Adaptive Direct FEM simulation for HiLiftPW-3

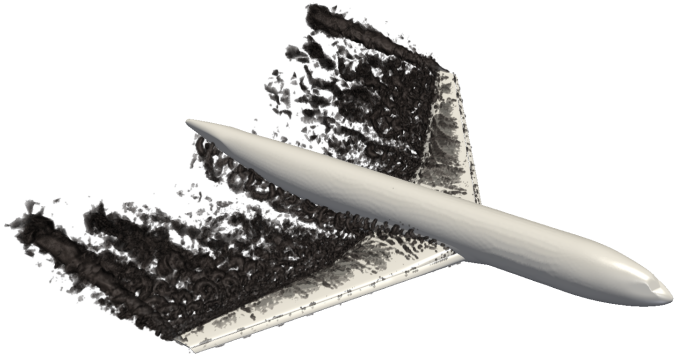
Johan Jansson[12], Ezhilmathi Krishnasamy[2], Massimiliano Leoni[12], Niclas Jansson[1], Johan Hoffman[1]

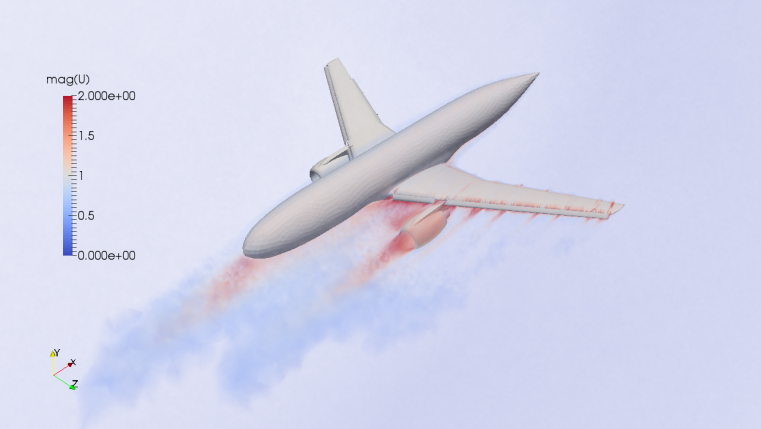
PID: 032

Computational Science and Technology CSC, KTH, Stockholm [1]

BCAM - Basque Center for Applied Mathematics, Bilbao [2]

3rd High Lift Prediction Workshop
Denver, CO, June 3-4, 2017





Summary of cases completed: Unicorn, custom grids, Direct FEM (no turb. mod.)

Focus only on adaptive JSM cases:

Case 2b, (no nacelle, w adaptation), Polar, fully turbulent

Case 2d, (with nacelle, w adaptation), Polar, fully turbulent

Preliminary results submitted.

Here improved and more complete results are presented.

Summary of code and numerics used - Automated Computational Modeling

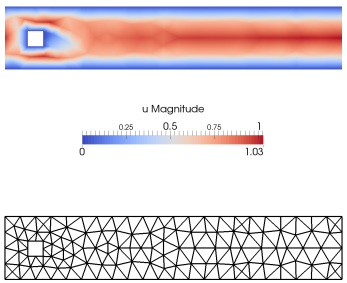
FEniCS(-HPC) open source FEM framework for automated solution of general PDE and Direct FEM Simulation (DFS) methodology enables:

Automated discretization (generate code for linear system from PDE):

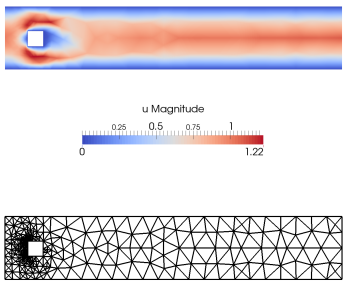
```
r = (inner(grad(u), grad(v)) - inner(f, v))*dx
```

$\Rightarrow$ `Poisson.cpp`

Automated error control (incl. parallel adaptive mesh refinement):



$+ \ |M(e)| \leq TOL \ \Rightarrow$



with $M(e)$ a goal functional of the computational error $e = u - U$.

Automated modeling of unresolved subscales (i.e. turbulence):

$(R(U), v) + h(R(U), R(v)) = 0, \forall v \in V_h$ (residual-based stabilization/dissipation)

Goal: Autom. generate the solution, mesh and program from PDE (residual) and goal functional $M(e)$ (e.g. drag).

Summary of code and numerics used - Direct FEM Simulation (DFS) methodology

Developed over a 20+ year period by Johnson, Hoffman, Jansson, etc.

For HiliftPW-3 incompressible Euler as model for high Re flow:

$$\begin{aligned} R(\hat{u}) &= \begin{cases} \partial_t u + (u \cdot \nabla) u + \nabla p = 0 \\ \nabla \cdot u = 0 \end{cases} \\ u \cdot n &= 0, x \in \Gamma \quad \text{(Slip BC)} \\ \hat{u} &= (u, p) \end{aligned}$$

Weak residual $r(\hat{u}, \hat{v}) = (R(\hat{u}), \hat{v})$

Space-time cG(1)cG(1) FEM with Galerkin/least squares stabilization

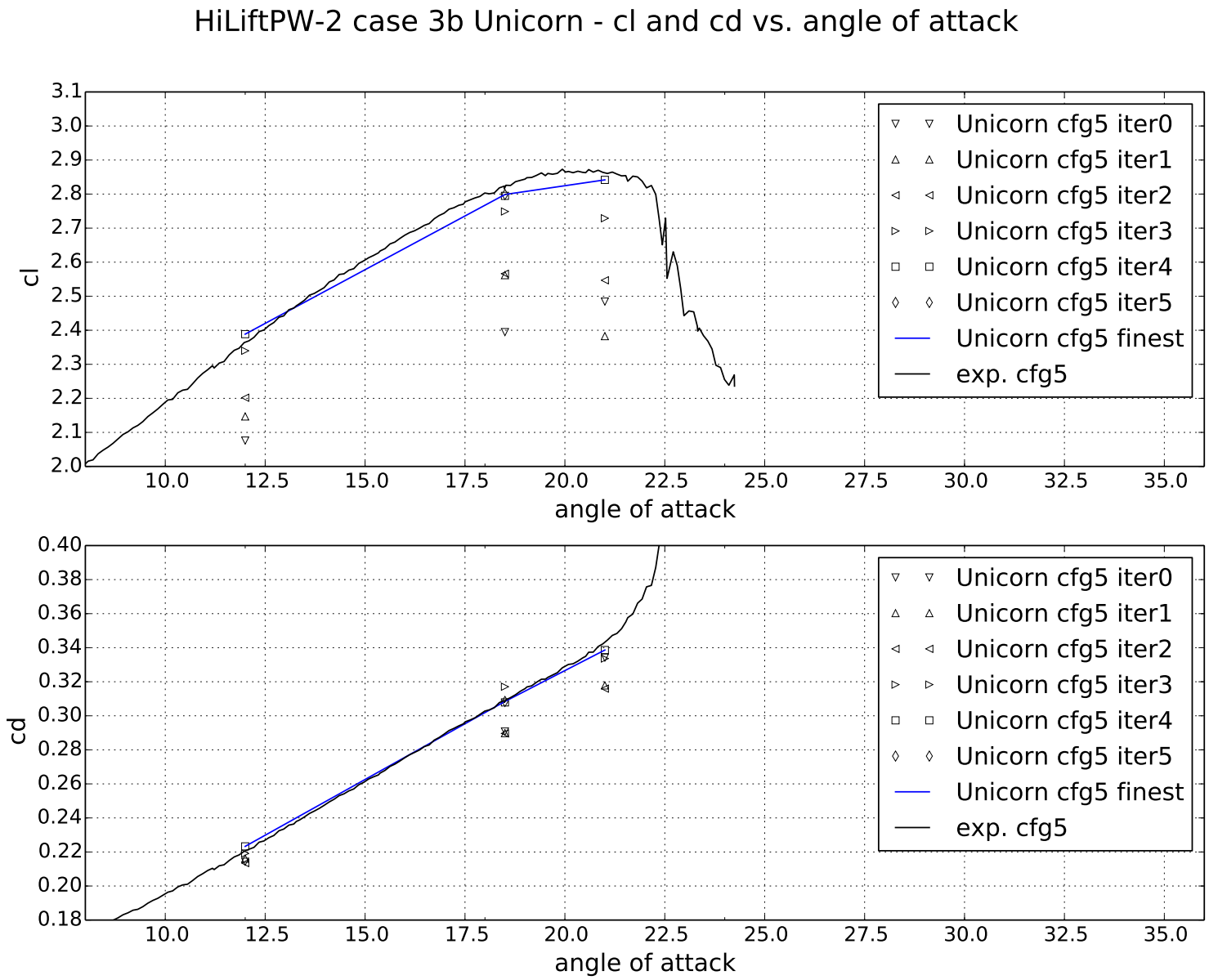
$$r(\hat{U}, \hat{v}) = (R(\hat{U}), \hat{v}) + (\delta R(\hat{U}), R(\hat{v})) = 0$$
$$\delta = h, \forall \hat{v} \in \hat{V}_h, \ \hat{U} \in \hat{V}_h$$

New "do-nothing" adaptive error control method

Directly use error representation: $M(\hat{e}) = r(\hat{U}, \hat{\phi})$
Error indicator: $\mathcal{E}_K = r(\hat{U}, \hat{\phi})_K$
Adjoint prob. autom. gen.: $r'(\hat{\phi}, \hat{v}) = M(\hat{v})$
Iteratively solve primal and dual problem, refine marked cells.

[Hoffman, Jansson, et. al., 2012 C&F], [Hoffman, Jansson, et. al. 2016, Encyclopedia of Computational Mechanics]

HiLiftPW-2 case: our results



Capture cl+cd to 2% of exp., no consistent stall. [Hoffman, Jansson, Johnson, JMFM, 2015] [Hoffman et. al., CMAME, 2015]

HiLiftPW-3: JSM case

Goals:

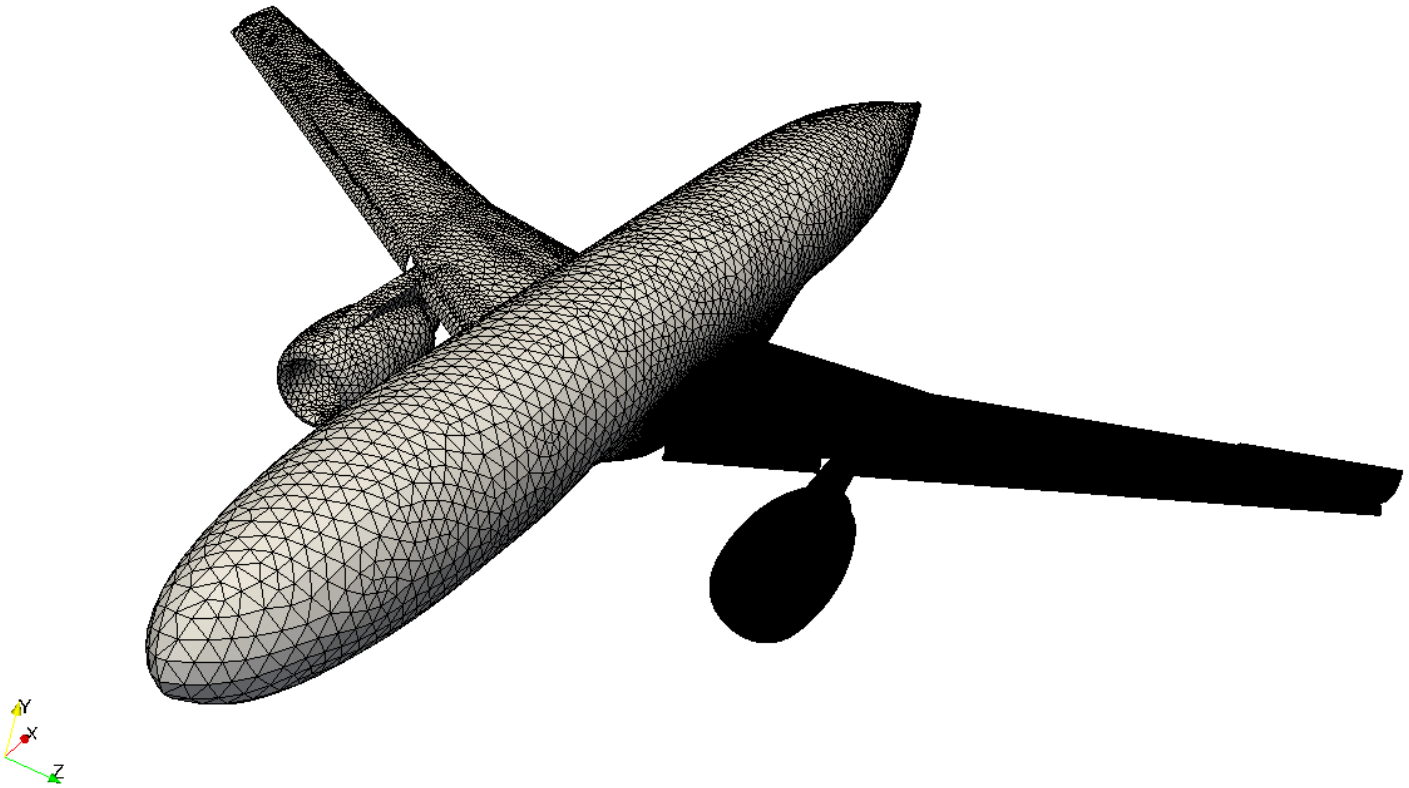
- Extend methodology to capture stall: add white noise "trip force" inspired by DNS, magnitude ca. 5% of $max(|\nabla p|)$.
- Test new do-nothing adaptivity.
- Parallel-in-time ensemble approach.
- Leverage higher efficiency of new developments (large timesteps enabled by a Schur preconditioning approach gives up to ca. 100x higher efficiency) and higher abstraction of implementation in FEniCS-HPC (weak form, projection of normals).

Brief overview of grid system(s)

Generated our own "starting guess" meshes for adaptive method with ANSYS. Target left half with goal functional. Resolve curvature, but otherwise as coarse as possible.

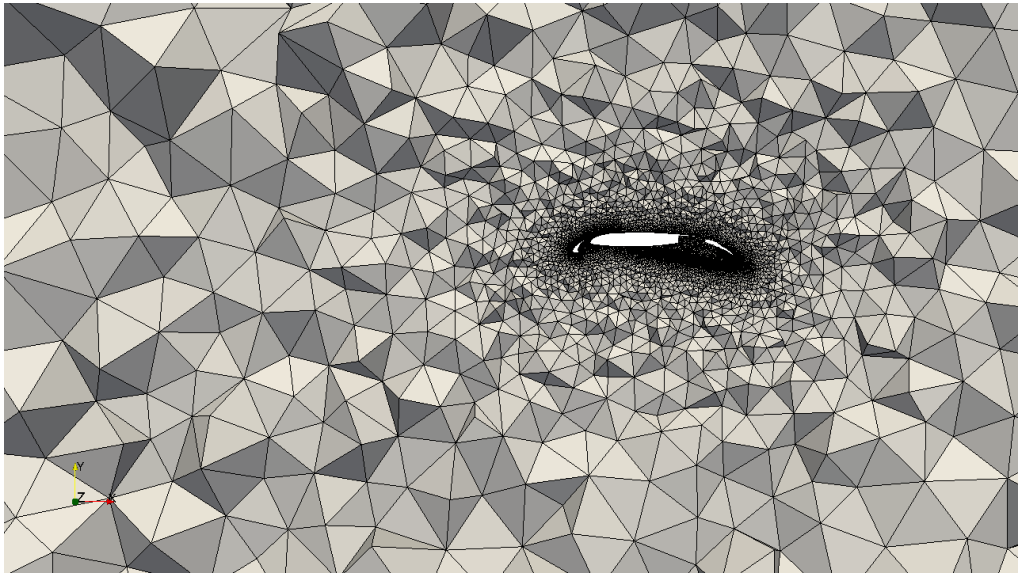
Presented results:
Case 2b no pylon/nacelle: 2.5M vertices
Case 2d with pylon/nacelle: 2.7M vertices

Use iterative goal-oriented adaptive method, refining 5% of the cells with highest error contribution every iteration.



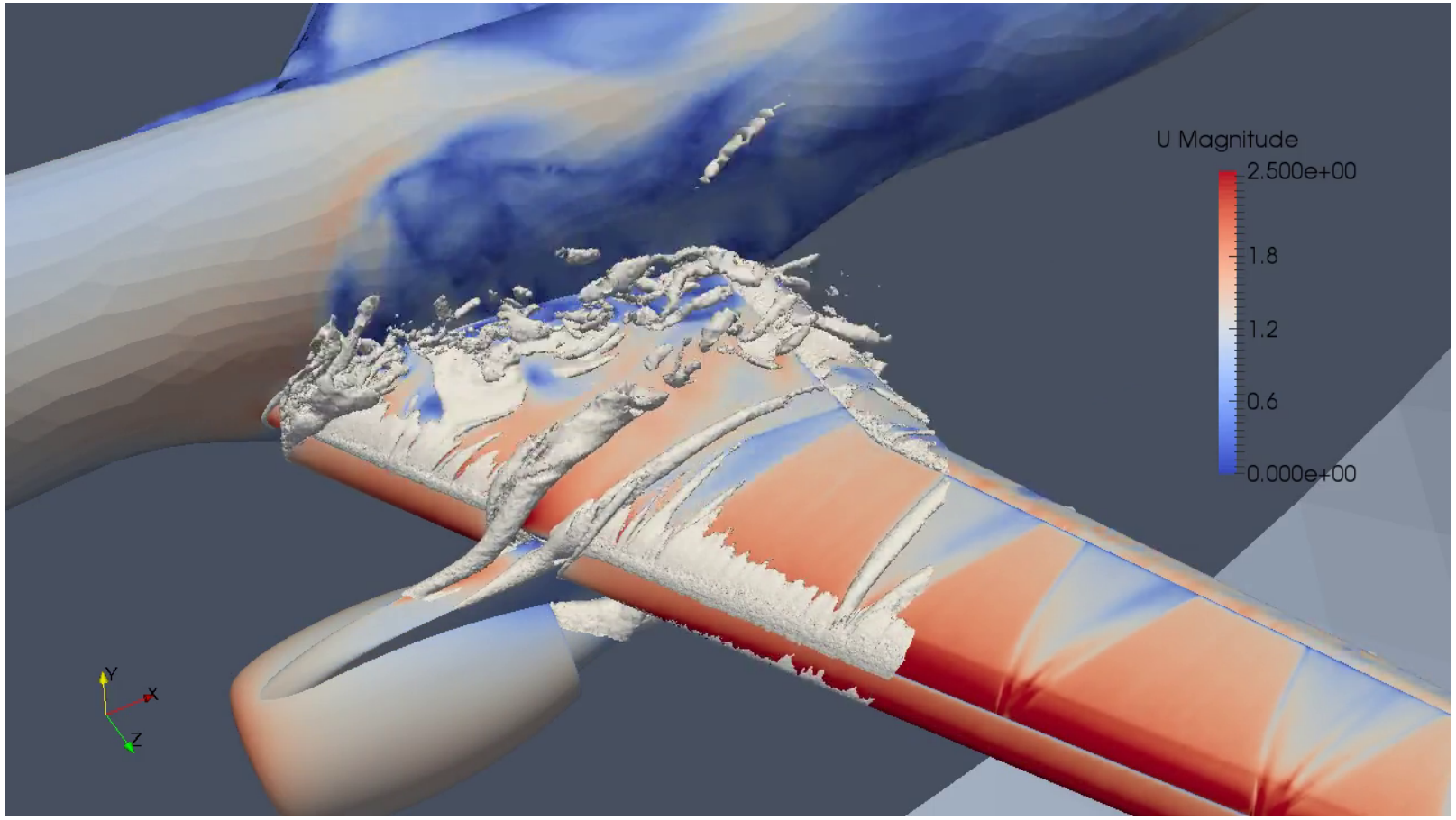
Brief overview of grid system(s)

Starting mesh has resolved geometry, coarse in the volume.



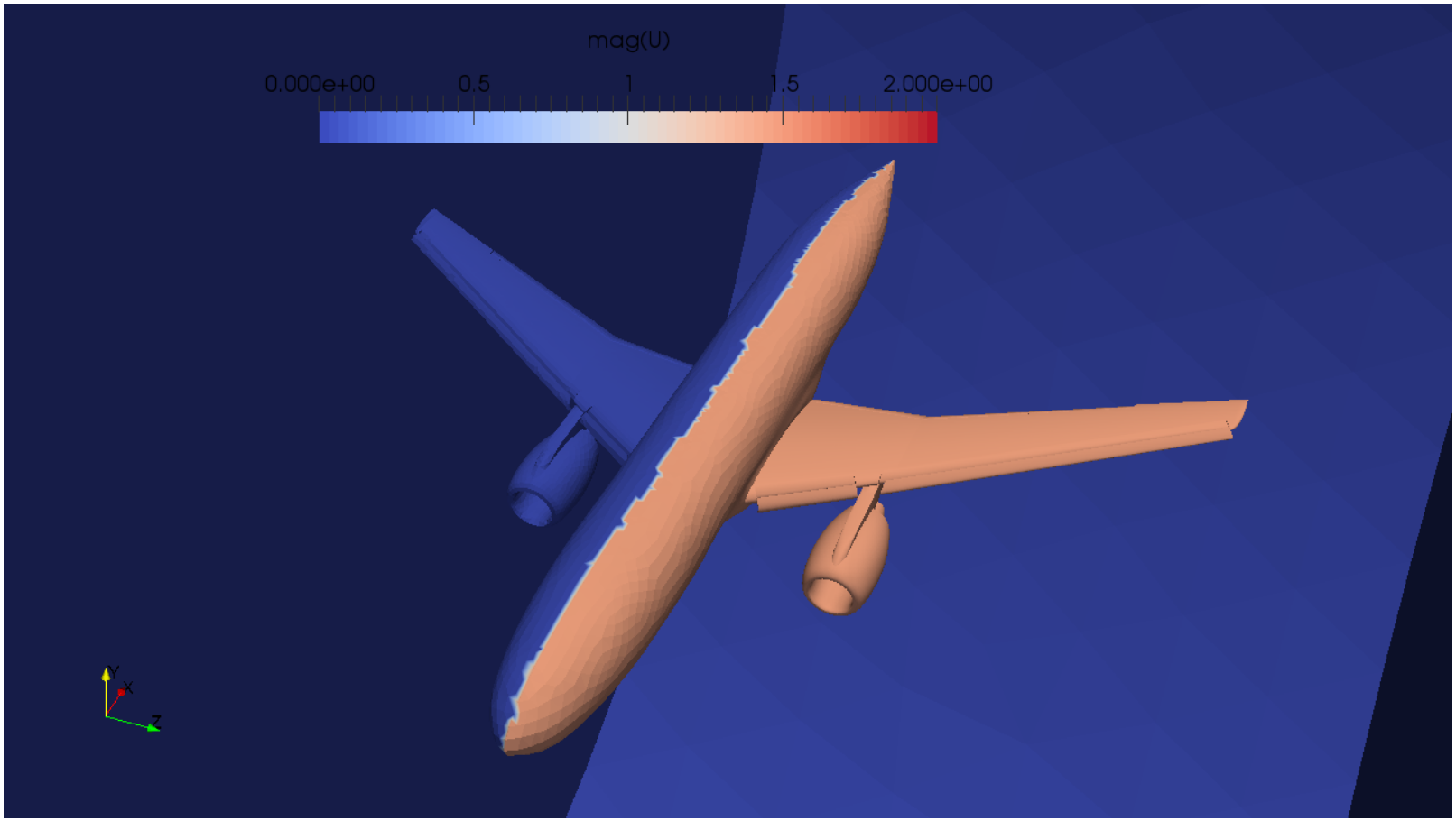
JSM case 2d (pylon-on)

Surface vel. $\alpha = 22.56$

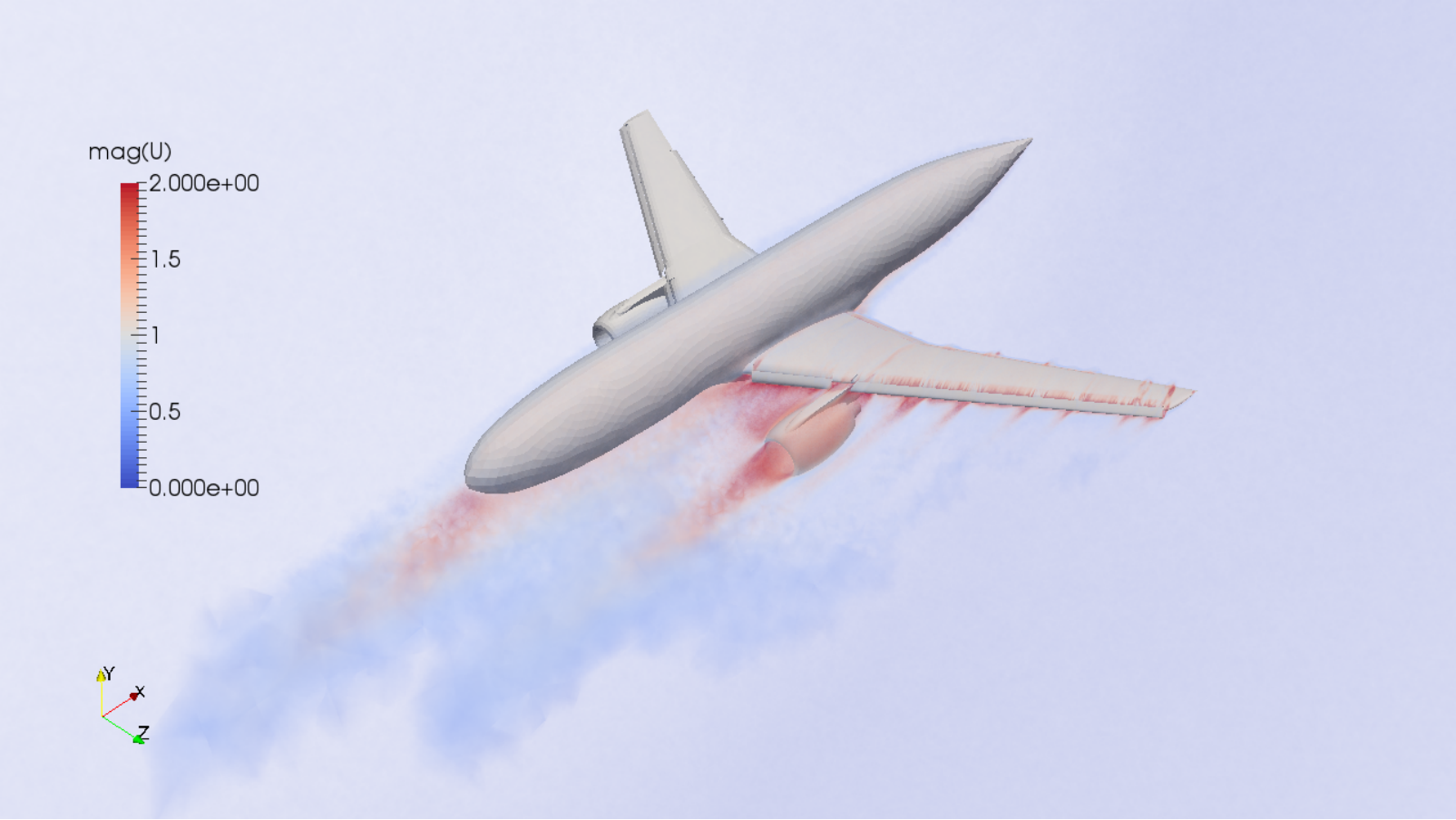


Goal functional for adaptivity

Target drag and lift for left half of aircraft.

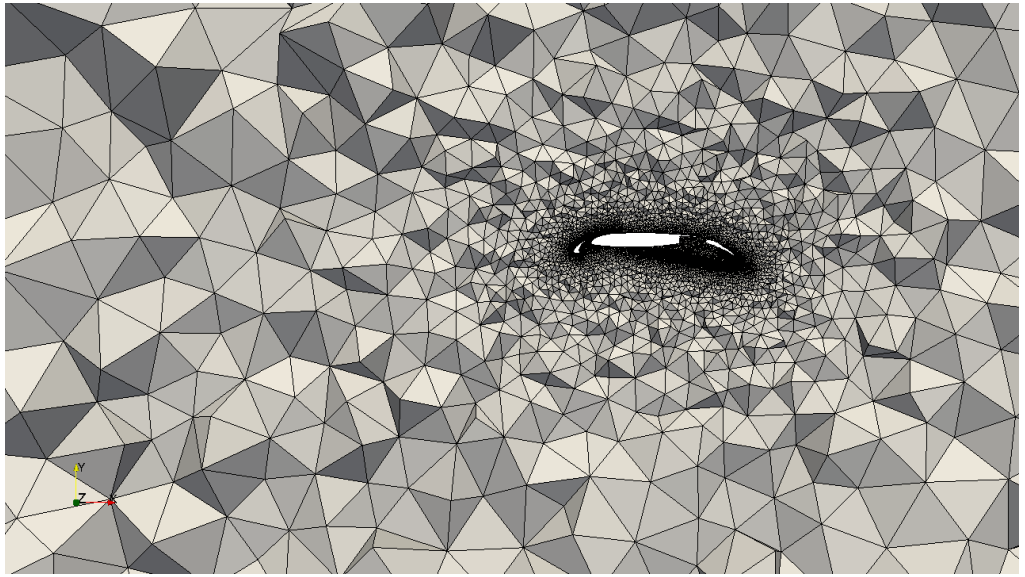


Adjoint velocity $\alpha = 4.36$

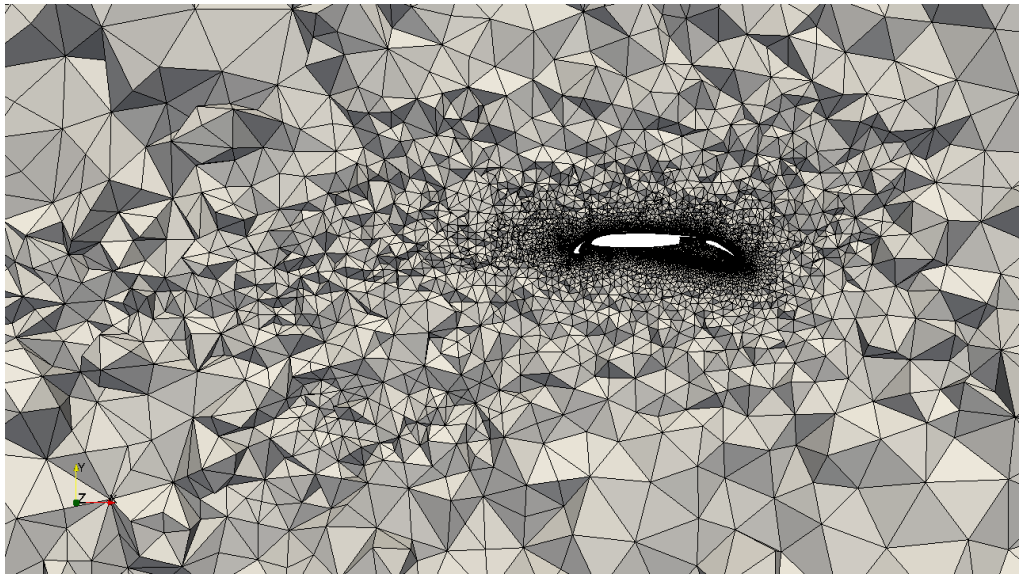


Goal-oriented adaptive refinement $\alpha = 4.36$ eta=TODO

Adaptive iteration 0 (starting mesh)

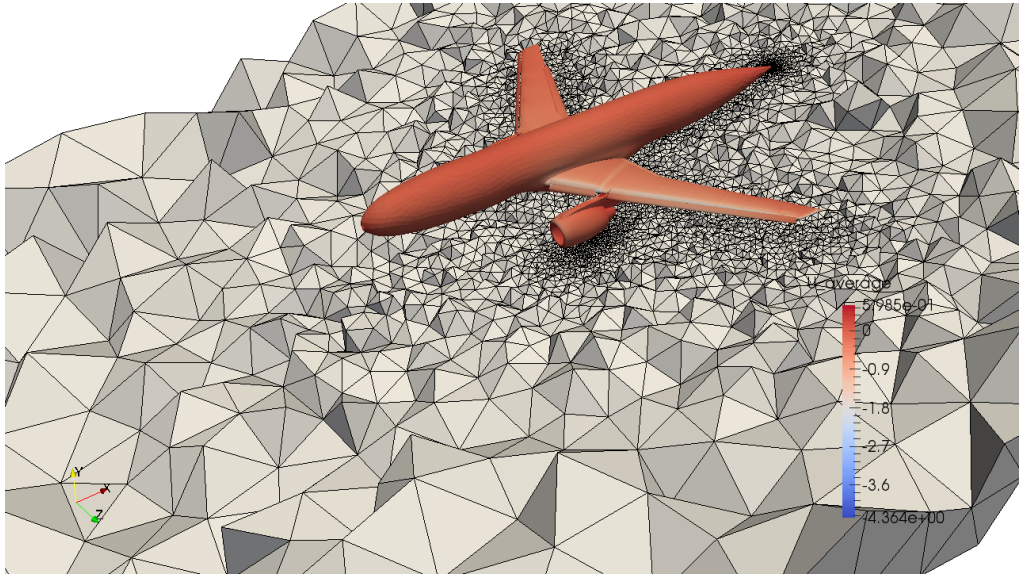


Adaptive iteration 3

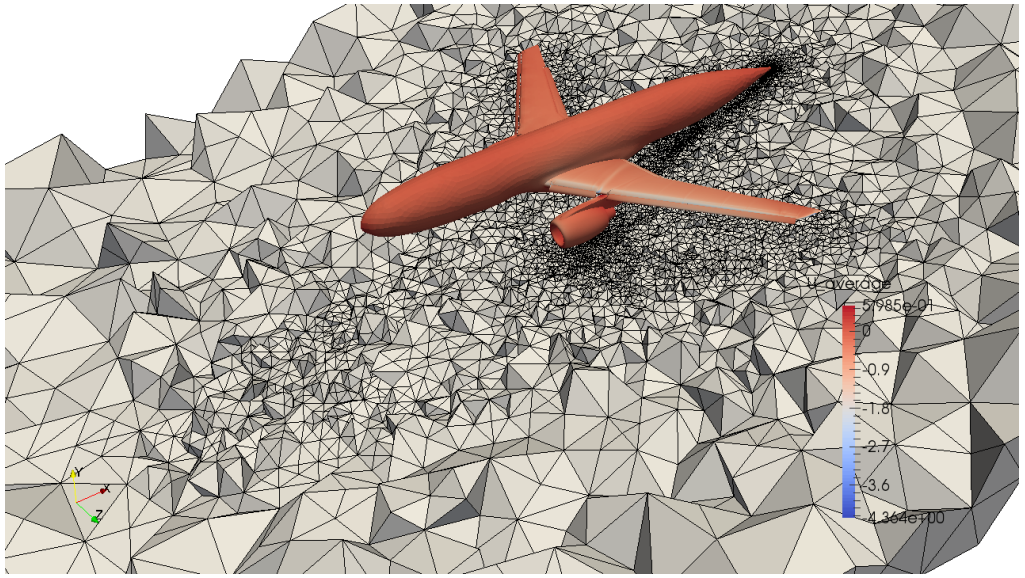


Goal-oriented adaptive refinement $\alpha = 10.48$ eta=TODO

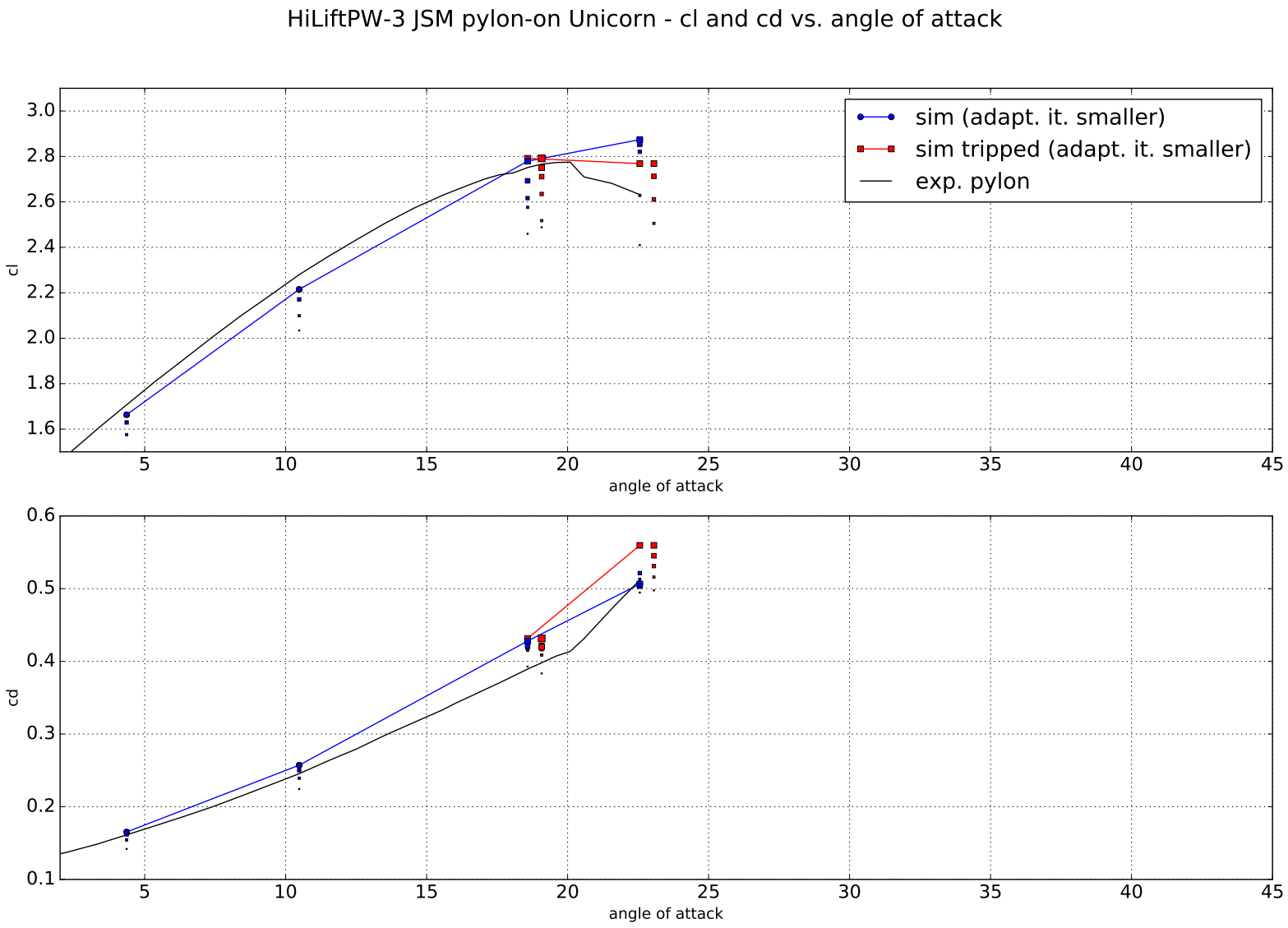
Adaptive iteration 0 (starting mesh)



Adaptive iteration 3

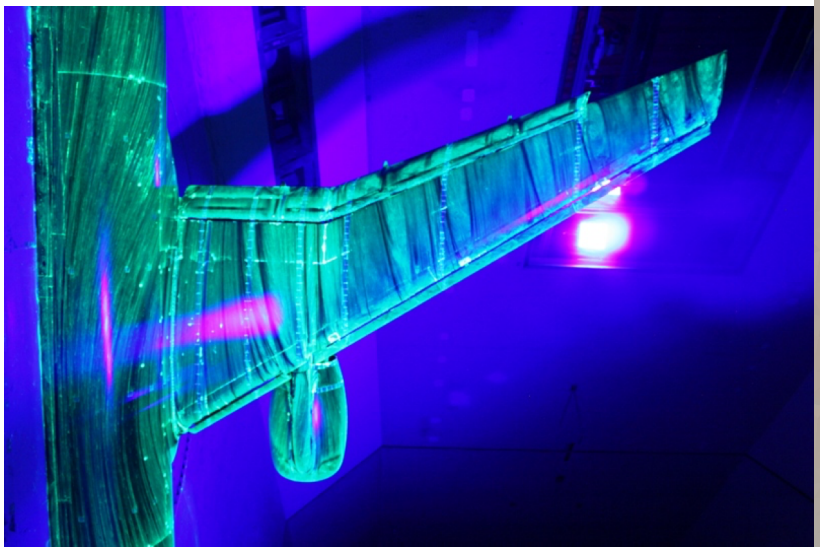


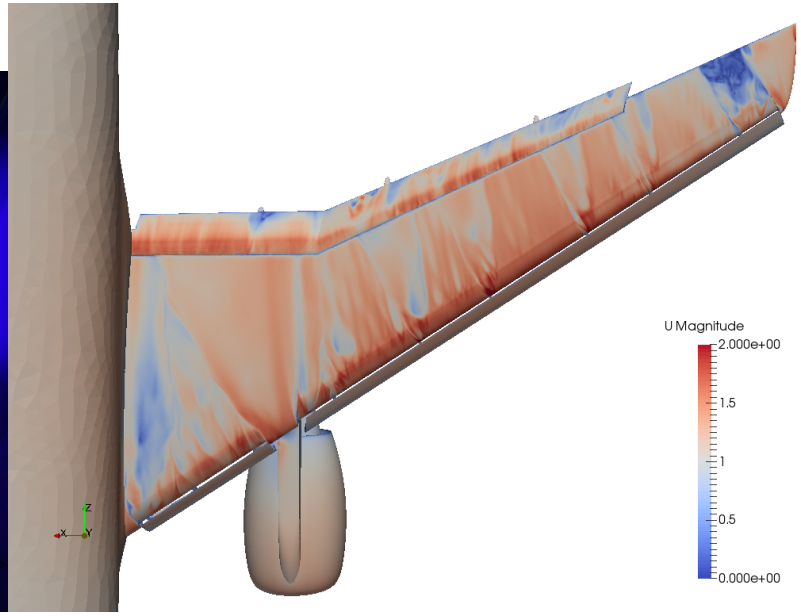
JSM case 2d (pylon-on) forces



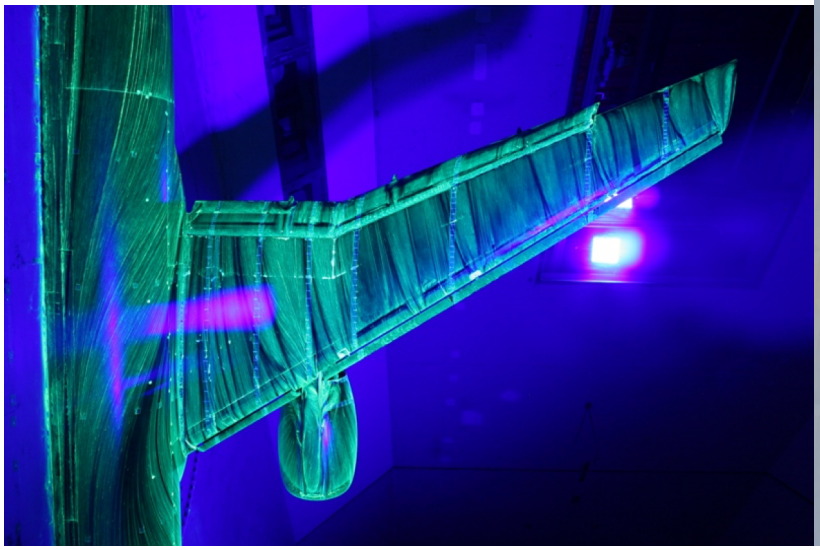
Surface velocity pylon-on

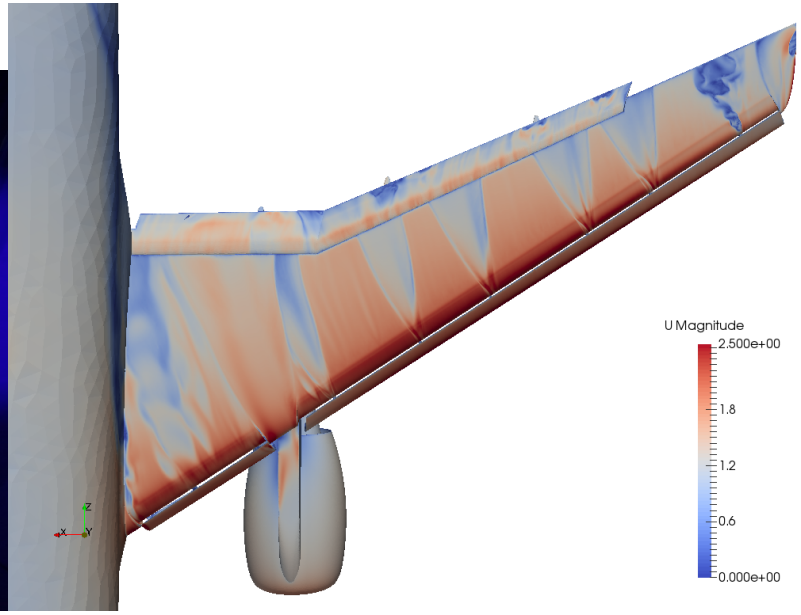
$\alpha = 4.36$





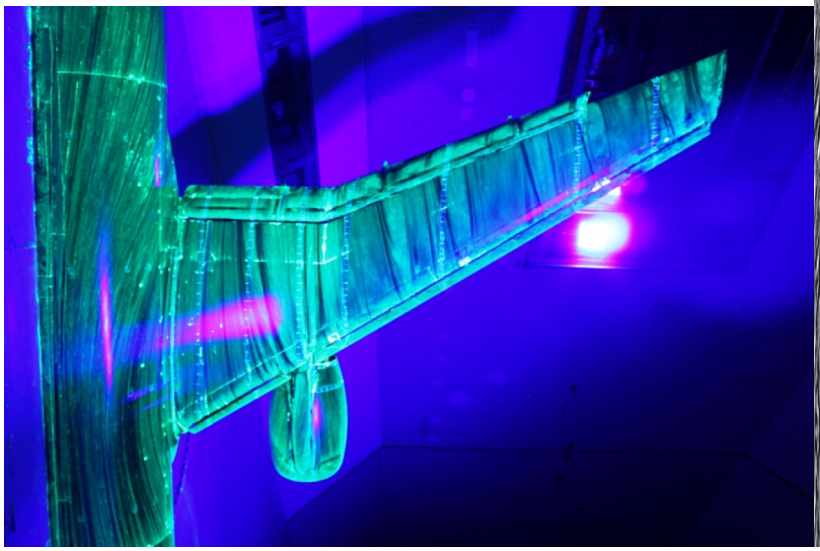
$\alpha = 18.58$

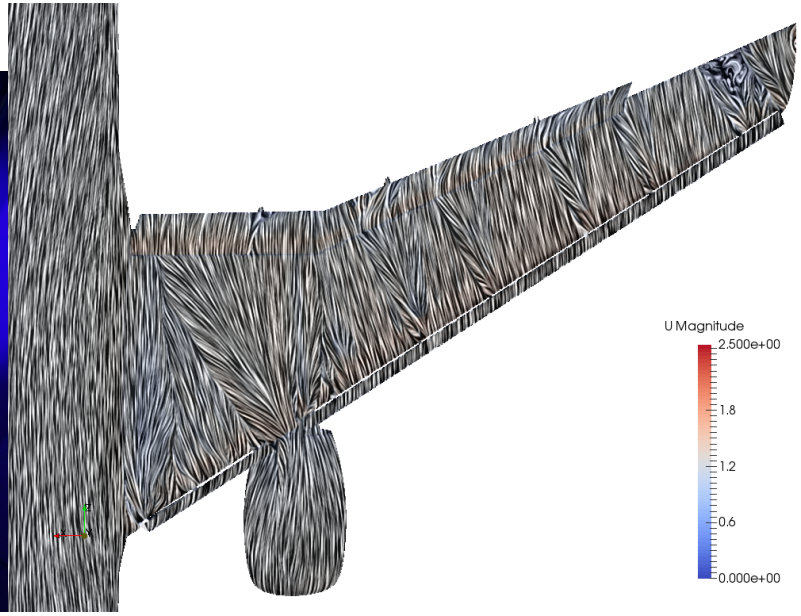




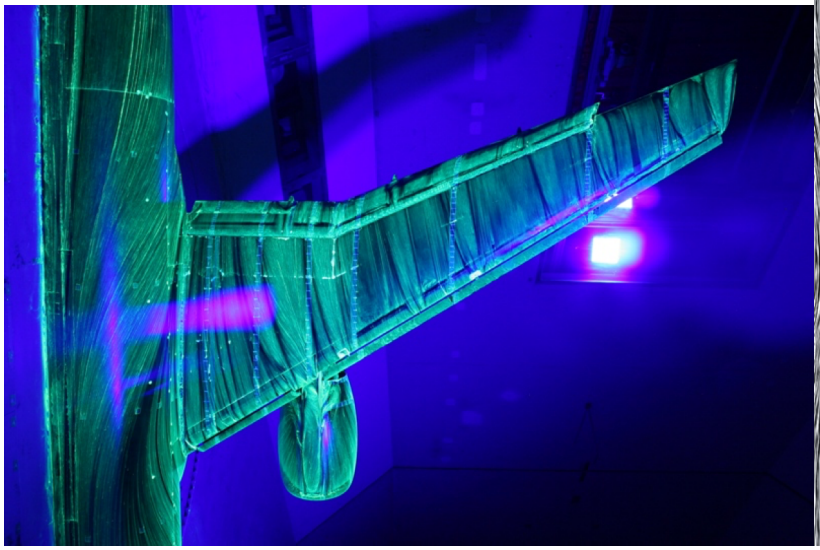
Surface velocity LIC pylon-on

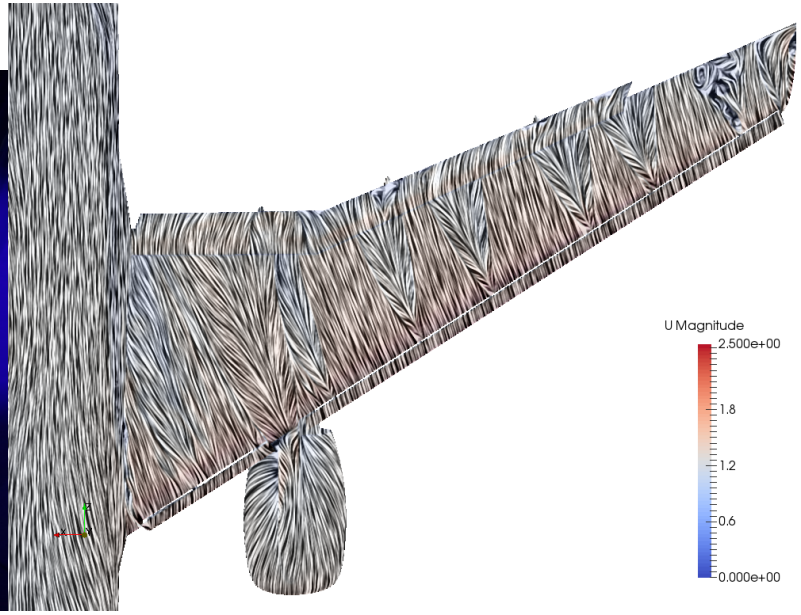
$\alpha = 4.36$





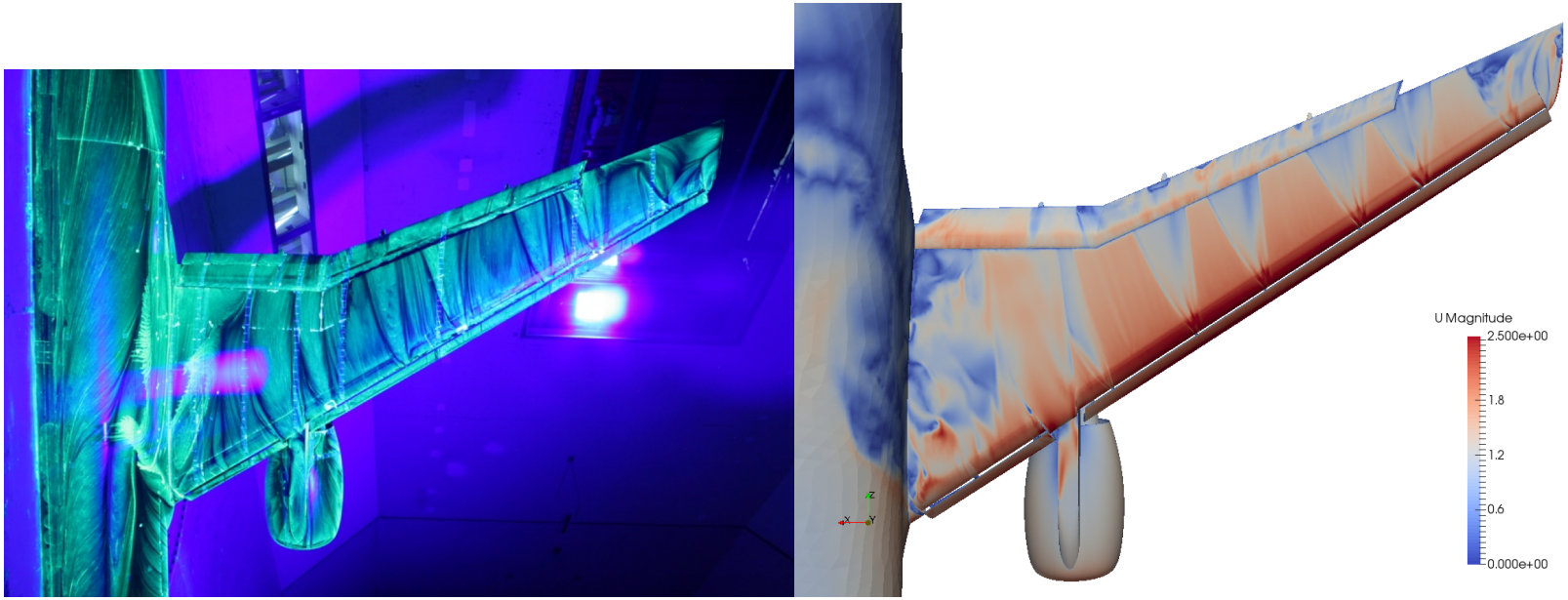
$\alpha = 18.58$





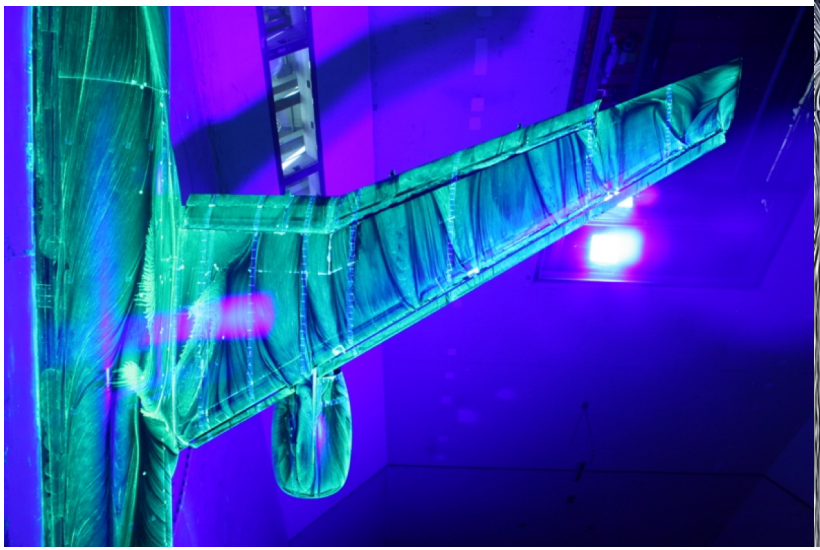
Surface velocity pylon-on

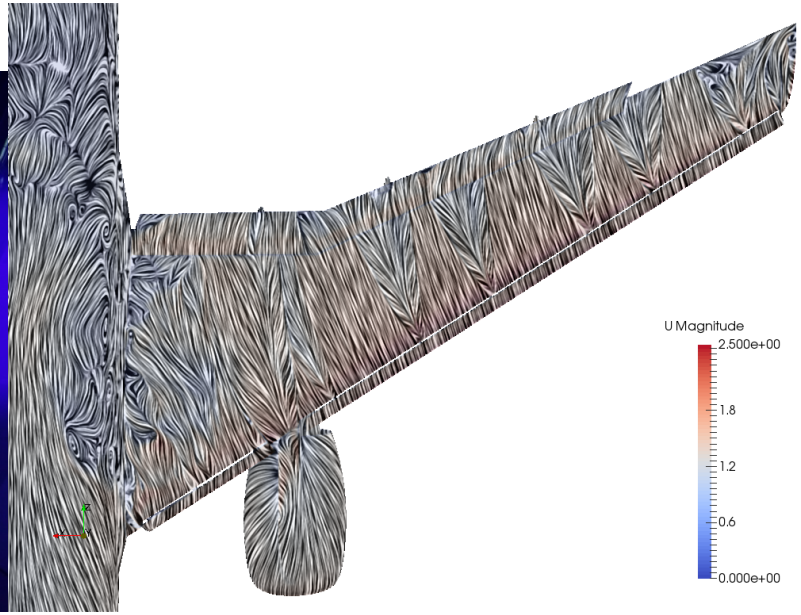
$\alpha = 21.57$ and $\alpha = 22.56$



Surface velocity LIC pylon-on

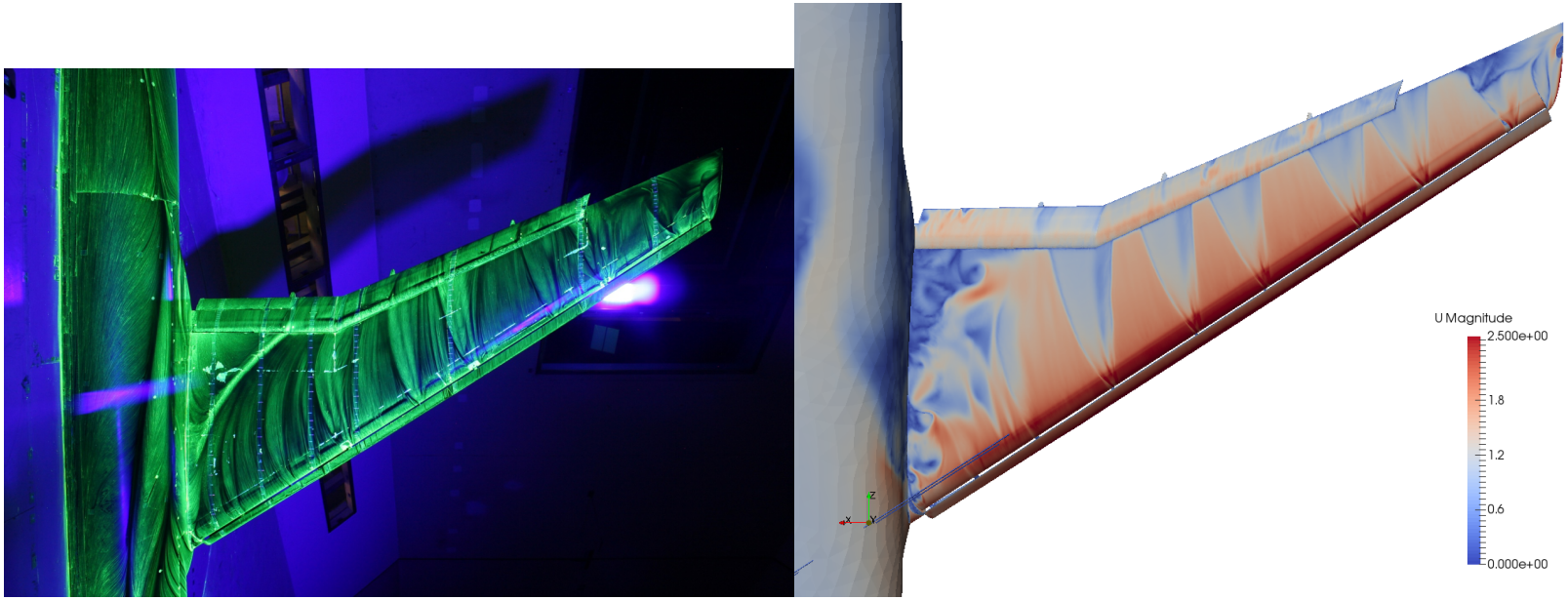
$\alpha = 21.57$ and $\alpha = 22.56$





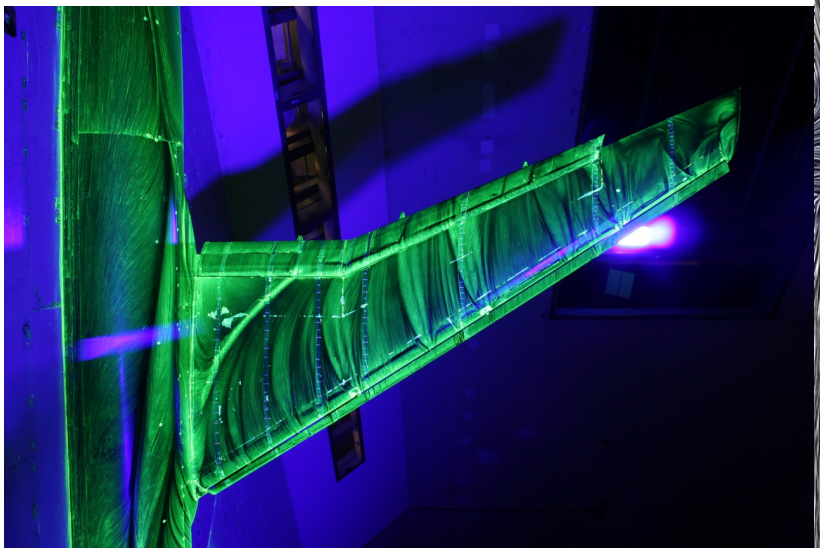
Surface velocity pylon-off

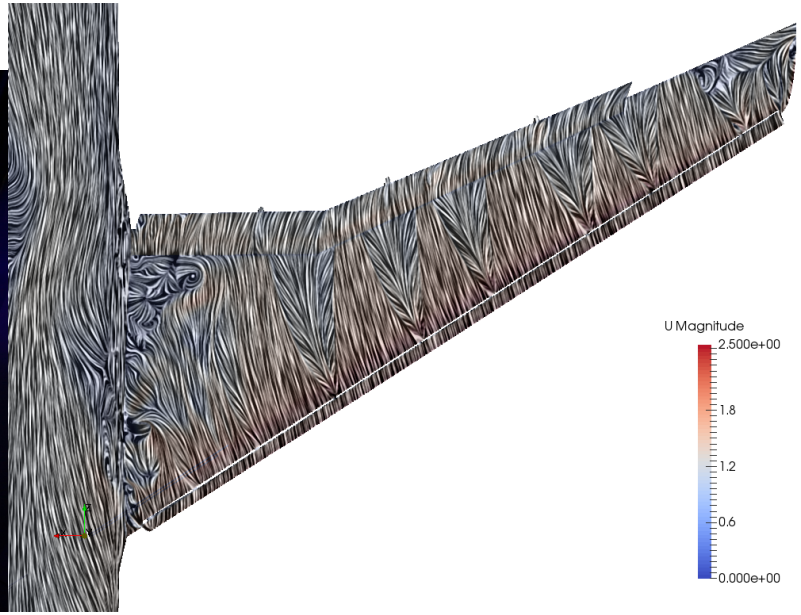
$\alpha = 21.57$ and $\alpha = 22.56$



Surface velocity LIC pylon-off

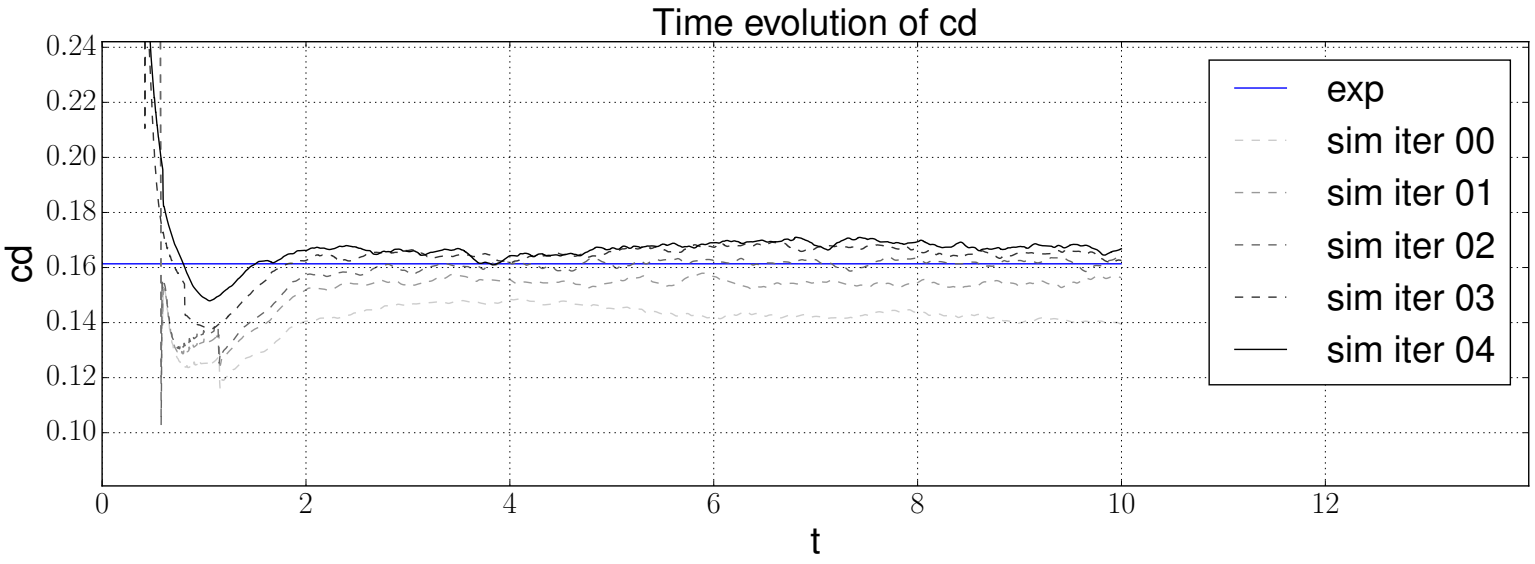
$\alpha = 21.57$ and $\alpha = 22.56$

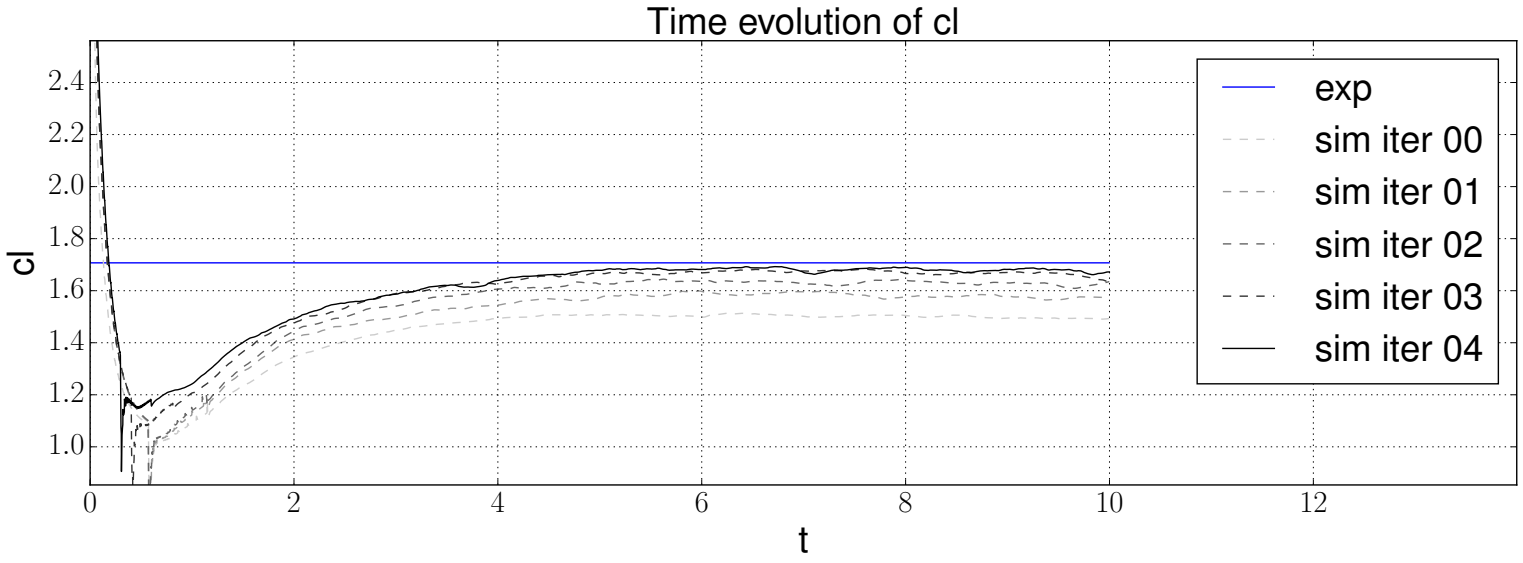




Adaptive mesh convergence

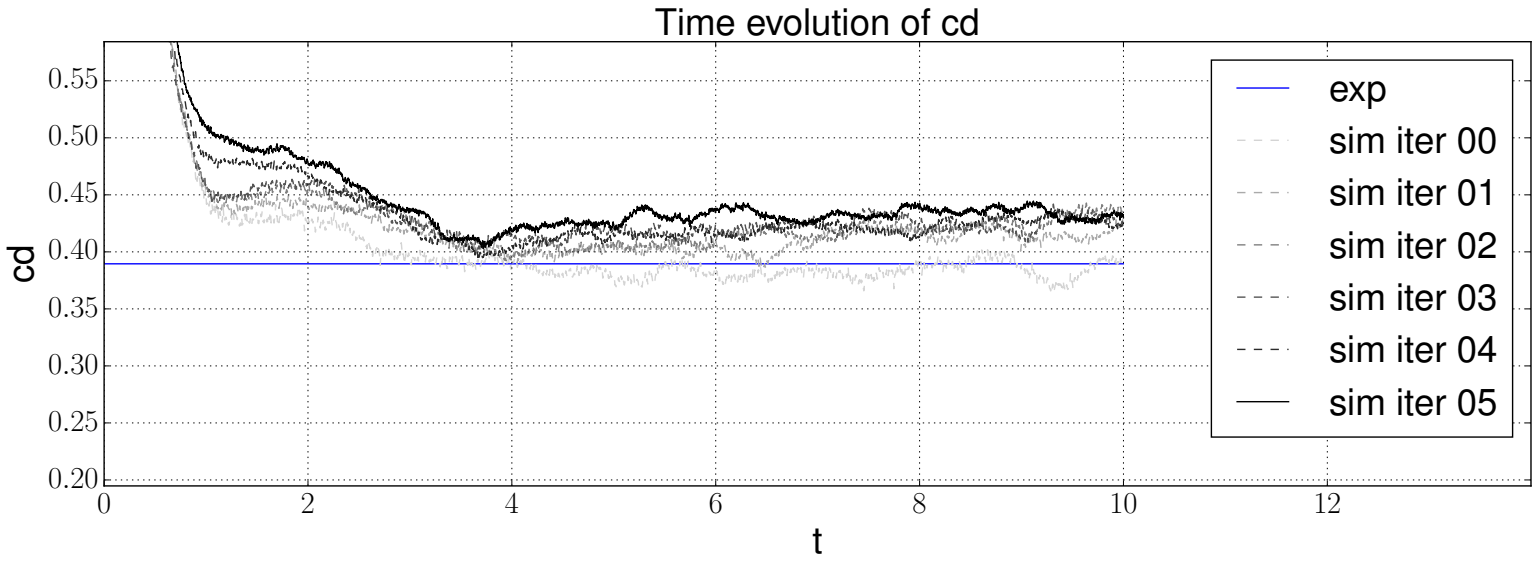
JSM pylon-on $\alpha = 4.36$

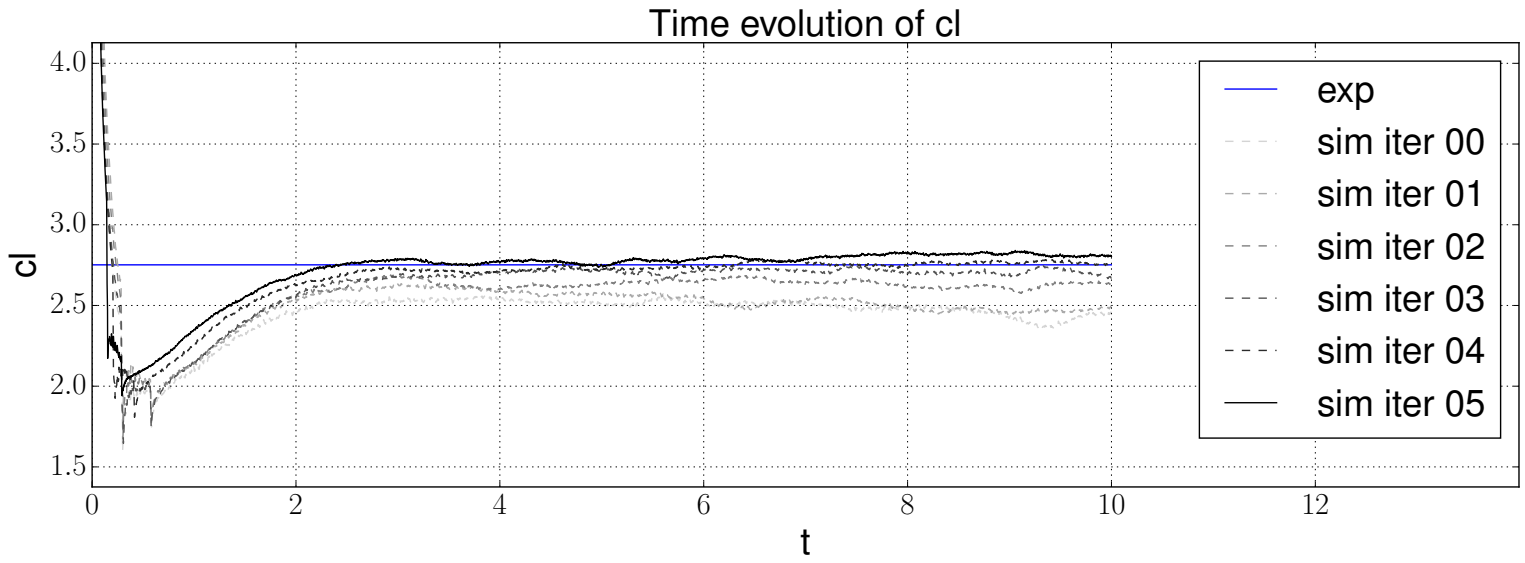




Adaptive mesh convergence

JSM pylon-on $\alpha = 18.58$

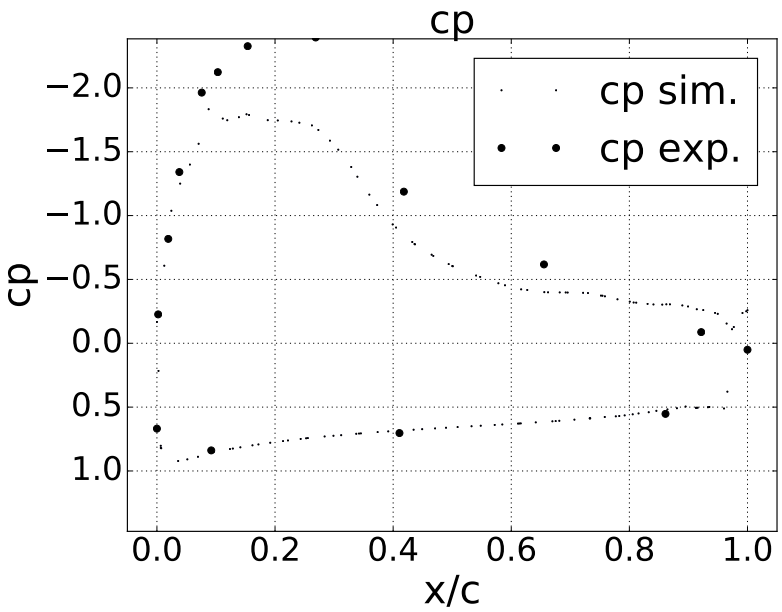




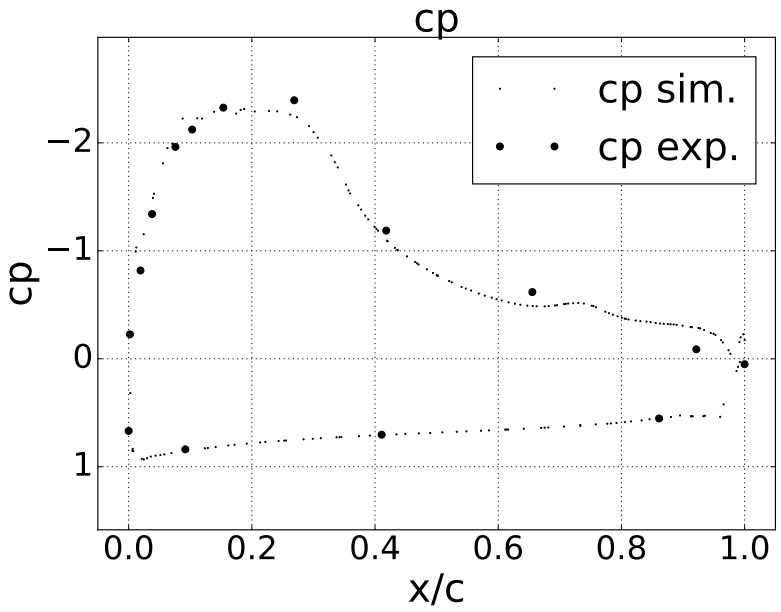
Illustrative cp ex.: adaptivity

JSM pylon-on, $\alpha = 4.36$, flap D-D
NB: Adaptivity targets mean quantity, not pointwise pressure

Adaptive iteration 0 (starting mesh)

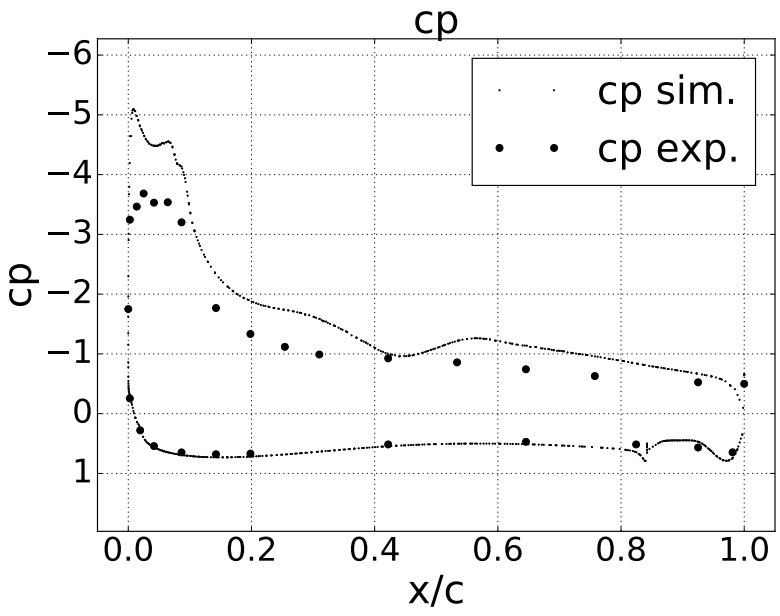


Adaptive iteration 3

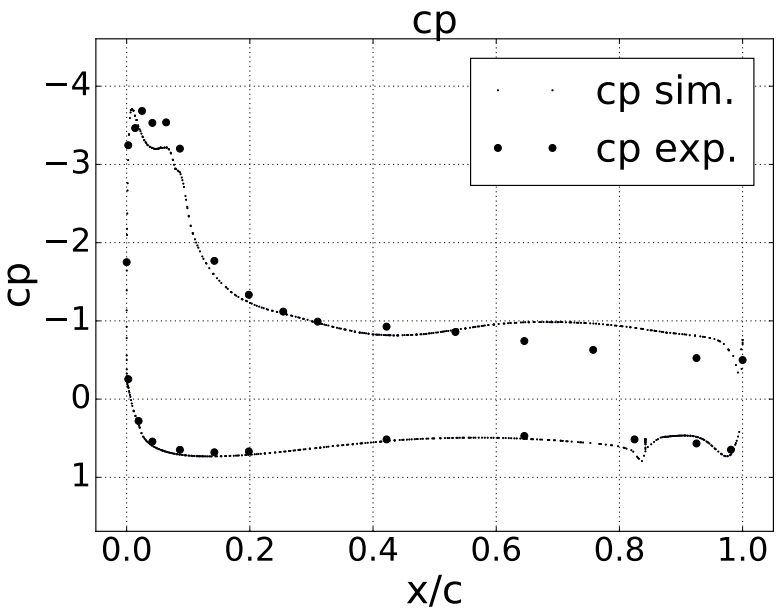


Illustrative cp ex.: stall

JSM pylon-on, $\alpha = 22.56$, wing B-B



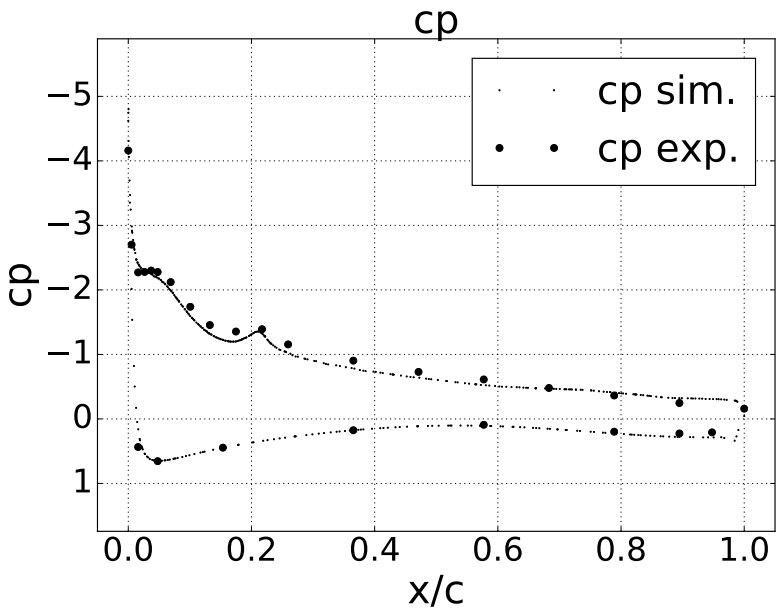
No trip-force



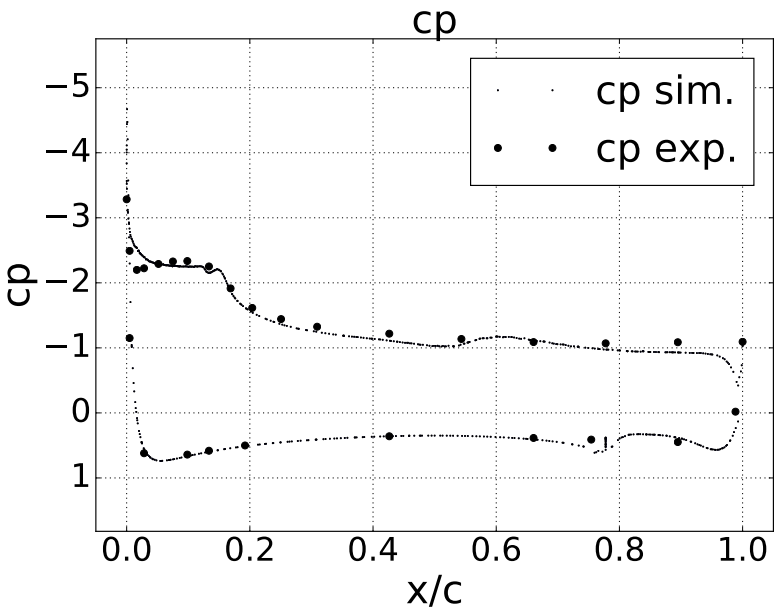
With trip-force

Illustrative cp ex.: outboard

JSM pylon-on, $\alpha = 4.36$



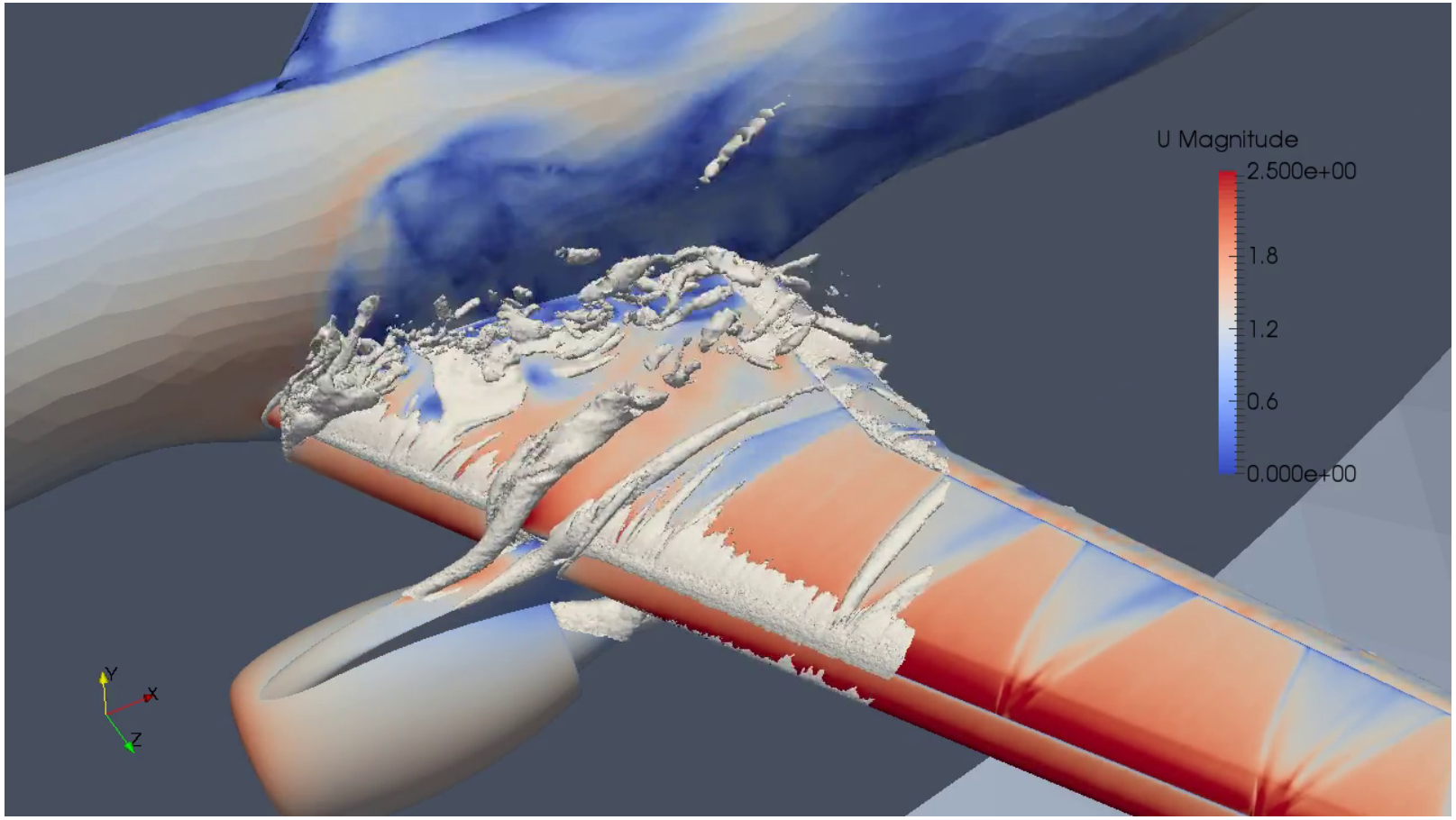
Outboard: wing H-H



Center: wing E-E

JSM case 2d (pylon-on)

Surface vel. + streamwise vorticity $\alpha = 22.56$



Summary

- Results for pylon-on/pylon-off similar, within comp. tol.
- Mesh convergence for new "do-nothing" goal-oriented adaptive method for $\alpha = 4.36, 18.58, 22.56$ to ca. 5% for lift and drag ($\alpha = 4.36$), drag ca. 10% over-predicted for higher angles.
- Capture $c_{L,max}$ to within 2% and stall mechanism with "trip force"- large-scale separation from leading edge of wing-body junction consistent with exp. Negl. effect at pre-stall angle(s). Will compute intermediate angle $\alpha = 21.57$, sensitivity of trip-force magnitude, more adapt. iters.
- Capture relevant qualitative and quantitative results to 5-10% of exp. Parameter-free, no explicit turb. modeling, slip BC.
- We still have to generate starting mesh with standard mesh generator, ANSYS here. Time-consuming, error-prone.
- Adaptivity and large timesteps allow high efficiency: meshes of ca. 5M vertices, wall clock 10h using 1k cores.

Acknowledgements:

MINISTERIO DE ECONOMÍA Y COMPETITIVIDAD

European Commission

VINNOVA

eunison

EXCELENCIA SEVERO OCHOA

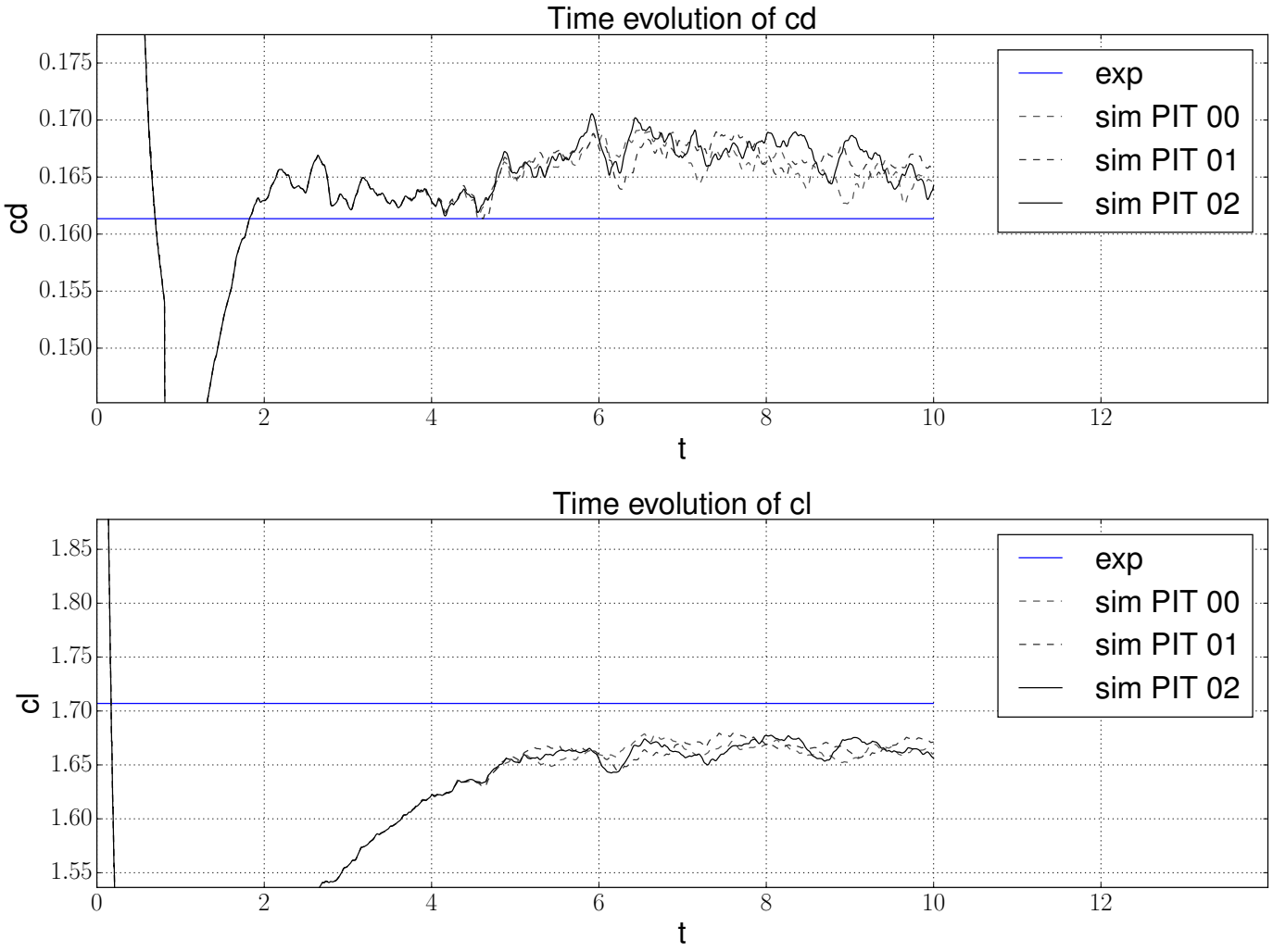
EUSKO JAURLARITZA GOBIERNO VASCO

PRACE

SNIC

Parallel-in-time

Ensemble approach of partitions of time interval for bluff body, here $\alpha = 4.36$



[Jansson, Hoffman, preprint, 2017]