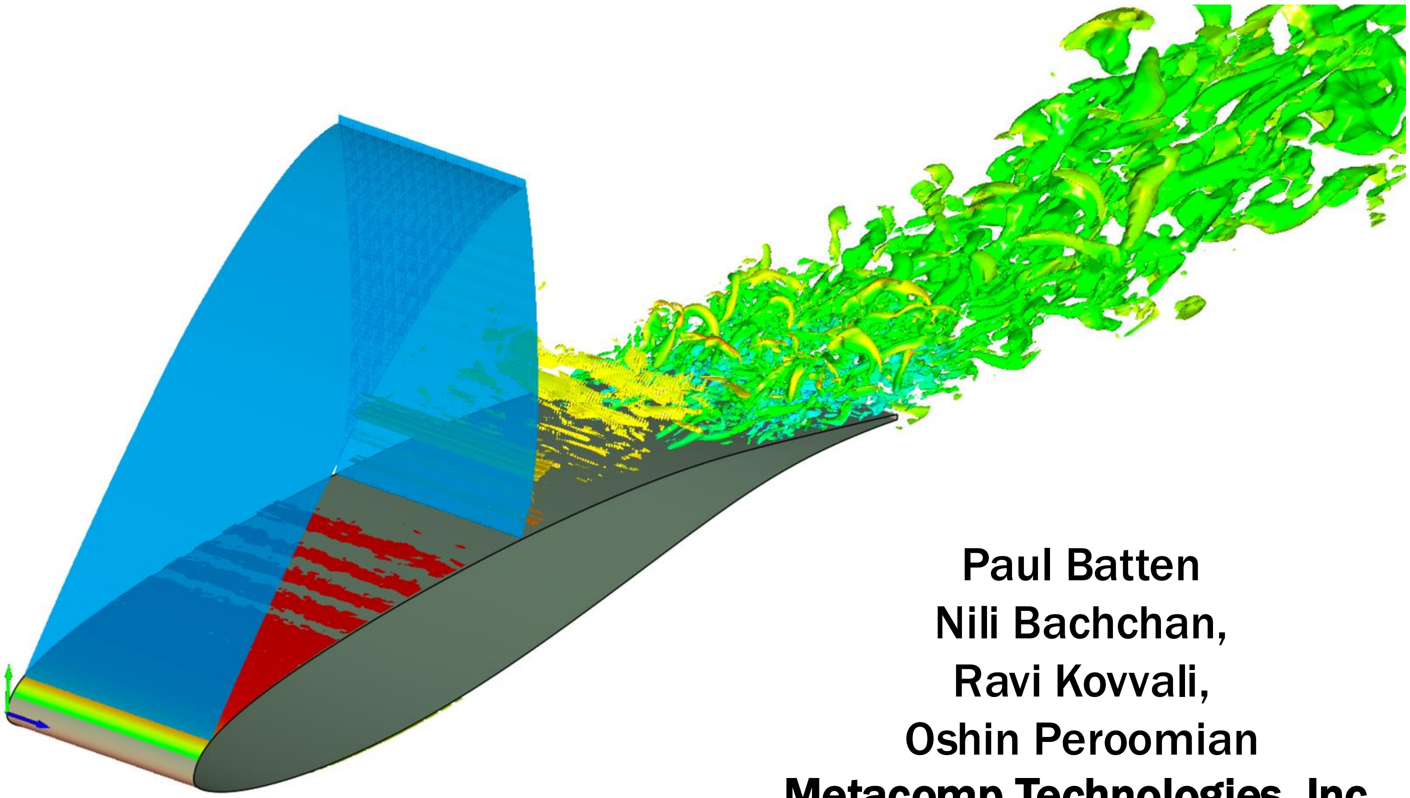
RANS, URANS and Hybrid RANS/LES Computations of the DPW-8 ONERA OAT15A Test Case Using CFD++



Paul Batten
Nili Bachchan,
Ravi Kovvali,
Oshin Peroomian
Metacomp Technologies, Inc.
AIAA Aviation, July 2025

Overview

- DPW8 ONERA OAT15A test case
- Solver, settings & models
- Traditional steady/unsteady approaches
- Something slightly unconventional
- Results
- Summary

DPW8 ONERA OAT15A Airfoil Test Case

1. Widely-studied aerodynamic benchmark case based on 2D supercritical airfoil geometry
2. Transition at x/c=7% on both upper & lower surfaces
3. Re(chord) = 3e6
4. M=0.73
5. AoAs = 1.36°, 1.5°, **2.5°**, **3.0°**, **3.1°**, **3.25°** 3.4°, **3.5°**, 3.6°, **3.9°**
6. Pressure/time histories recorded from:
   probe 1 (x/c=0.9 on upper surface)
   probe 2 (x/c=0.55 on lower surface)
   probe 3 (x/c=0.45 on upper surface)
7. Shock-oscillation period is ~ 0.012 s or ~ 13 CTUs

*__Data available__ from Jacquin et. al., *Experimental Study of Shock Oscillation over a Transonic Supercritical Profile*, AIAA J. Vol. 47, No. 9, 2009

Solver & Settings

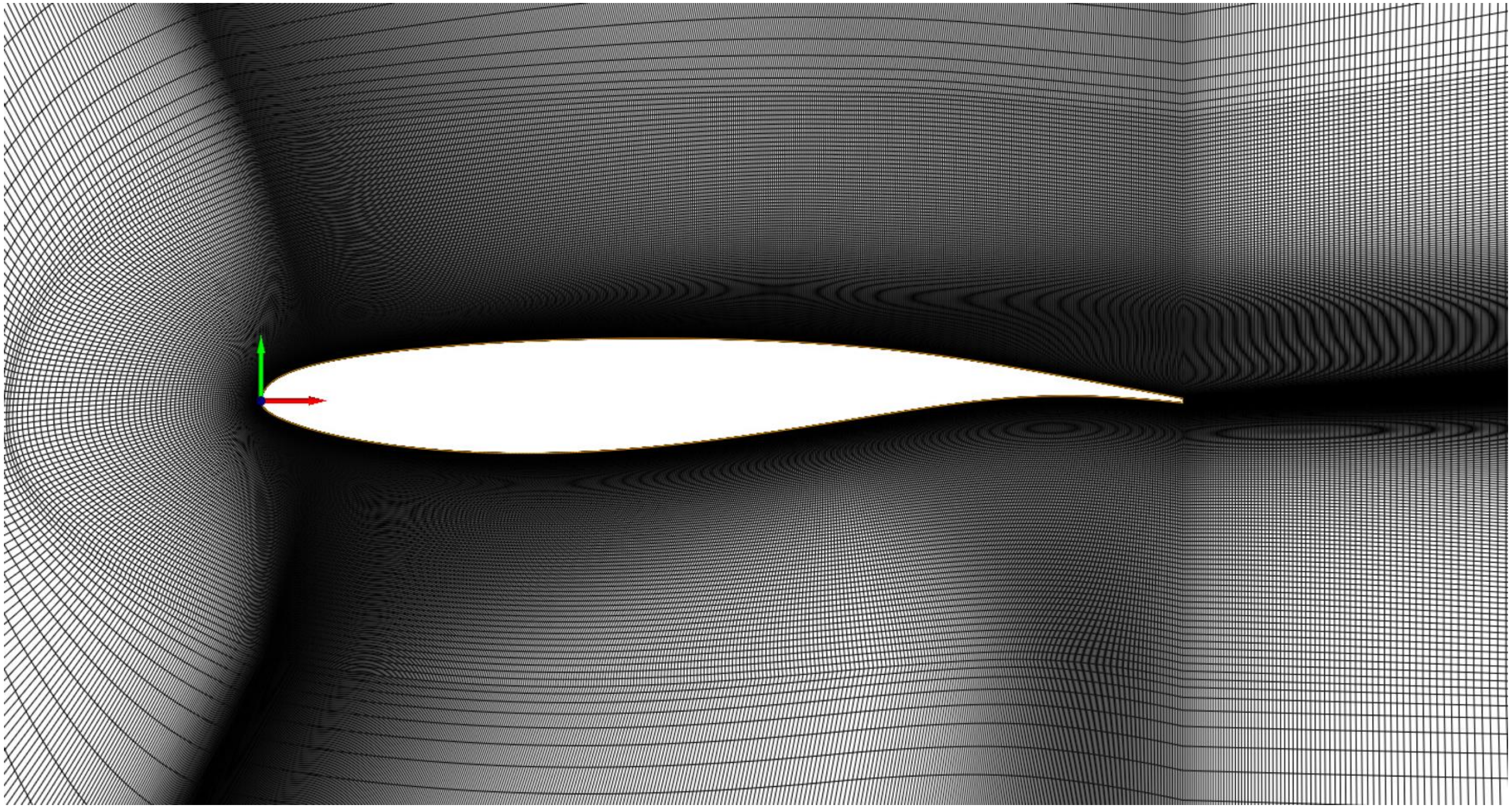
- CFD++ version 22.5
- Nodal-reconstruction transport scheme
- HLLC Riemann solver
- Implicit solver with algebraic multi-grid acceleration
- SA(neg)RC+QCR2013 and SST (U)RANS
- SA-based DDES (+ Deck/Renard EPSF)

Solver & Settings Cont.

- Self-tuning far-field absorbing layers
- NNDB TVD limiter (low diffusion, but *bounded*)
- ONERA-supplied mesh:
  - convective CFL=~1 gives Δt ~ 1e-6 s (920 steps/CTU)
  - acoustic CFL=~1 gives Δt ~ 7e-7 s (1314 steps/CTU)
- For URANS, Δt set as 9.2e-6 s (100 steps/CTU)
- URANS run on 2D (single-plane) and 3D meshes
- All HRLES run on extruded 3D meshes

ONERA Structured Mesh

- 2D mesh with 397,640 cells:



*Extruded through 145 planes for HRLES mesh with Δz = 0.00139c (~57.3 M cells)

Traditional Solution Options

1. Steady-state RANS
2. URANS
3. HRLES
4. WMLES
5. WRLES
6. DNS

One More Option…

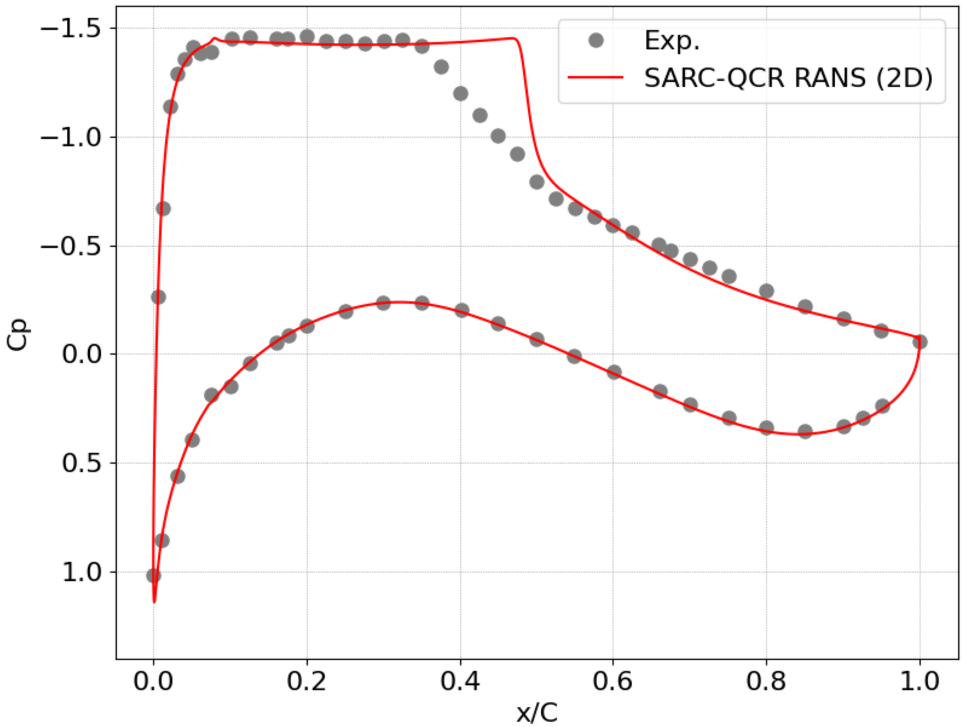
1. Steady-state RANS
2. URANS

2.5 Large time-step HRLES

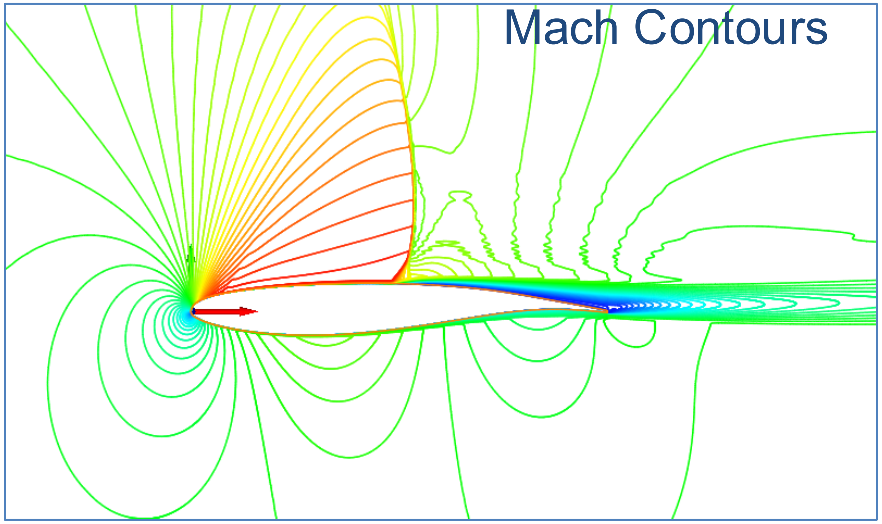
3. HRLES
4. WMLES
5. WRLES
6. DNS

Steady-State RANS – 3.5 Degree AoA

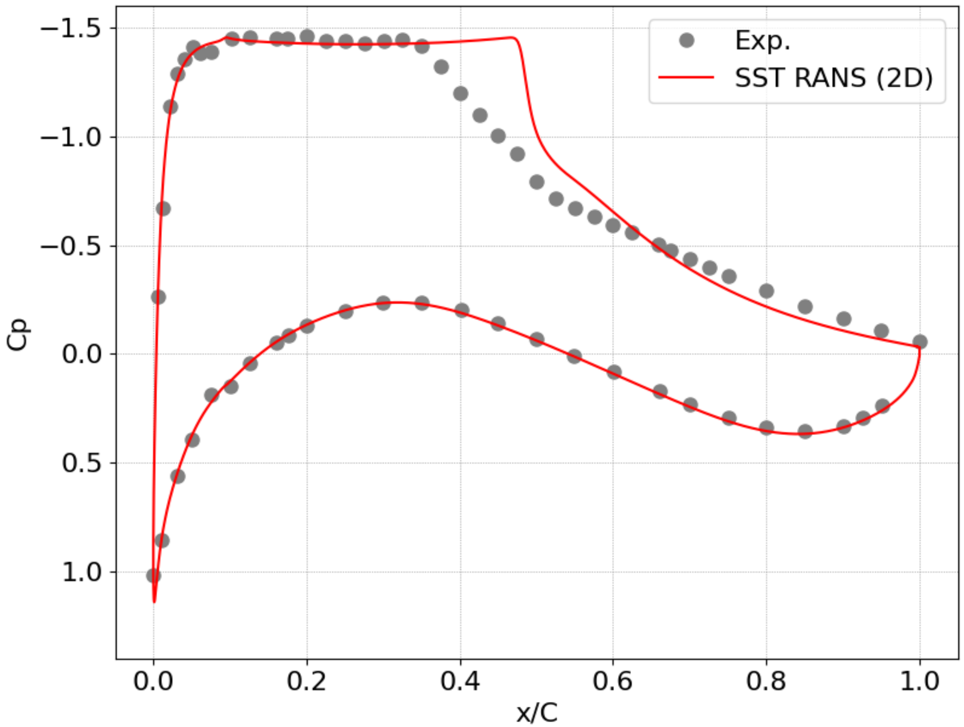
RANS – SA(neg)RC-QCR (AoA=3.5°)



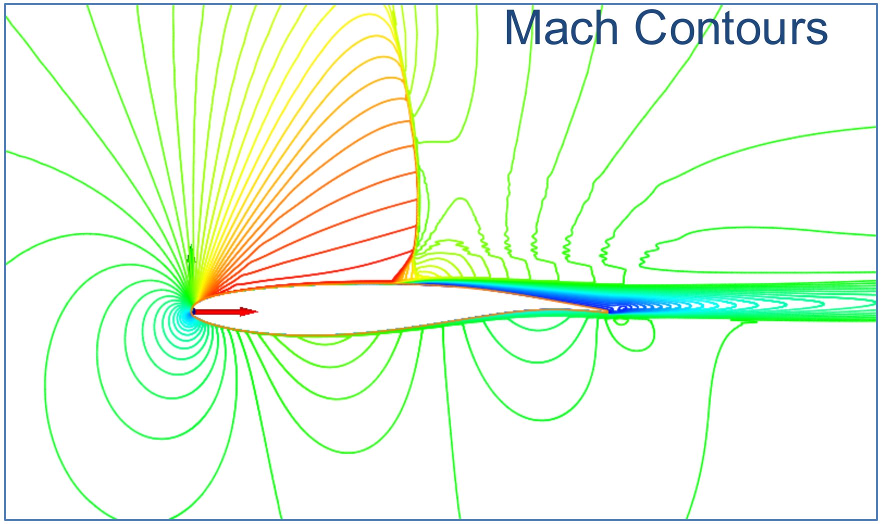
Surface Cp distribution



RANS – SST (AoA=3.5°)



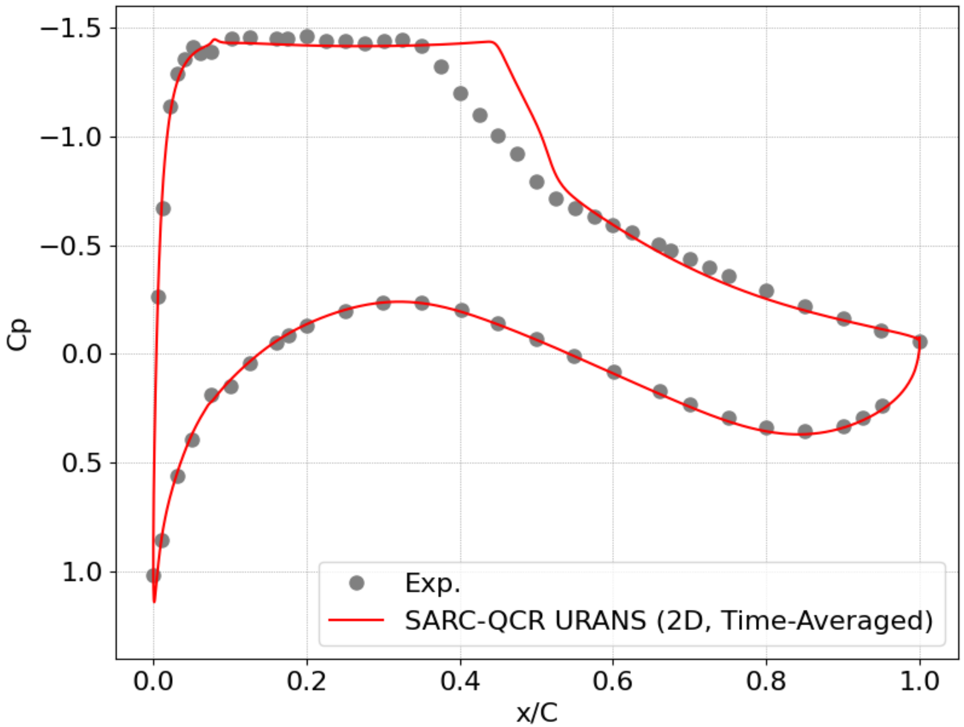
Surface Cp distribution



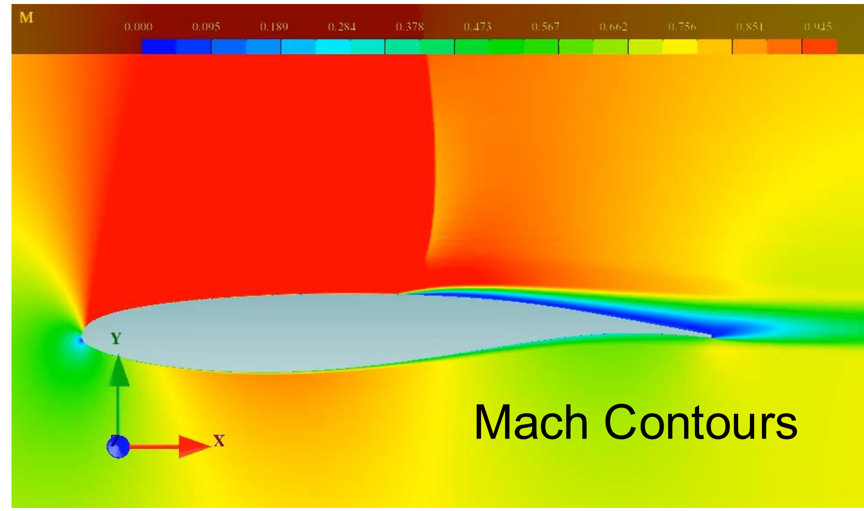
URANS – 3.5 Degree AoA

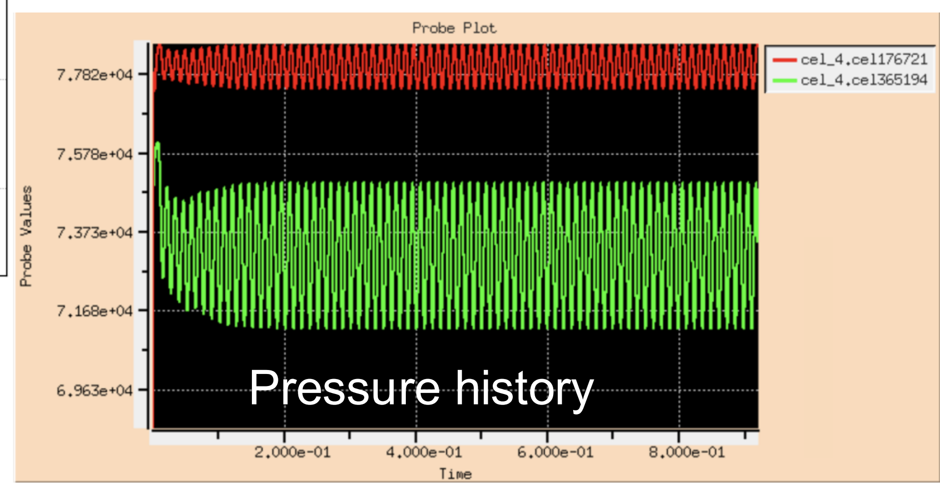
URANS – SA(neg)RC-QCR (AoA=3.5°)

Δt = 9.2e-6 s (100 steps/CTU)



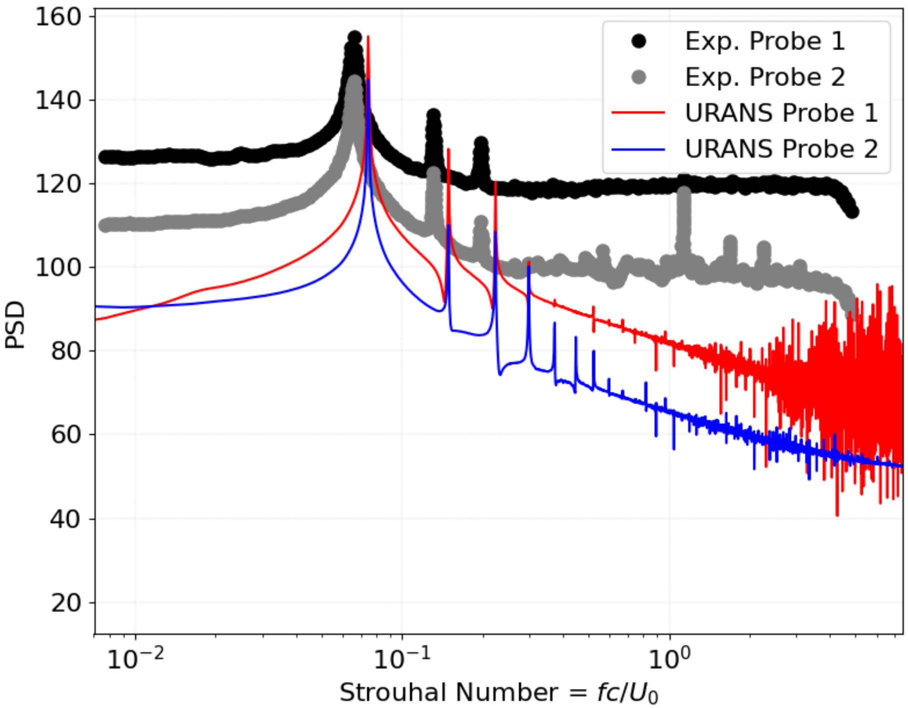
Surface Cp distribution (time-averaged)



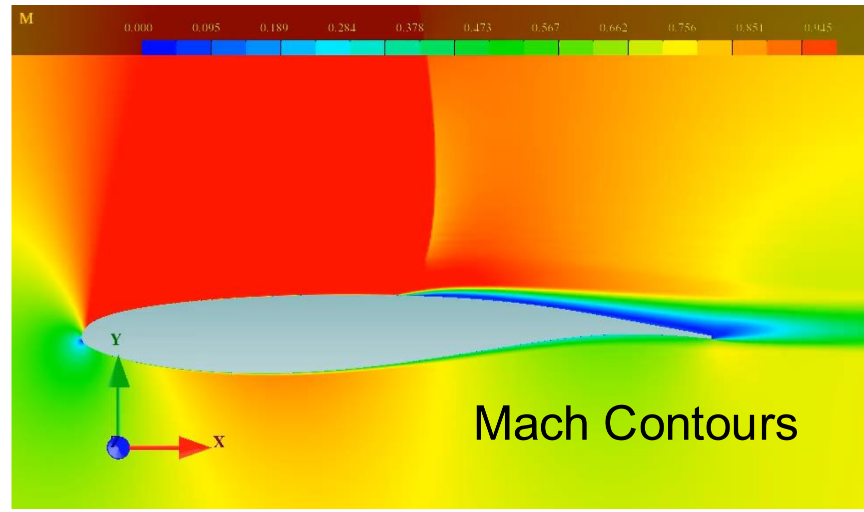


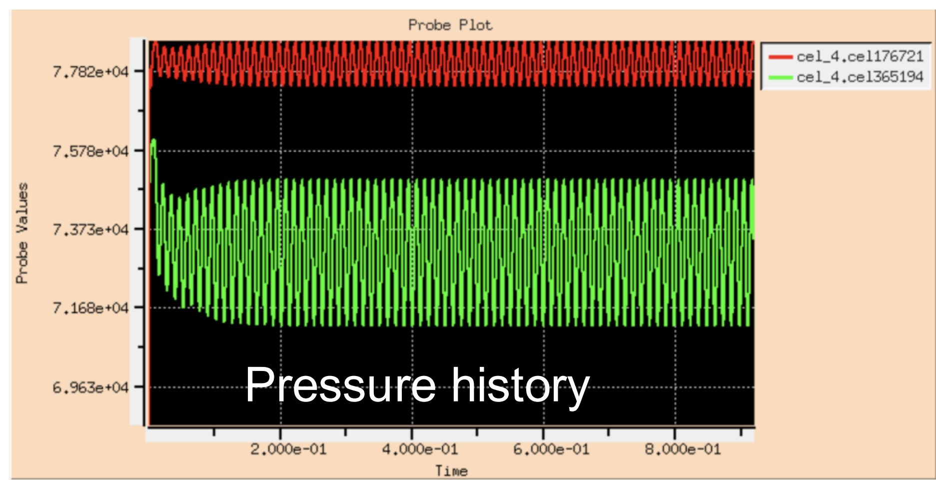
URANS – SA(neg)RC-QCR (AoA=3.5°)

Δt = 9.2e-6 s (100 steps/CTU)



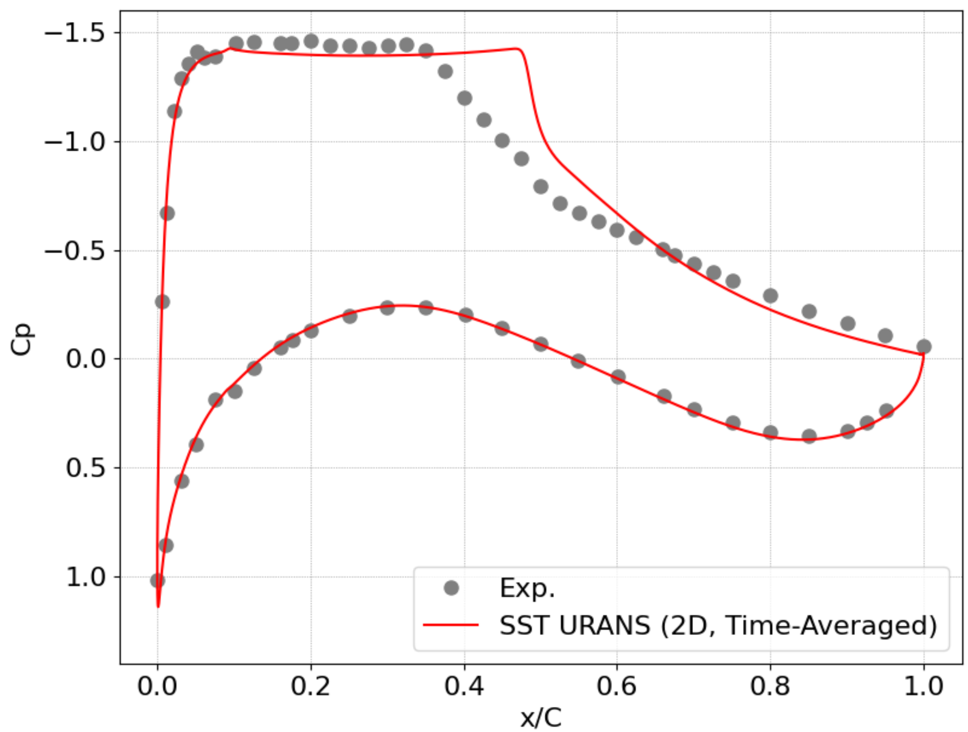
Pressure power spectral density



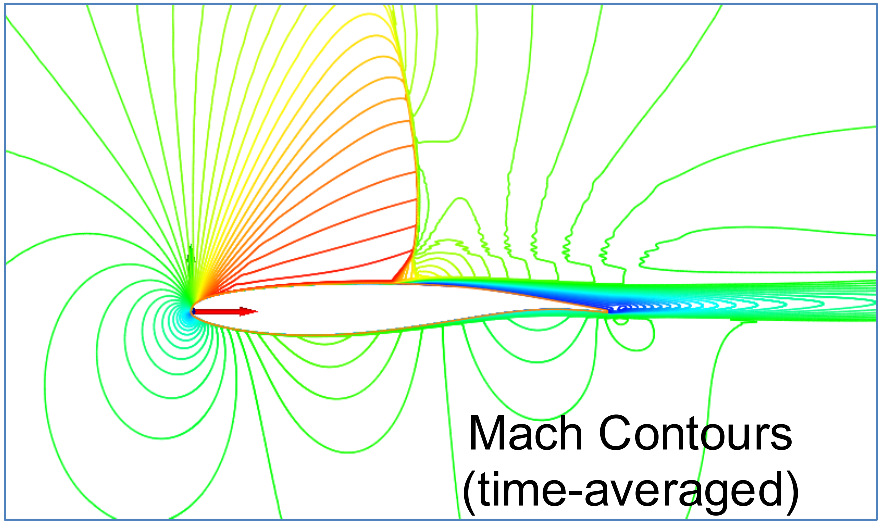


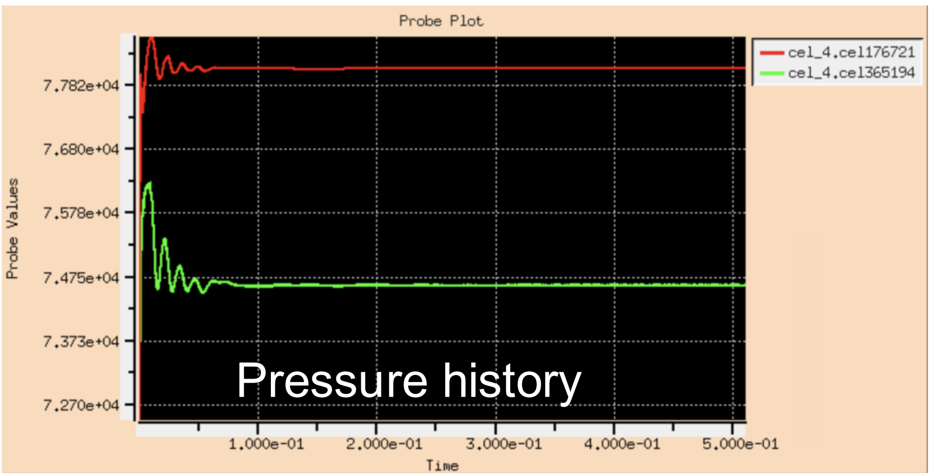
URANS – SST (AoA=3.5°)

No unsteadiness achieved



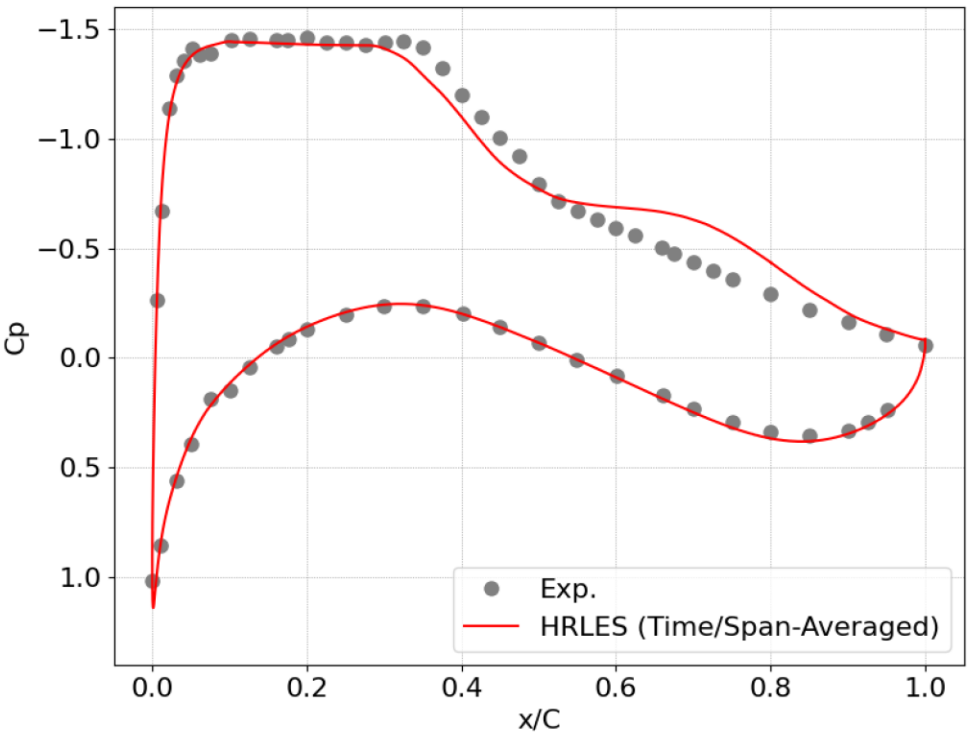
Surface Cp distribution (time-averaged)



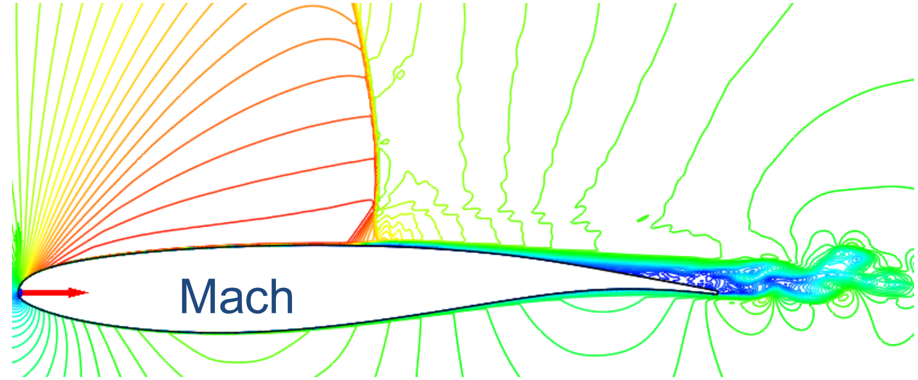


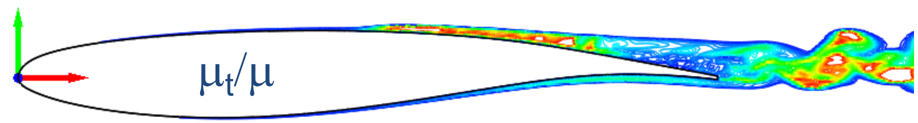
Hybrid RANS/LES – 3.5 Degree AoA

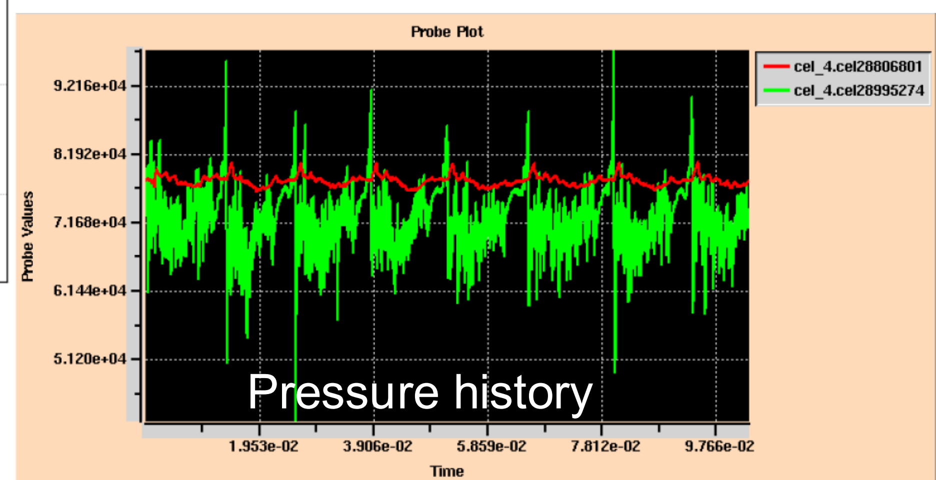
HRLES – SARC-QCR-DDES+DR (AoA=3.5°)



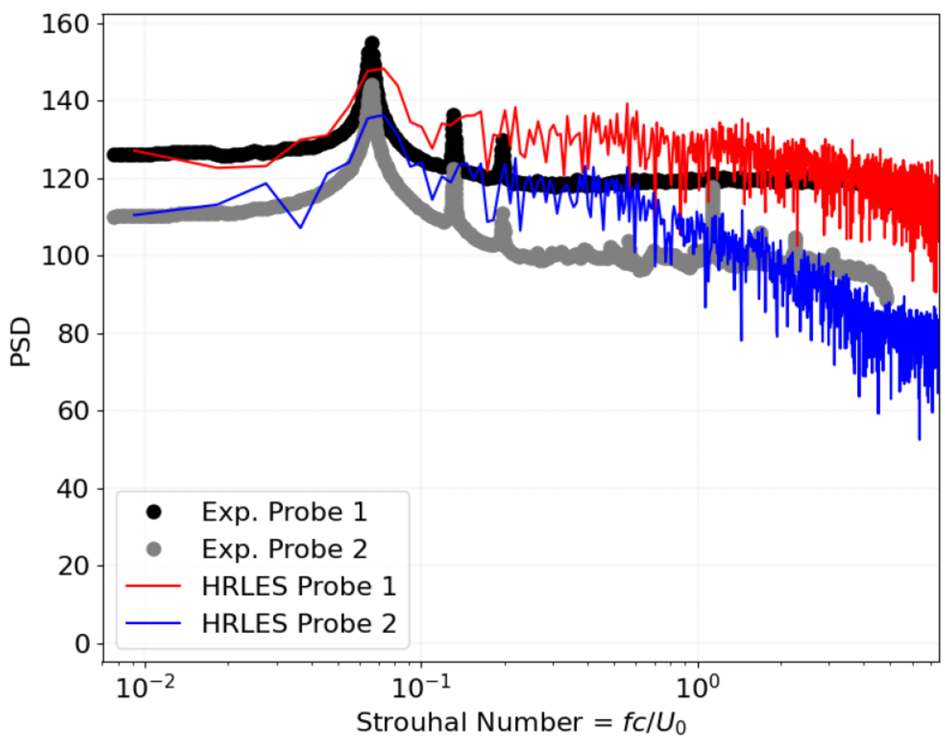
Surface Cp distribution
(time- & span-averaged,
~100 CTUs, ~8 periods, 57.3 M cell
ONERA mesh)



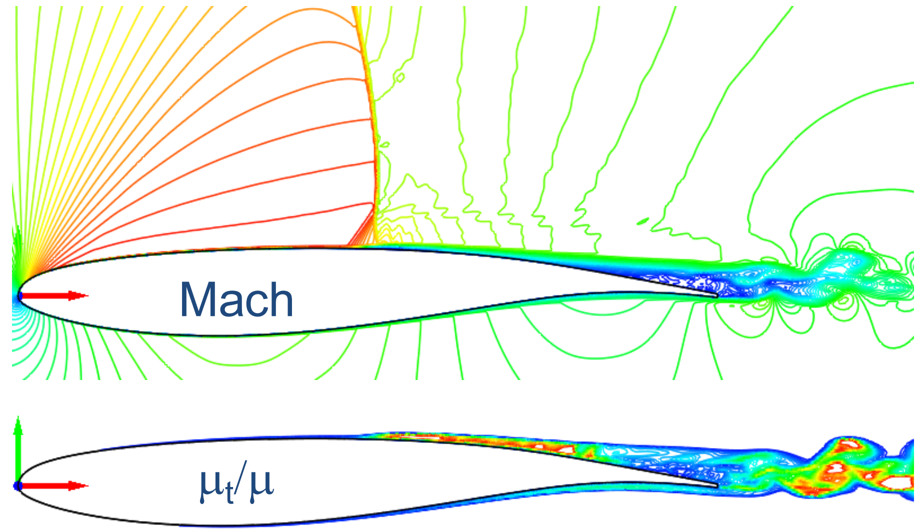


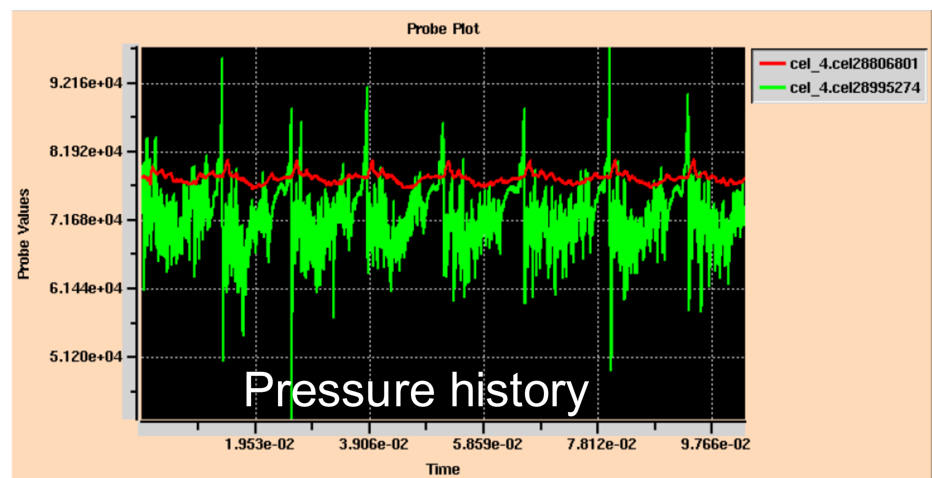


HRLES – SARC-QCR-DDES+DR (AoA=3.5°)



Pressure power spectral density
~100 CTUs, ~8 periods





Cost Overview

1. **URANS**: 100 steps/CTU*, 1 buffet cycle ~ 0.012 s, or about 13 CTUs
2. Recommended minimum of 8 buffet cycles to average over (ONERA)
3. Need about 13 cycles/periods of start-up transient to reach phase stationarity
4. (8+13)*13 = 273 CTUs, or 27,300 URANS time steps (spectral resolution suffers with smaller-length histories) – remember 2D mesh had only ~400,000 cells
5. **HRLES**: 3D extruded ONERA mesh (same Δz as Deck/Renard, 57.3 M elements) results in much larger grid and smaller Δt – results in ~1440 times the effort to simulate the same history length

*Suggested by Jeff Housman and Daniel Maldonado, **Evaluation of a Physics-Based Unsteady RANS Method for Buffet Onset Prediction**, NASA Advanced Supercomputing AMS seminar, **September 26, 2024**

The Problem with RANS

1. Well-understood limitations of phenomenological modeling

2. Models reach mesh-converged limit solutions – accuracy cannot be improved with finer meshes

3. Accuracy is 100% at the mercy of modeling, which usually excludes the effects of coherent-structure motion (significant at buffet)

4. Absence of unsteadiness results in a) SPL/Cp_rms of zero b) no averaged shock motion, hence a ‘time-averaged’ shock position which is too sharp and too far aft

The Problem with URANS

It's not reliable…

For example, SARC-QCR gives good convergence in RANS mode, but predicts unsteady flow in URANS mode; SST gives poor convergence in RANS mode, but predicts no unsteadiness in URANS mode

How can we know *a priori* what type of RANS model is appropriate, or *would even give an unsteady solution* for any given flow scenario?

1. There's no guarantee that URANS will even become unsteady
2. When resolved-scale structures do appear in URANS, decay rates are wrong:

   D. Israel, 'The Myth of URANS', Journal of Turbulence, Vol. 24, pp. 367-392, 2023.

The Problem with HRLES, WMLES, WRLES and DNS

Each of these approaches is forced to use a convective CFL of ~1 in regions of resolved-scale turbulence – failure to do so results in under-prediction of the total Reynolds stresses

Numerical resolution requirements set Δx, Δy, Δz, from which Δt is carved in stone (because of accuracy restriction on CFL)

The total number of steps, n, required for the simulation is then also carved in stone:

$$\mathbf{n} = \frac{\textbf{start-up transient time + time to gather sufficient history for reliable statistics}}{\Delta \boldsymbol{t}}$$

For these buffet problems, the numerator is so large that the problem becomes impractical, if not impossible

A Large Δt Approach to Simulation Cost-Reduction in HRLES

It's appealing to use large Δt in HRLES, similar to that used in URANS

… but large Δt in LES limits the ability to directly capture resolved-scale turbulence – ***so we cannot just naively increase Δt***

Potential solution is to sensitize the filter width to *Δt* so that we don't remove as much modeled stress as Δt increases:

$$\tilde{\Delta}^{II} = (1 - test_{\Delta})\tilde{\Delta}_{max} + test_{\Delta}(\tilde{\Delta}_{\omega}),$$

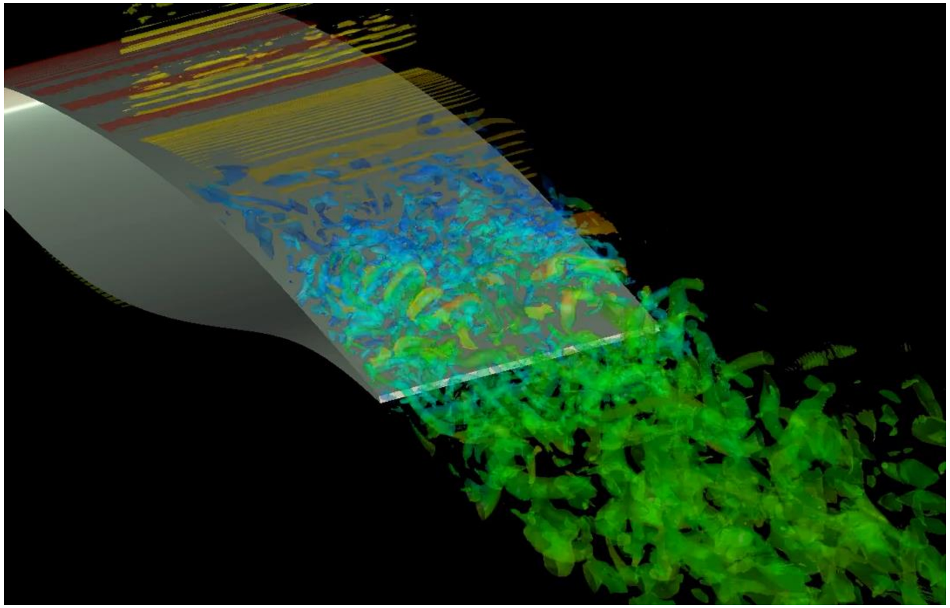
$$\tilde{\Delta}_{max} = \max(\Delta x, \Delta y, \Delta z, \sqrt{(u_i - \dot{x}_i)^2}\delta t)$$

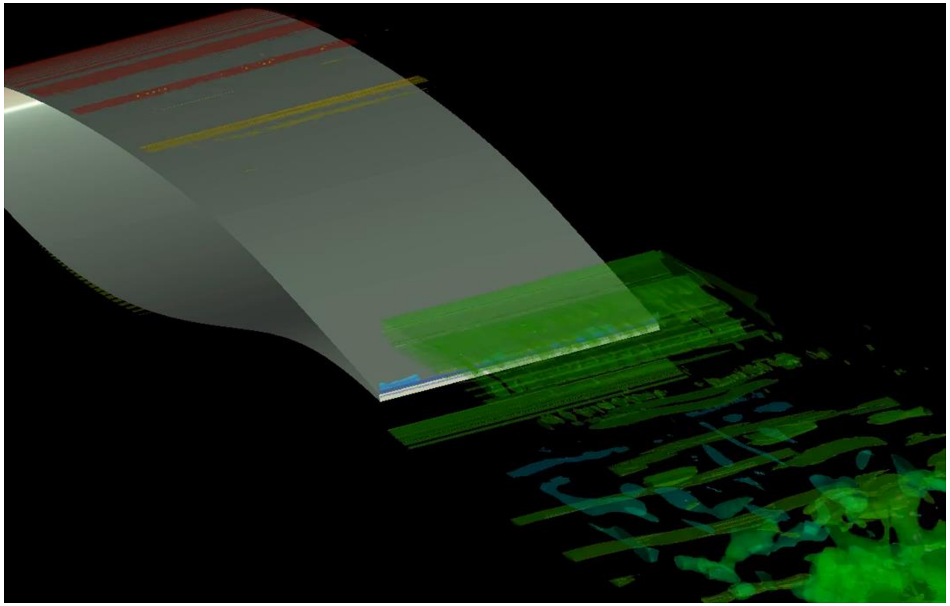
$$\tilde{\Delta}_{\omega} = \max(\Delta_{\omega}, \sqrt{(u_i - \dot{x}_i)^2}\delta t)$$

*Deck & Renard's `enhanced dissipation' option is not used here, as its goal is destruction of μ_t at separation, regardless of Δt

Traditional HRLES vs Large Δt HRLES

The following comparison was made on the same extruded (ONERA-supplied) mesh:





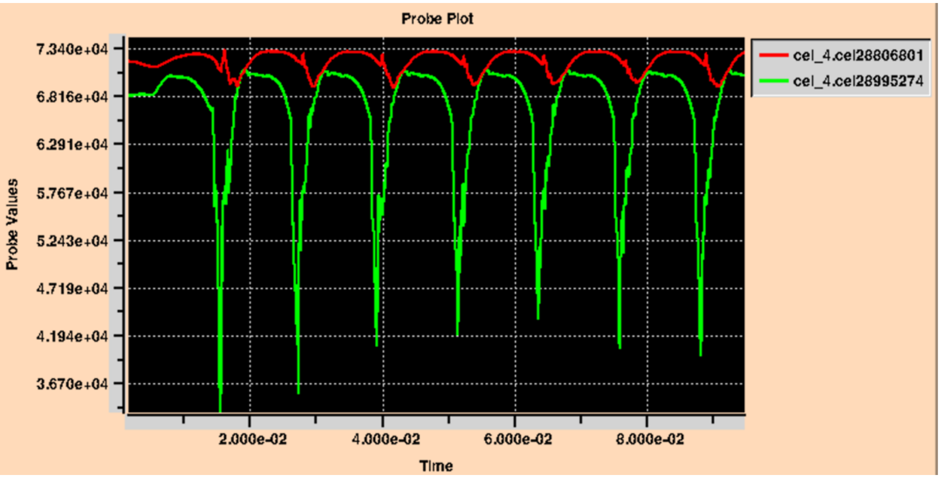
a) Traditional (CFL~1) DDES-DR

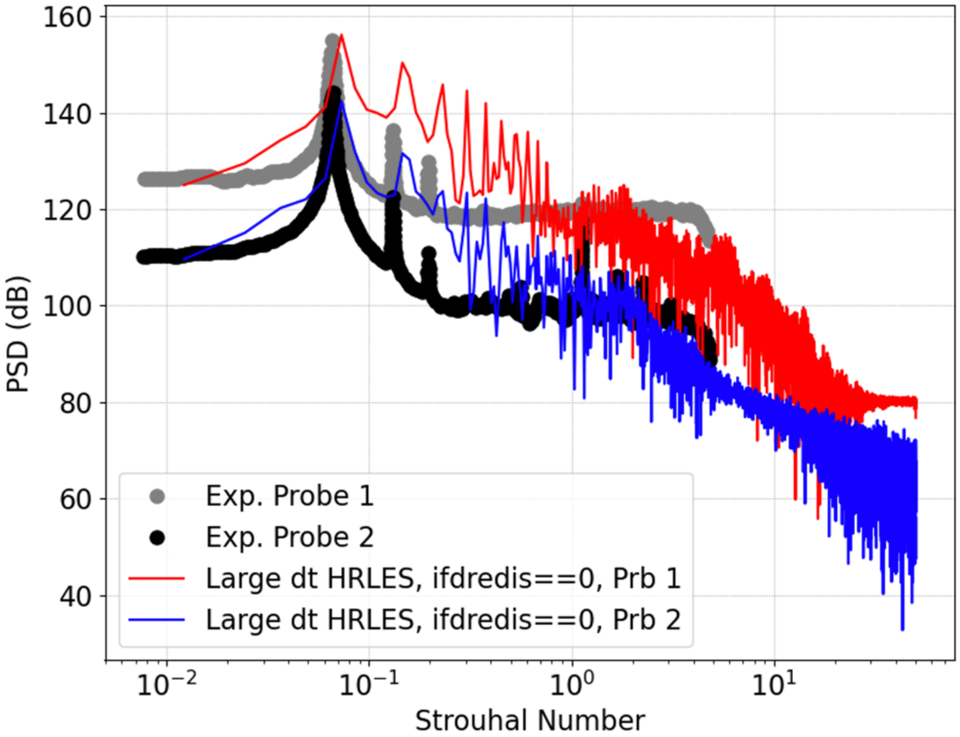
b) Large Δt DDES-DR

- Solution b) is severely lacking in resolved-scale content
- Usual assumption would be that solution a) is superior, however…

Large Δt DDES-DR HRLES Results

SWBLI motion is still apparent in large Δt DDES-DR and spectra (at least at lower st) are still reasonable:

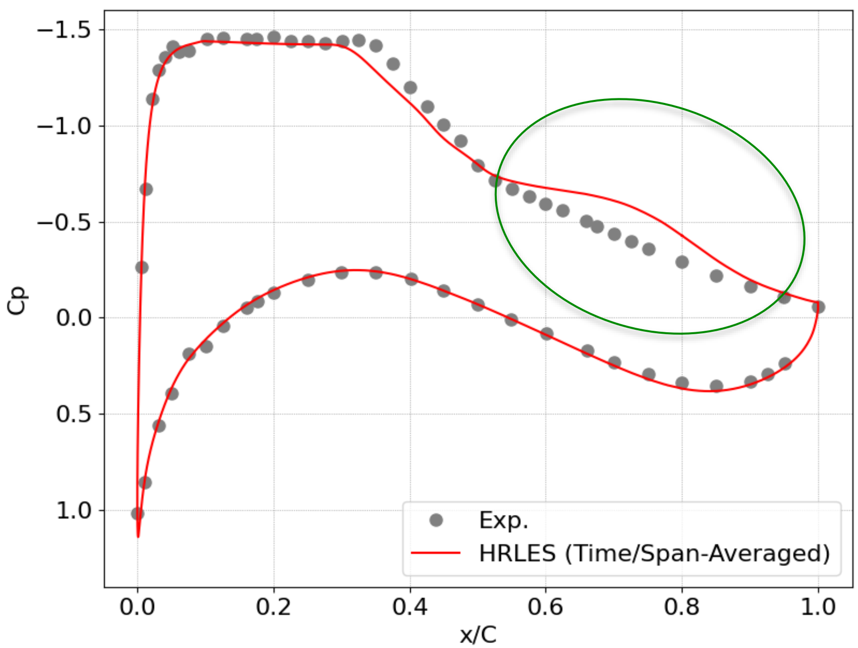




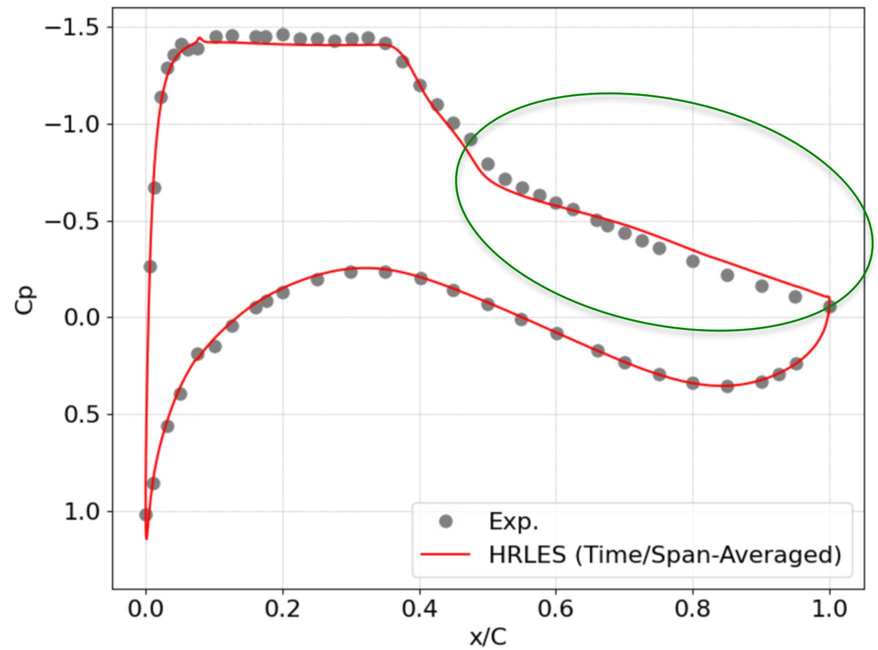
DDES-DR with Courant-sensitive filter width and URANS-sized (9.6e-6 s) Δt:

*Enhanced dissipation off

Comparison of Traditional DDES-DR and Large Δt DDES-DR



Traditional DDES-DR

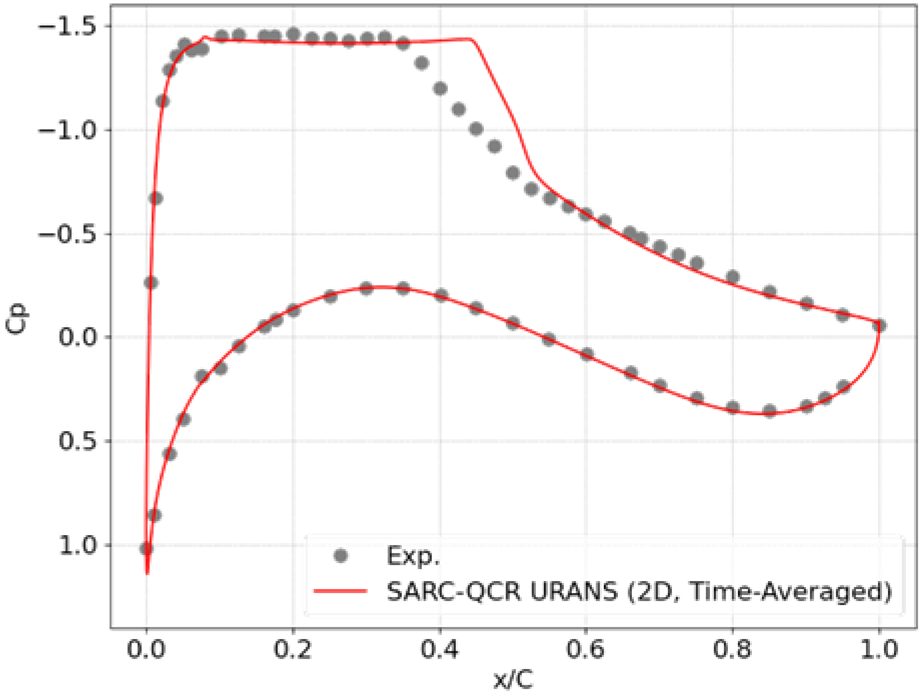


Large Δt DDES-DR

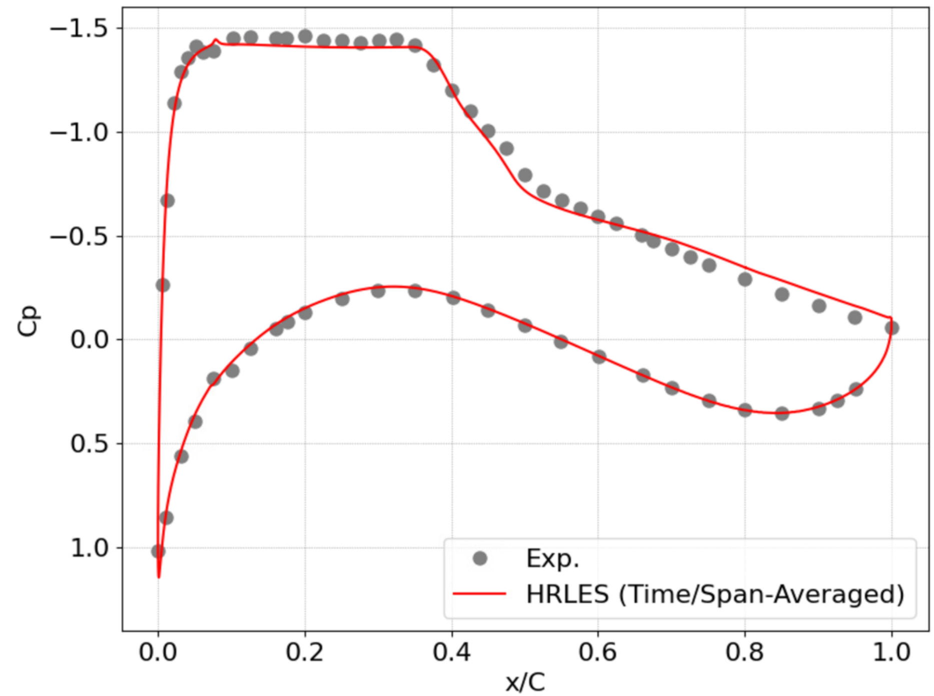
Note absence of modeled-stress depletion effects in large Δt DDES-DR

URANS vs Large Δt HRLES

- 2D URANS vs DDES-DR* with Courant-sensitive filter width (in both cases, Δt=9.6e-6 s)



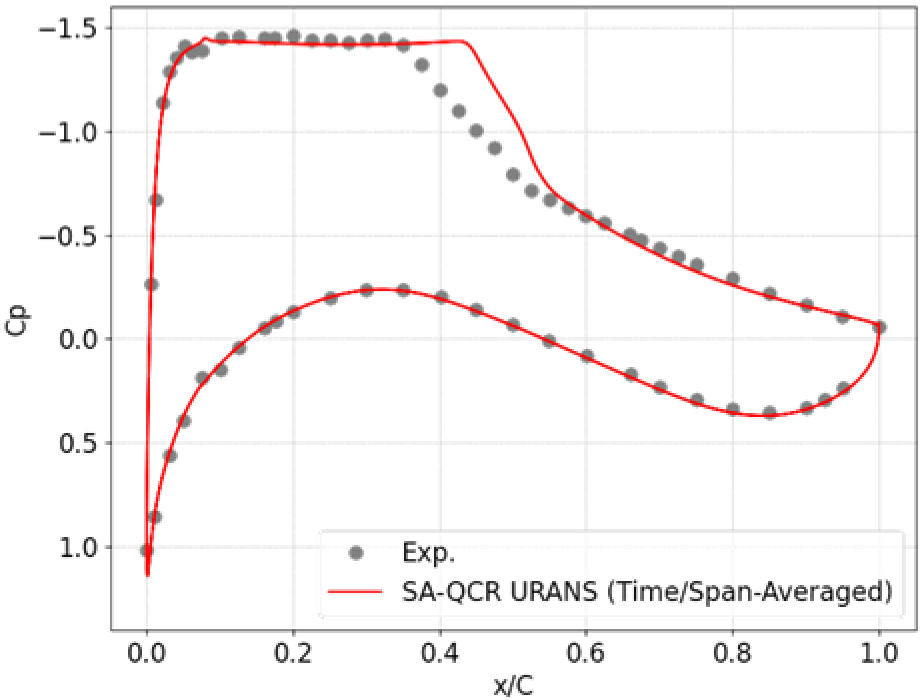
URANS, 2D ONERA mesh



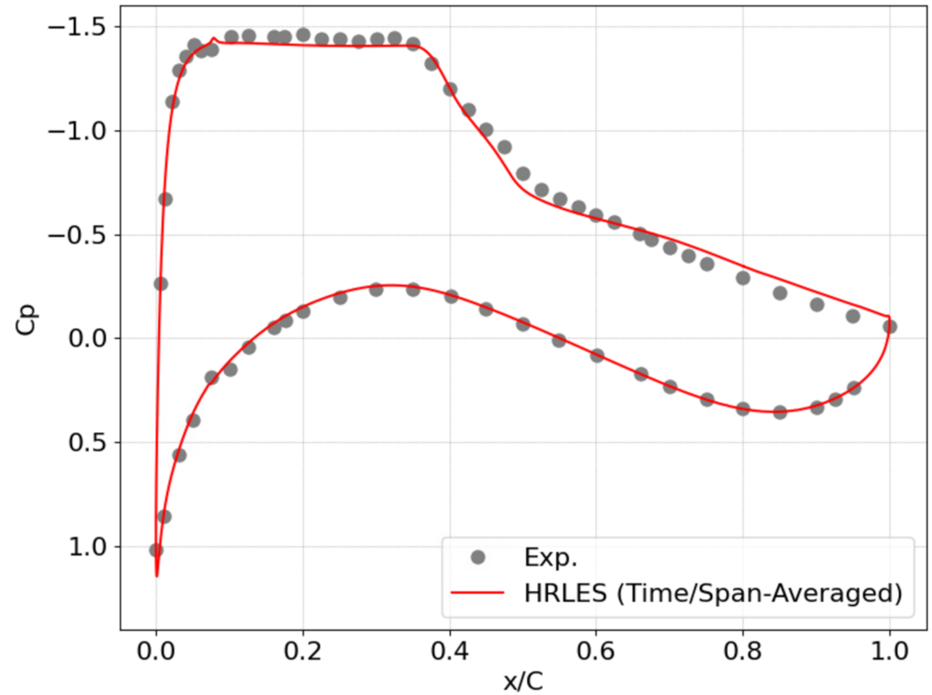
Large Δt HRLES, 3D ONERA mesh

3D URANS vs Large Δt HRLES

- 3D URANS vs DDES-DR* with Courant-sensitive filter width (in both cases, Δt=9.6e-6 s, identical 57.2 M cell ONERA mesh)



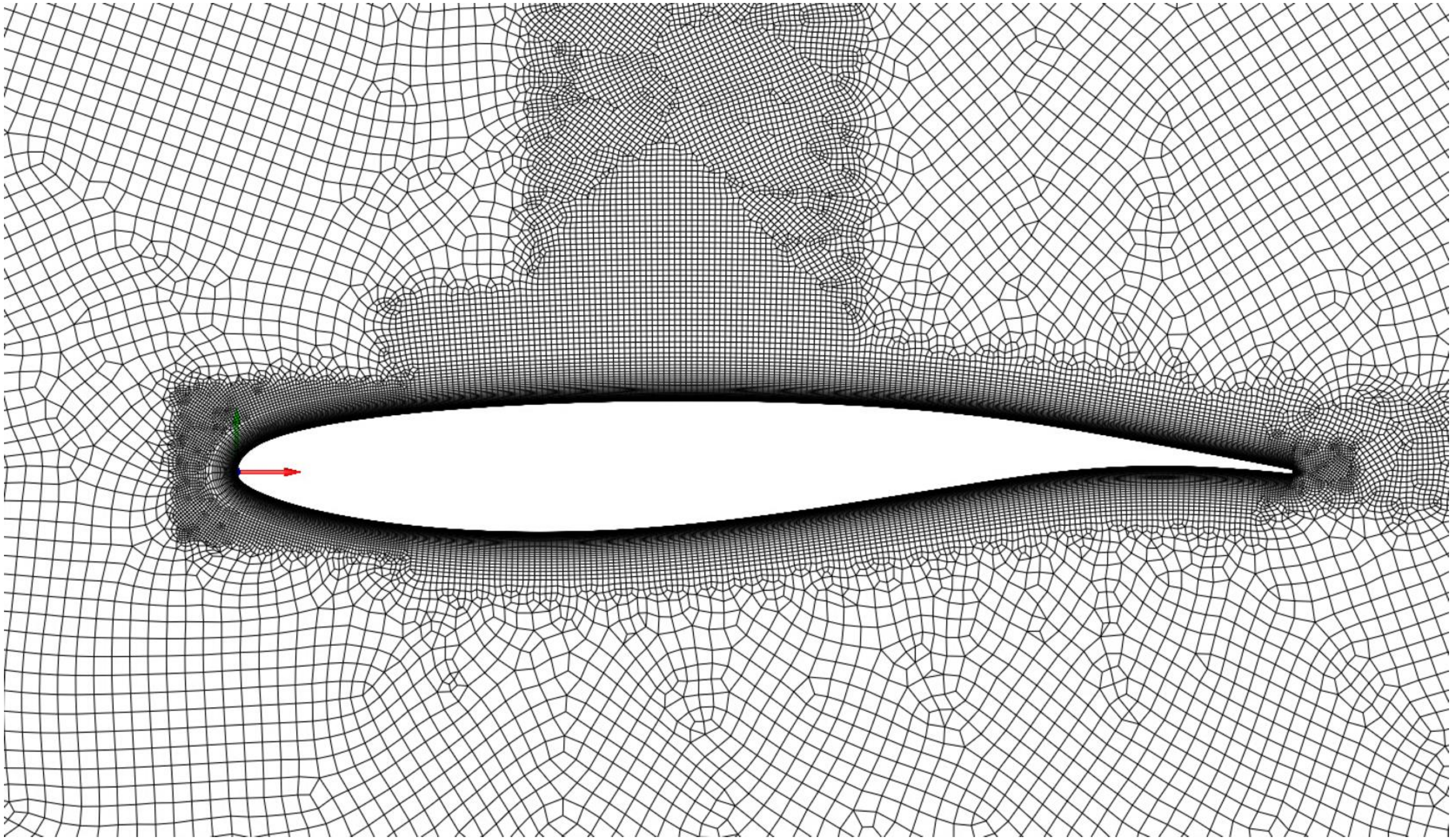
URANS, 3D ONERA mesh



Large Δt HRLES, 3D ONERA mesh

MIME Mesh

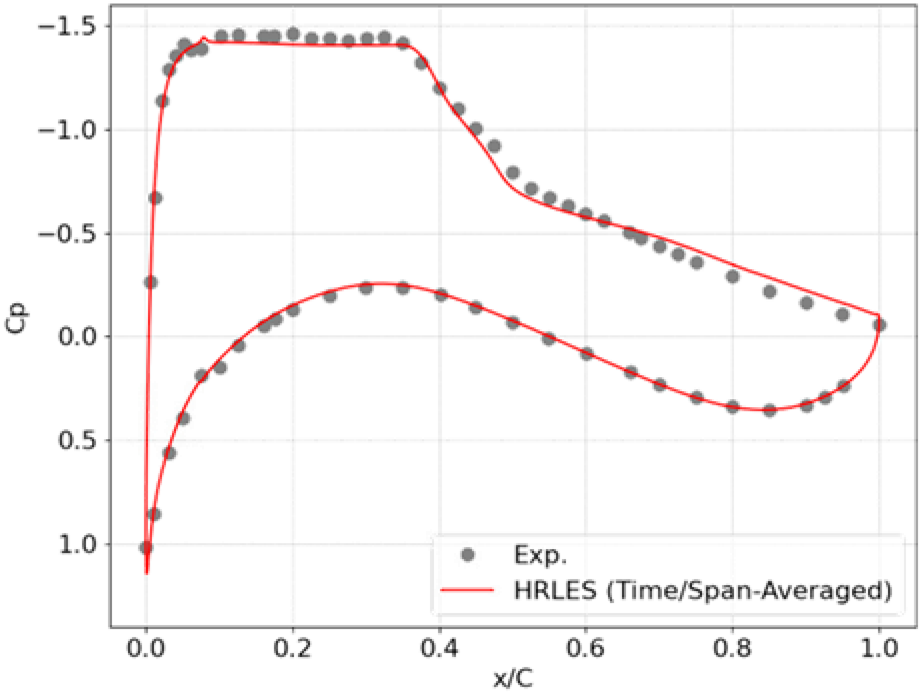
If we're using a larger Δt and an associated larger filter width, couldn't we also get away with using a much coarser mesh?



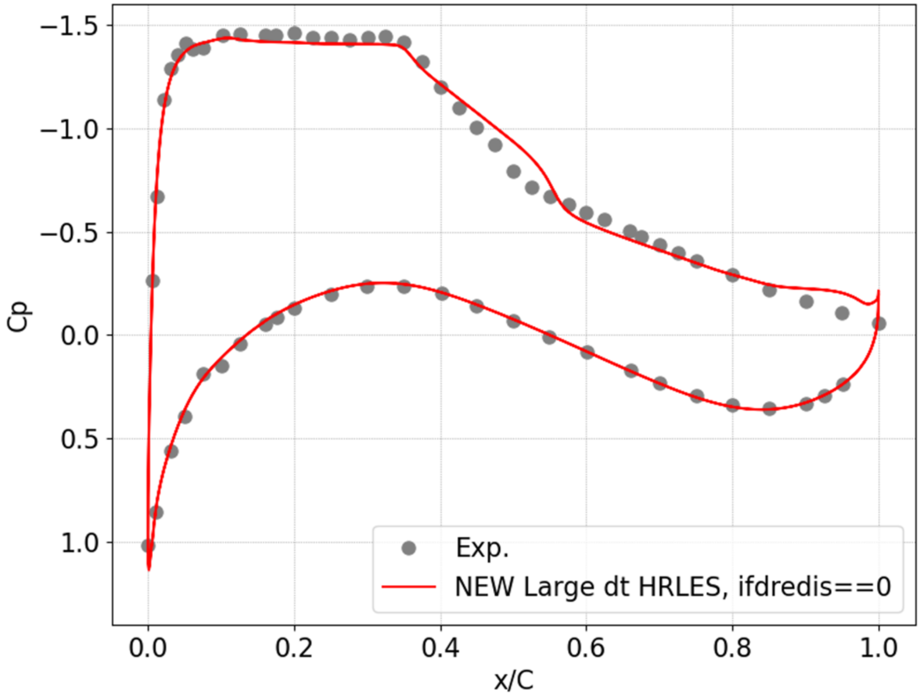
*Extruded over the same span, 3D MIME mesh has only ~2.2 M elements

Large Δt HRLES: ONERA vs MIME Mesh Results

- Rather similar results (Δt=9.6e-6 s in both simulations)

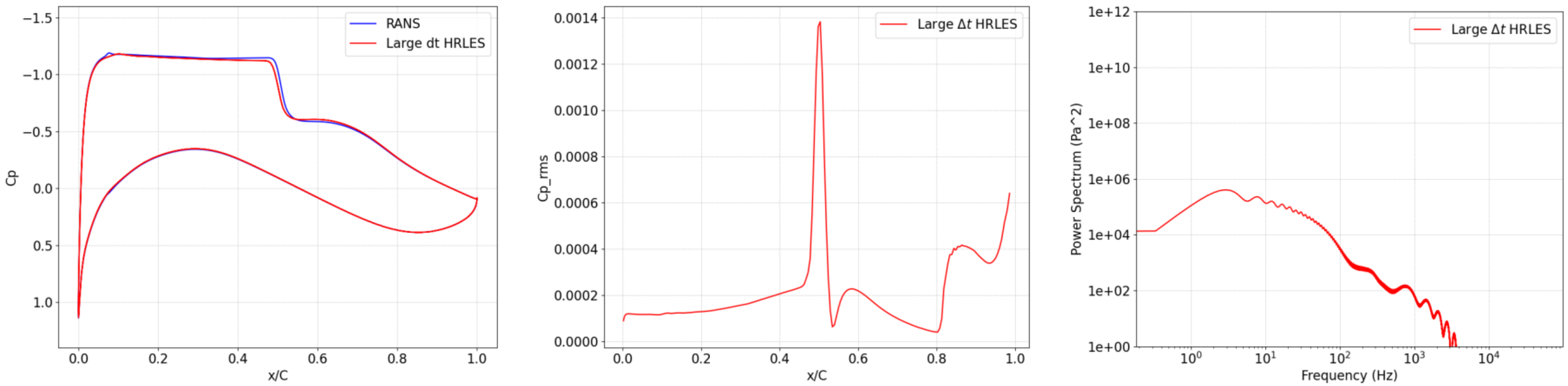


Large Δt HRLES, 3D ONERA mesh (57 M cells)



Large Δt HRLES, 3D MIME mesh (2.2 M cells)

MIME/Large Δt DDES-DR Alpha Sweep



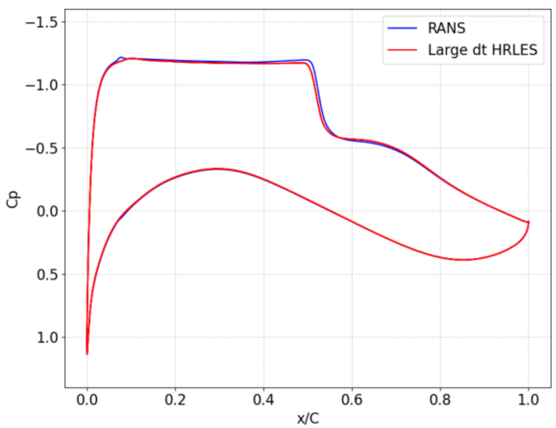
Angle of Attack: α = 1.36°

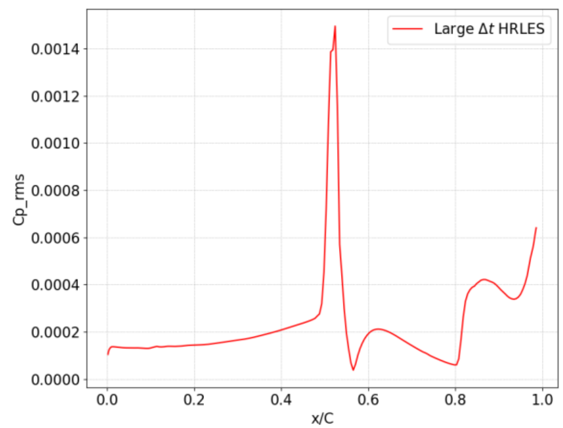
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

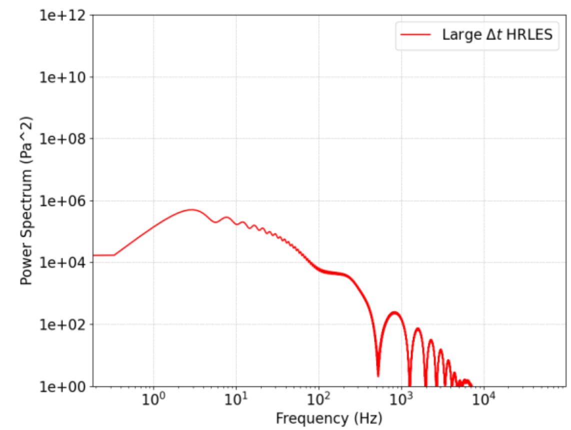
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







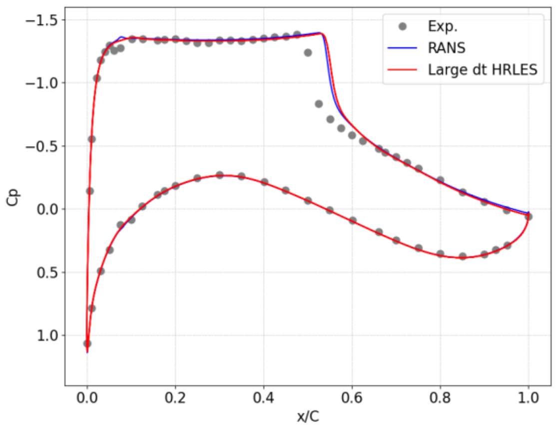
Angle of Attack: α = 1.50°

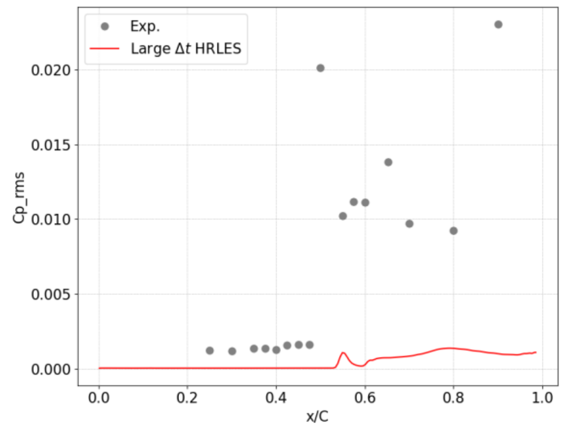
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

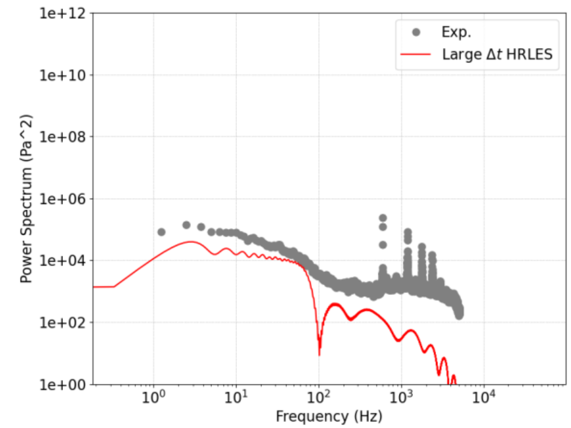
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







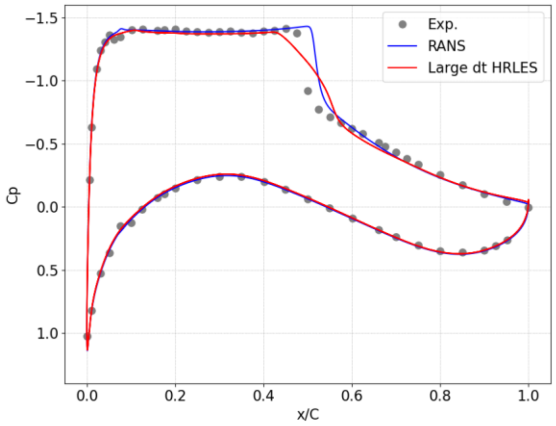
Angle of Attack: α = 2.50°

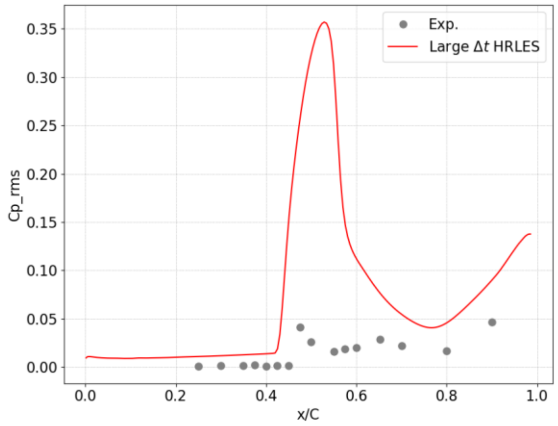
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

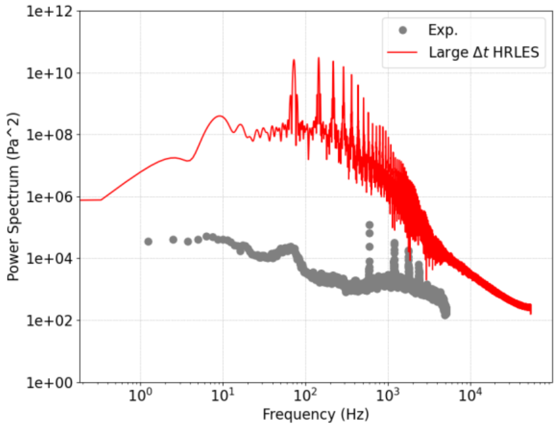
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







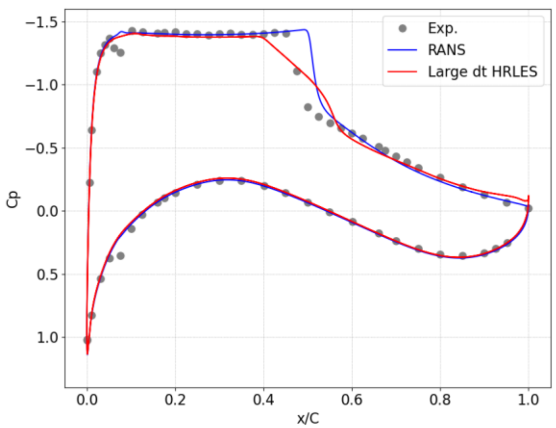
Angle of Attack: α = 3.00°

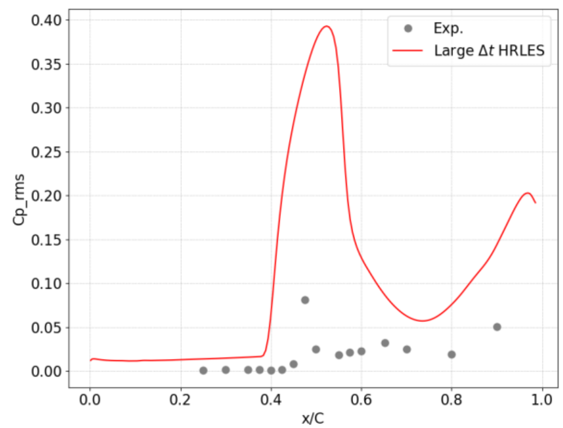
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

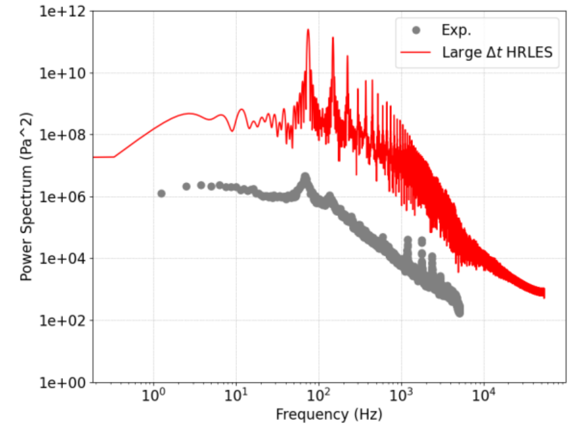
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







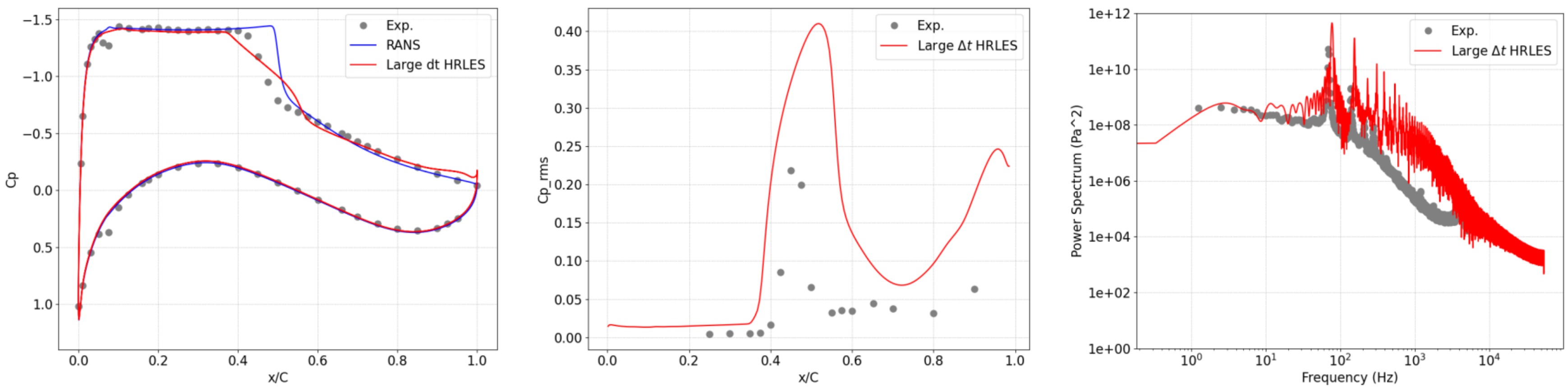
Angle of Attack: α = 3.10°

***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep



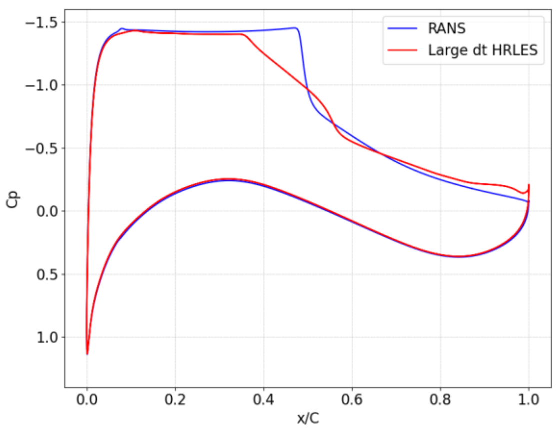
Angle of Attack: α = 3.25°

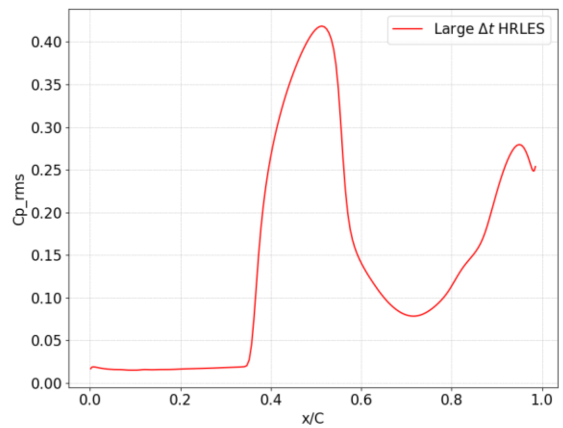
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

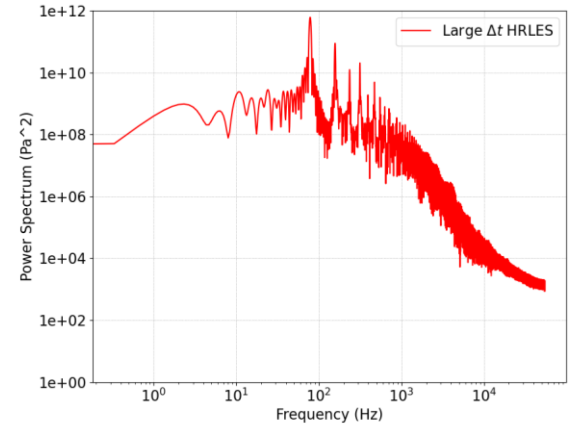
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







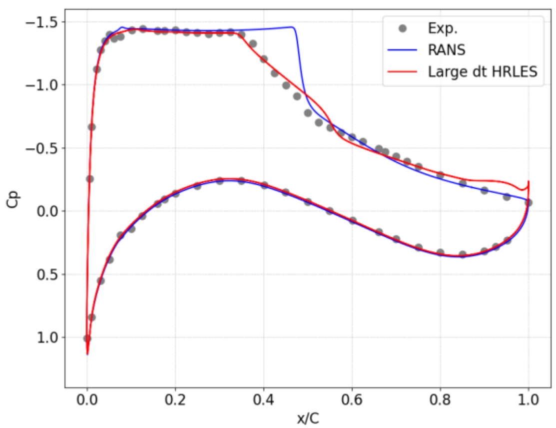
Angle of Attack: α = 3.40°

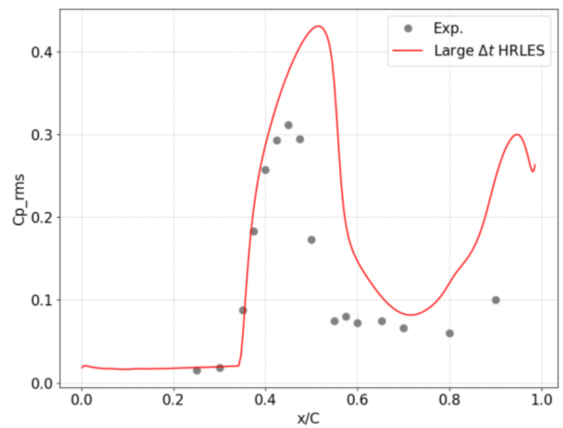
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

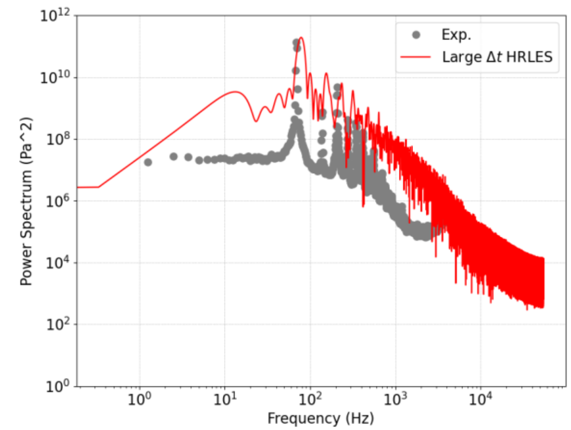
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







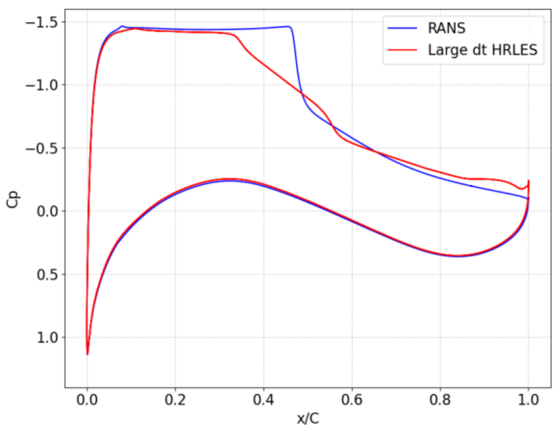
Angle of Attack: α = 3.50°

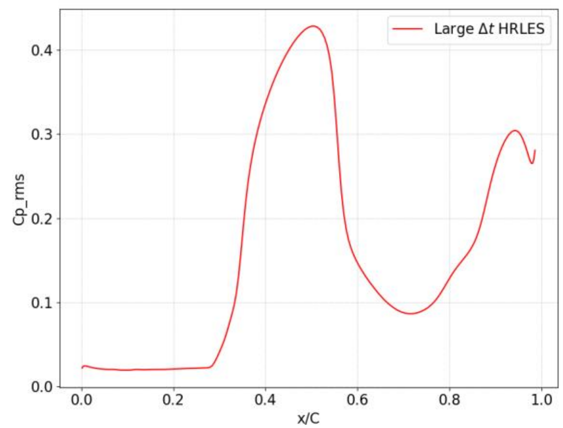
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

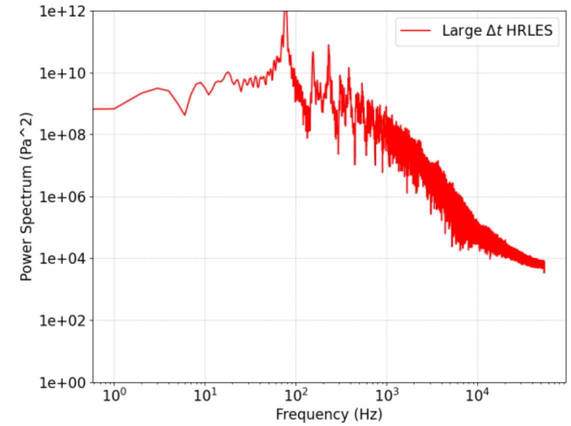
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







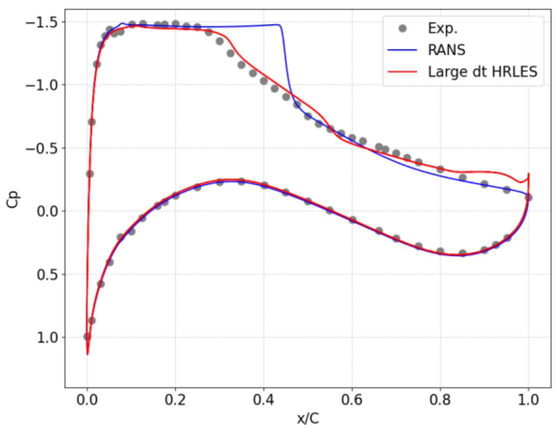
Angle of Attack: α = 3.60°

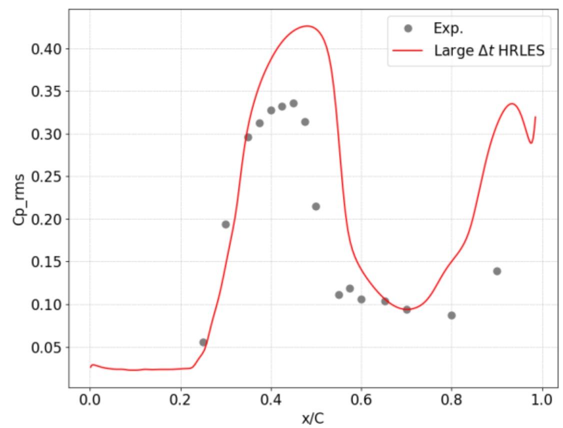
***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

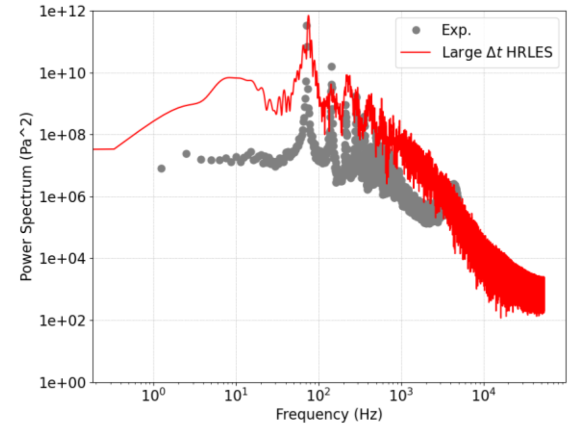
compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

MIME/Large Δt DDES-DR Alpha Sweep







Angle of Attack: α = 3.90°

***All power spectra (incl. those from Jacquin et al.) computed from E. Molina's script with:**

compute_psd_10212024.py -f filename -o 0.50 -d 1.0

****RANS solutions computed from finest Cadence-supplied mesh**

Summary

RANS is inaccurate for these SWBLI cases (shock too sharp/too far aft)

URANS is unreliable, and even when using dubious mechanisms to force unsteadiness, results aren't as accurate as those of HRLES

Traditional scale-resolving methods requiring convective CFL ~1 are outrageously expensive due to this huge number:

$$\mathbf{n} = \frac{\text{start-up transient + time to gather sufficient history}}{\Delta t}$$

Large Δt HRLES offers an affordable approach to SWBLI modeling

For the ONERA OAT15A test case, large Δt HRLES is substantially faster (cost factor of ~250), and gives *better agreement* with mean Cp, than traditional HRLES (because it avoids most RANS->LES MSD issues)