Numerical study of the airflow over a high-altitude pseudo-satellite wing

PhD update

Carlo Brunelli

October 17, 2023

PhD presentation after 1.5 years

► Numerical Methods

► Performances

► Synthetic Eddy Method

► Laminar Separation Bubble

► Other activities

► Next steps

The numerical method currently implemented are:

- Variational Multiscale Method (VMS)
- Streamline upwind Petrov–Galerkin (SUPG)

Different solution method are available for all of them

- Non-linear (NLIN)
- Linearized-Coupled (LC-VMS)
- Linearized-Segregated (LS-VMS)

Variational Multiscale Method

1 Numerical Methods

- Evolution of the SUPG
- Implict LES
- It does not need calibration
- Residual-based stabilization

Galkerkin formulation

Conservation of mass:

$$\nabla \cdot \vec{u} = 0 \tag{1}$$

Conservation of momentum:

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u} + \nabla p - \nu \Delta \vec{u} - f = 0 \tag{2}$$

Variational Formulation

$$B^G = \int_\Omega \frac{\partial \vec{u}}{\partial t} \cdot \vec{v}\, d\Omega + \int_\Omega (\vec{u} \cdot \nabla)\vec{u} \cdot \vec{v}\, d\Omega + \int_\Omega \nabla(p) \cdot \vec{v}\, d\Omega +$$
$$\int_\Omega \nu \nabla \vec{u} \cdot \nabla \vec{v}\, d\Omega - \int_\Omega f \cdot \vec{v}\, d\Omega + \int_\Omega q(\nabla \cdot \vec{u})\, d\Omega = 0 \tag{3}$$

Stabilization equations

$$B^{SUPG}(t,(\vec{u},p),(\vec{v},q)) = \int_\Omega (\tau_m(\vec{u}\cdot\nabla\vec{v}+\nabla q)\cdot\vec{R_m}\ d\Omega + \int_\Omega \tau_c(\nabla\cdot\vec{v})R_c\ d\Omega \qquad (4)$$

$$B^{VMS1}(t,(\vec{u},p),(\vec{v},q)) = \int_\Omega (\vec{u}\cdot\nabla\vec{v}')\odot(\tau_m\vec{R_m})\ d\Omega \qquad (5)$$

$$B^{VMS2}(t,(\vec{u},p),(\vec{v},q)) = -\int_\Omega (\nabla\vec{v}\odot(\tau_m\vec{R_m}\otimes\tau_m\vec{R_m})\ d\Omega \qquad (6)$$

$$\tau_m = \left(\frac{4}{\Delta t^2} + \vec{u} \cdot G\vec{u} + C_I \nu^2 G : G \right)^{-1/2} \tag{7}$$

$$\tau_c = (\tau_c \vec{g} \cdot \vec{g})^{-1} \tag{8}$$

Where G is the inverse of the gradient of the map cell. For a cubed shaped element, with h the edge length, $G_{ij} = \frac{1}{h^2}\delta_{ij}$, where δ_{ij} is the Kronecker delta.

$$\tilde{u} = 2.1875u^n - 2.1875u^{n-1} + 1.3125u^{n-2} - 0.3125u^{n-3} \tag{9}$$

Boundary Layer Initialization

- Avoid Instabilities (close to the leading edge)
- Avoid velocity ramping
- Higher time-step

Using the function exploited by RANS solvers for the wall distance for computing turbulent parameters but now used to detect the airfoil's contour

$$\begin{cases} \nabla \cdot (|\nabla u_p|^{p-2} \nabla u_p) = -1 & x \in \Omega \\ u_p = 0 & x \in \Omega_D \end{cases} \quad (10)$$

The true-wall distance is provided by solving the equation for $p \to \infty$ Solved using a method that resembles Picard's

Boundary Layer Initialization

The velocity in the *x* direction in the region identified can be with a simple cubic function (11) where $dn = d/\delta_{99}$, d is the minimum distance to the airfoil, u_∞ is the free-stream flow speed.

$$f(dn) = \begin{cases} u_\infty & dn > 1 \\ (-dn^2 + 2 \cdot dn) \cdot u_\infty & dn < 1 \end{cases} \tag{11}$$

The boundary layer function has been obtained by fixing the following boundary conditions:

- Continuity with the external flow, $f(1) = 1$
- Smooth transition between boundary layer and external flow, $f'(1) = 0$
- Non-slip condition at the wall, $f(0) = 0$

Boundary Layer Initialization

It results in a low-speed zone close to the airfoil, avoiding high speed in really small cells useful for capturing the boundary layer.
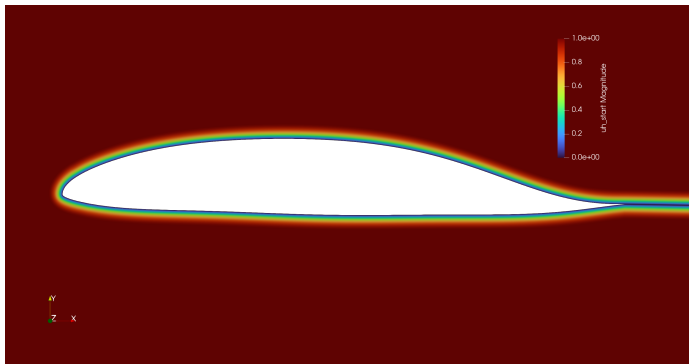
Figure: Boundary Layer Initialization

► Numerical Methods

► Performances

► Synthetic Eddy Method

► Laminar Separation Bubble

► Other activities

► Next steps

Weak Scaling
Strong Scaling

▶ Numerical Methods

▶ Performances

▶ Synthetic Eddy Method

▶ Laminar Separation Bubble

▶ Other activities

▶ Next steps

- publication
- presented at *JuliaCon2023* at MIT

Features:

- Create fluctuations that respect the divergence-free condition (DFSEM)
- Create velocity fluctuations for inlet boundary conditions
- Create coherenteddies in 3D domain
- Define custom Reynolds Stress Tensor
- Import from file custom Reynolds Stress Tensor

Reynolds decomposition:

$$\vec{u}(\vec{x}, t) = \vec{U}(\vec{x}, t) + \vec{u'}(\vec{x}, t) \tag{12}$$

Compute velocity fluctuations, using a suitable shape function:

$$u_i(\mathbf{x}) = U_i(\mathbf{x}) + \frac{1}{\sqrt{N}} \sum_{k=1}^{N} a_{ij} \epsilon_j^k f_{\sigma(\mathbf{x})} \left(\mathbf{x} - \mathbf{x}^k \right) \tag{13}$$

▶ Numerical Methods

▶ Performances

▶ Synthetic Eddy Method

▶ Laminar Separation Bubble

▶ Other activities

▶ Next steps

- VMS linearized coupled sd7003s - Re $60\,000$ - AoA $4°$
- VMS linearized segregated DU89 - Re $250\,000 - 500\,000$ - AoA $1° - 5°$

VMS linearized coupled sd7003s

4 Laminar Separation Bubble

Copuled: velocity and pressure are solved at the same time. Re 60 000 - AoA 4° .
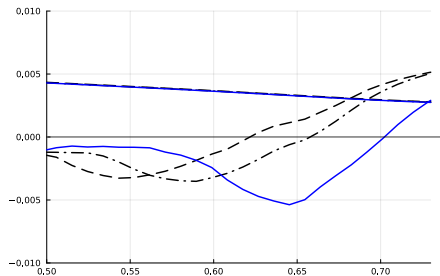Initialization with velocity-ramping
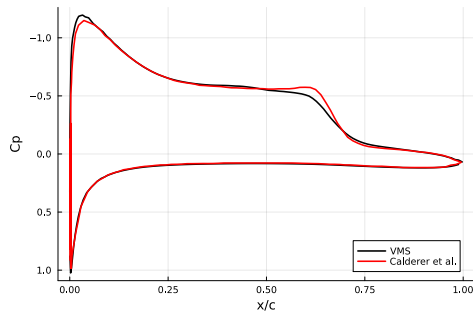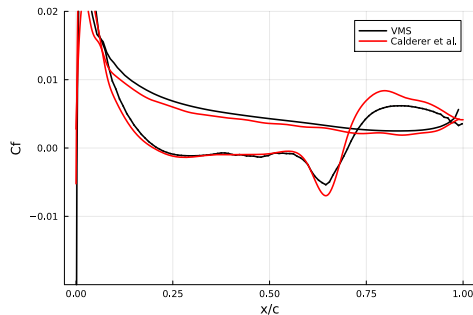
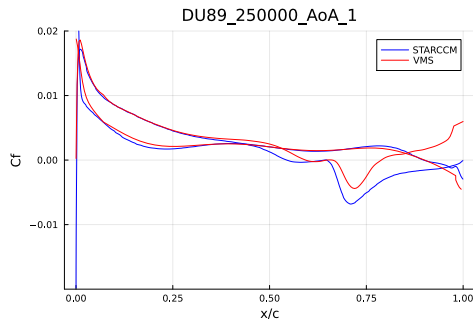Figure: Comparison with VMS literature results

VMS linearized coupled sd7003s

4 Laminar Separation Bubble
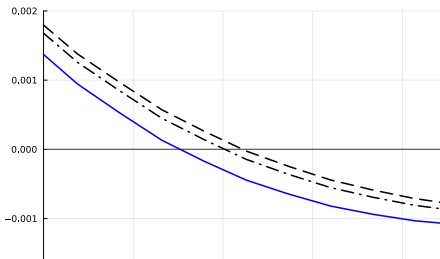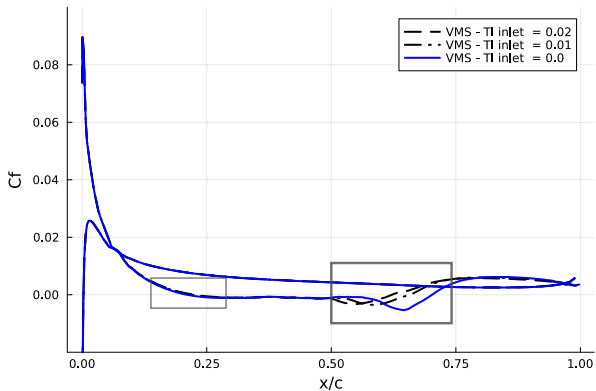
VMS Linearized-Segregated
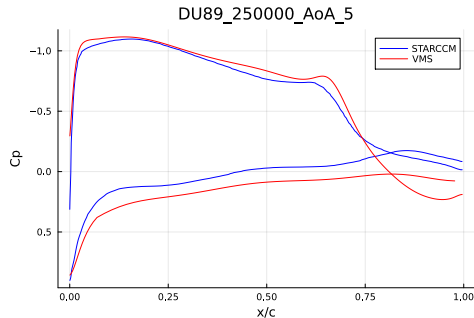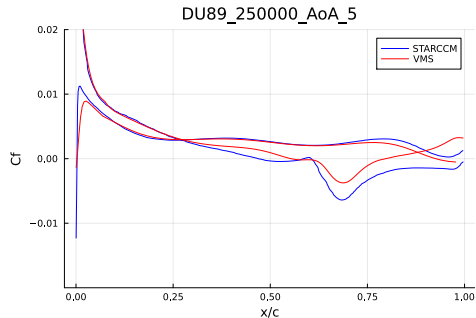
4 Laminar Separation Bubble

Segregated: each time step pressure and velocity sistem are solved one after the other multiple times. It is possible to re-use the matrices and preconditioner. Iterative method. Re 250 000 - AoA 1°
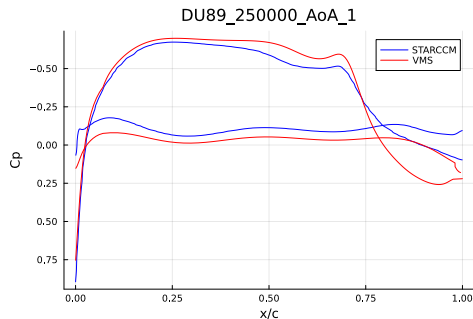
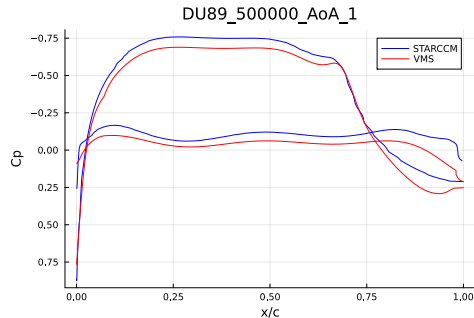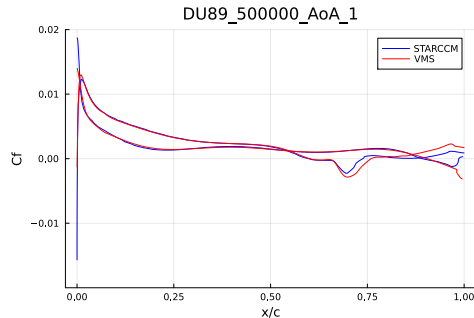# VMS Linearized-Segregated

4 Laminar Separation Bubble

Re 250 000 - AoA 5°

DU89_250000_AoA_5

DU89_250000_AoA_5

4 Laminar Separation Bubble

Re 500 000 - AoA 1°
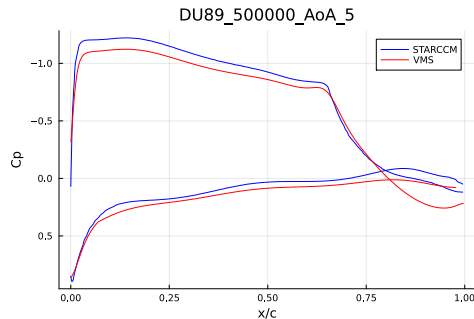
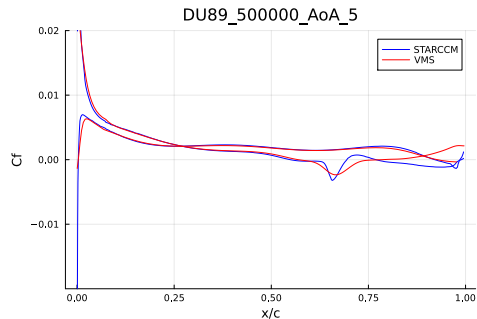

DU89_500000_AoA_1

DU89_500000_AoA_1

VMS Linearized-Segregated

4 Laminar Separation Bubble

Re 500 000 - AoA 5°

DU89_500000_AoA_5

DU89_500000_AoA_5

▶ Numerical Methods

▶ Performances

▶ Synthetic Eddy Method

▶ Laminar Separation Bubble

▶ Other activities

▶ Next steps

- Synthetic Eddy Method publication
- JuliaCon 2023 conference
- Co-author AIAA 2023 conference paper

► Numerical Methods

► Performances

► Synthetic Eddy Method

► Laminar Separation Bubble

► Other activities

► Next steps

- VMS paper (prof. Janssens reading)
- Experimental validation of the LS-VMS - publish paper
- LC-VMS - publish paper?
- AIAA2024 conferences (LS-VMS, LC-VMS)
- DLES14
- ICAS2024