frac{\partial X}{\partial X_S}^T\right)}_{Parameterisation\ \ gradient} \underbrace{\left(\frac{\partial L}{\partial X}^T - \frac{\partial R}{\partial X}^T \frac{\partial L}{\partial R}^T\right)}_{Flow-field\ \ gradient} \tag{5}$$

Laplacian smoothing

- Based on diffusion equation (for scalar ϕ):

$$\frac{\partial \phi}{\partial t} = \Delta \phi \tag{6}$$

- Can be discretised as:

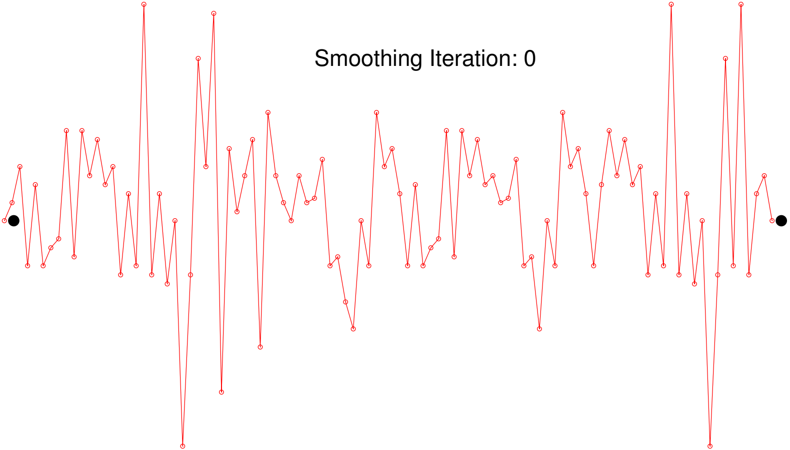
$$\phi_i^{n+1} = \phi_i^n + \frac{\triangle t}{\triangle x^2} \left(\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n\right) \tag{7}$$

- The general formula for explicit Laplacian smoothing of perturbation field δX can be written:

$$\delta X_i^{n+1} = \delta X_i^n + \beta \frac{\sum_{j=1}^{m} \left(\delta X_j^n - \delta X_i^n\right)}{m} \tag{8}$$

m - total number of edges connected to node 'i', - Constant (0.5 for most effective explicit smoothing)

Simple example



● - fixed points (constraints)

Node vs CAD example

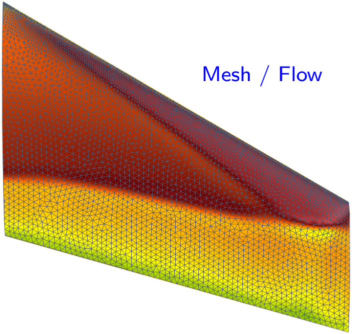
Case setup - flow

Boundary conditions / Solver setup

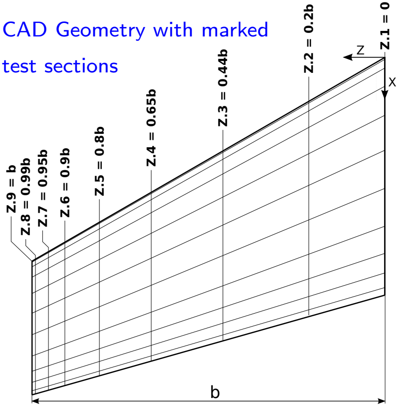
- Case: Onera M6 wing transonic Euler flow
- Boundary Conditions: Ma = 0.84, AoA = 3.06
- Solver: JT-KIRK solver, Roe flux, 2^{nd} O
- Mesh: Tet, ∼100K nodes

CAD Geometry with marked test sections

Mesh / Flow

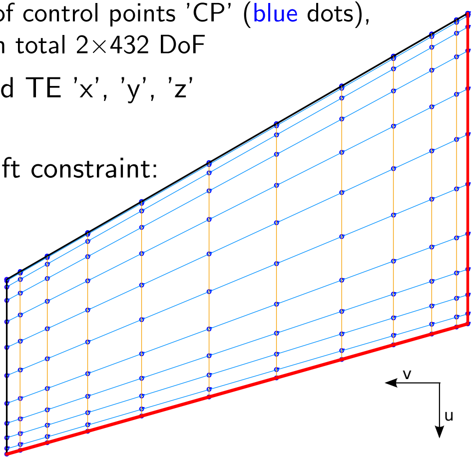


⟸



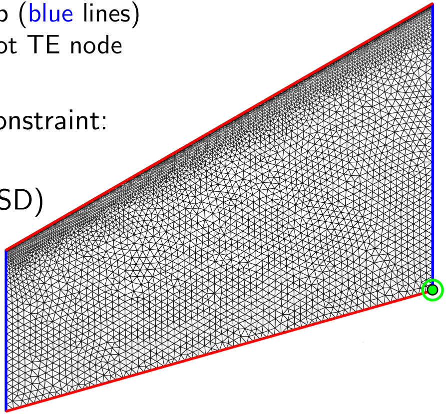
CAD-based (C-B) optimisation setup

- Design variable (α):
  - wing defined as two separate top and bottom NURBS patches (polynomial degree: 5)
  - 'x', 'y', 'z' coordinates of control points 'CP' (blue dots), 12x12 points ($u \times v$), in total 2×432 DoF
- Constraints - fixed root and TE 'x', 'y', 'z' coordinates (red lines)
- Cost function: drag with lift constraint: $L = C_D + 4(C_L - C_{L0})^2$
- Optimiser: L-BFGS(7)



Node-based (N-B) optimisation setup

- Design variable (α): Mesh nodes ($\sim$26K DoF) of top and bottom wing surface
- Constraints
  - 'x' and 'z' coordinates on LE and TE fixed (red lines)
  - 'z' coordinate of root and tip (blue lines)
  - 'x', 'y', 'z' coordinates of root TE node (green point)
- Cost function: drag with lift constraint: $L = C_D + 4(C_L - C_{L0})^2$
- Optimiser: Steepest Descent (SD) with Armijo Line Search (LS).
- Explicit Laplacian smoothing applied to perturbation field (20 iterations).



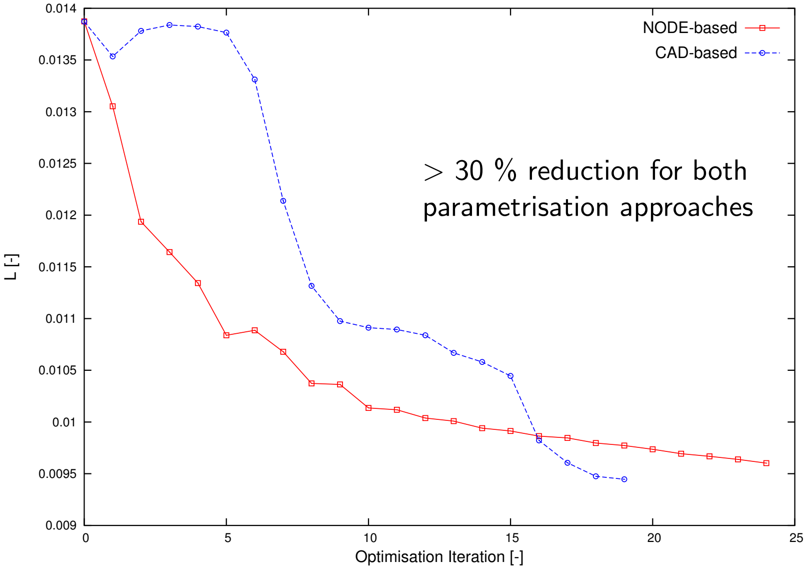
Setup discrepancies

There are several differences in setup between CAD-based and Node-based optimisation case:

- Different mesh sizes - 70k(C-B) vs 100k(N-B) mesh nodes
- Differences in constraints - Fixed root and TE, other free to move (C-B) vs Planform (N-B)
- Different optimisers

Work towards more rigorous comparison in progress...

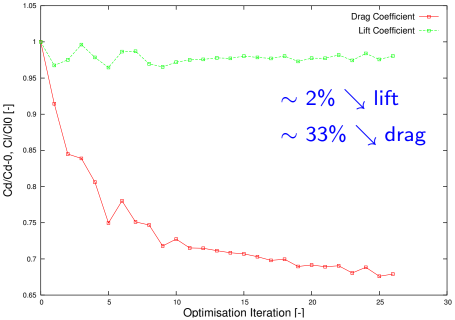
Cost function convergence



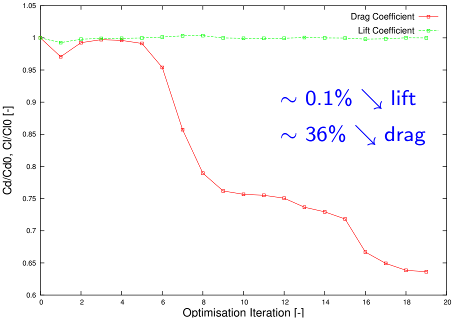
Drag and lift convergence

- Significant reduction of drag with almost unchanged lift
- More optimisation steps to be run / use L-BFGS for both cases (these are intermediate results)

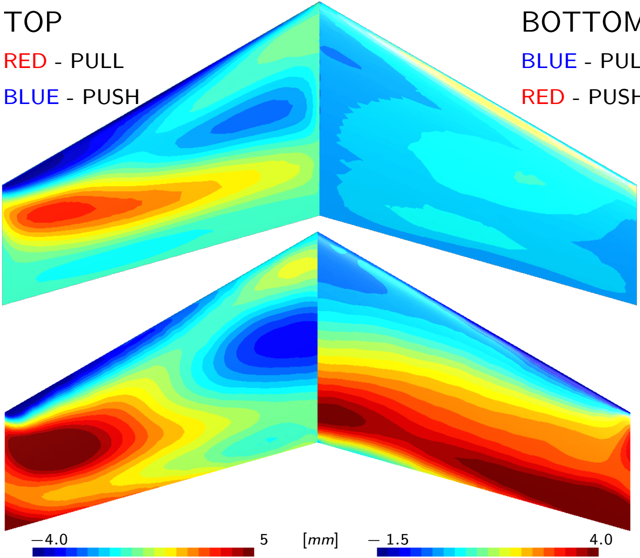
Node-based, SD w. Armijo LS



CAD-based, L-BFGS(7)



Total shape Y-displacement

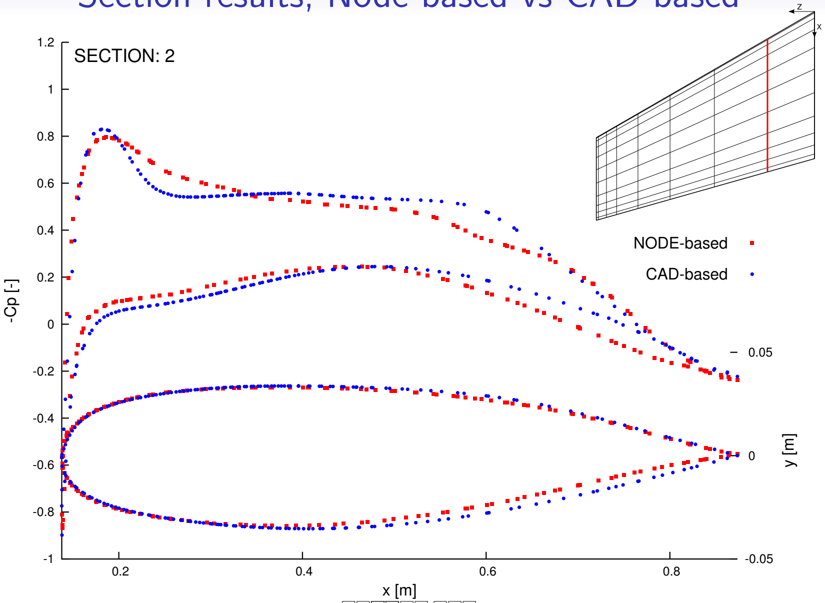


'x', 'z' displacements are order of magnitude lower

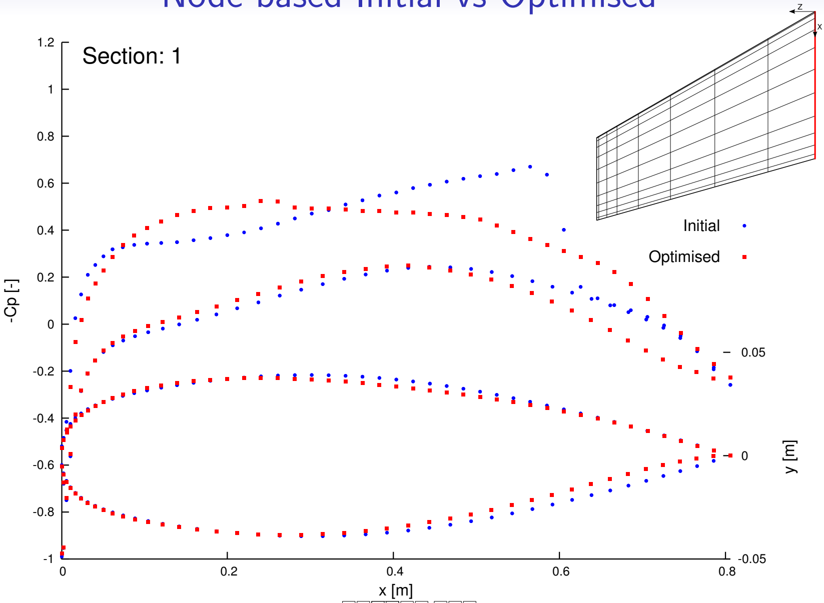
← CAD-based

← Node-based

Section results, Node-based vs CAD-based



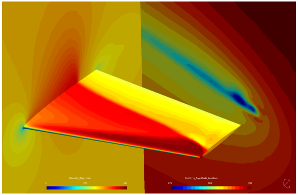
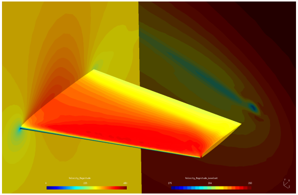
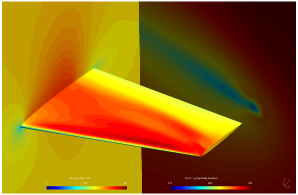
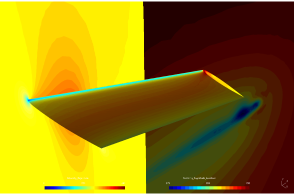
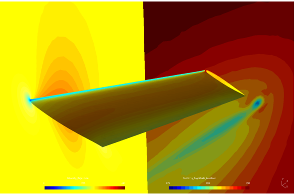
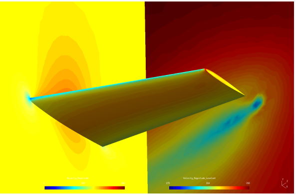
Node-based Initial vs Optimised



Velocity Base/Node/CAD

Smeared shocks / Weaker wake for optimised solution

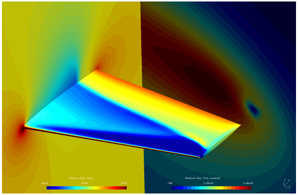
Base | Node-based | CAD-based

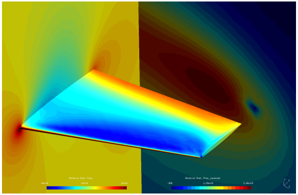
Pressure Base/Node/CAD

Reduced wing tip vortex intensity

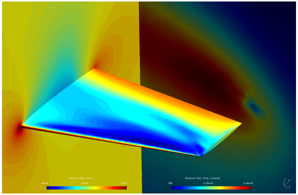
Base

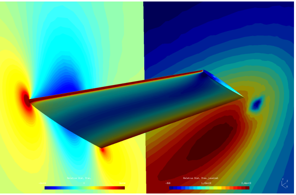


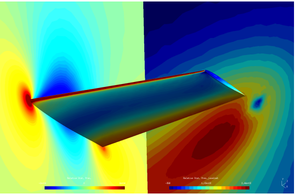
Node-based

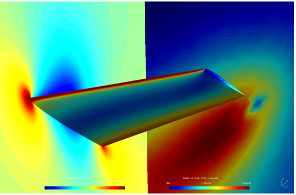


CAD-based









Summary

Conclusion / Future work

Summary:

1. Both parametrisation methods works well for test case - similar improvements obtained.
2. As expected, the total shape displacement maps are similar for top wing surface apart from the root and TE region - due to differences in constraints.
3. There are discrepancies in total shape displacement for bottom wing surface.
4. The choice of parametrisation method depends on the application, design stage, constraints complexity.

Future/Current work:

1. Perform more rigorous comparison, re-run cases with same constraints, use same mesh.

Industrial partners:

University partners:

National Technical University of Athens

Acknowledgement

"This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no [317006]".

Research funded by the European Commission

Node-based optimisation steps