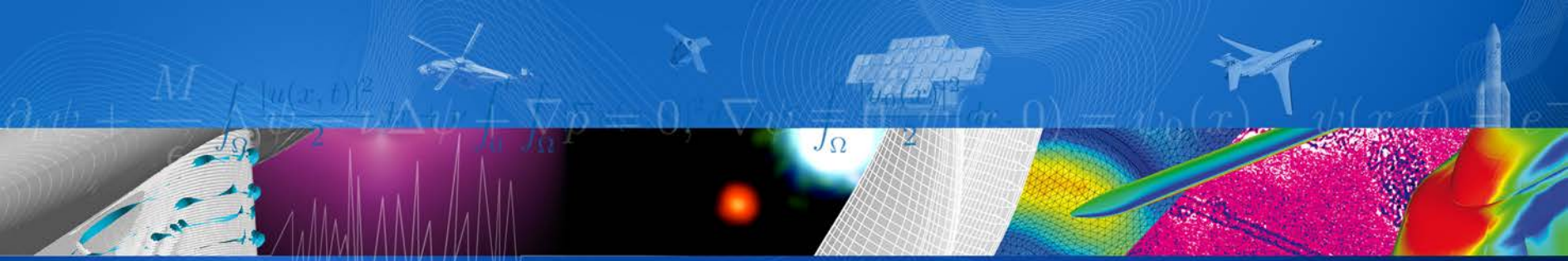


A review of Overset Grid Technology at ONERA

Stéphanie PERON – CFD Department

stephanie.peron@onera.fr

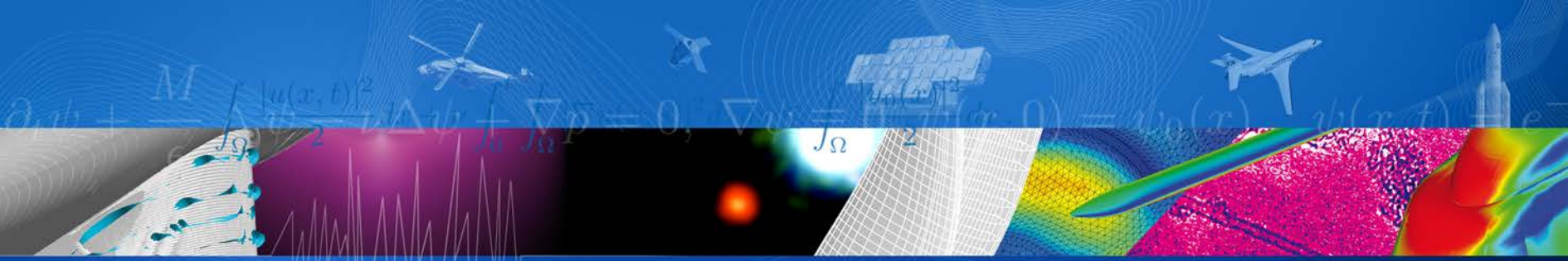
13th Overset Grid Symposium

October, 17-20, 2016

Future of Flight Aviation Center, Mukilteo, WA, USA

ONERA

THE FRENCH AEROSPACE LAB



Contributors

CFD Department:

C. Benoit, S. Landier, S. Péron, P. Raud, T. Renaud

Applied Aerodynamics Department:

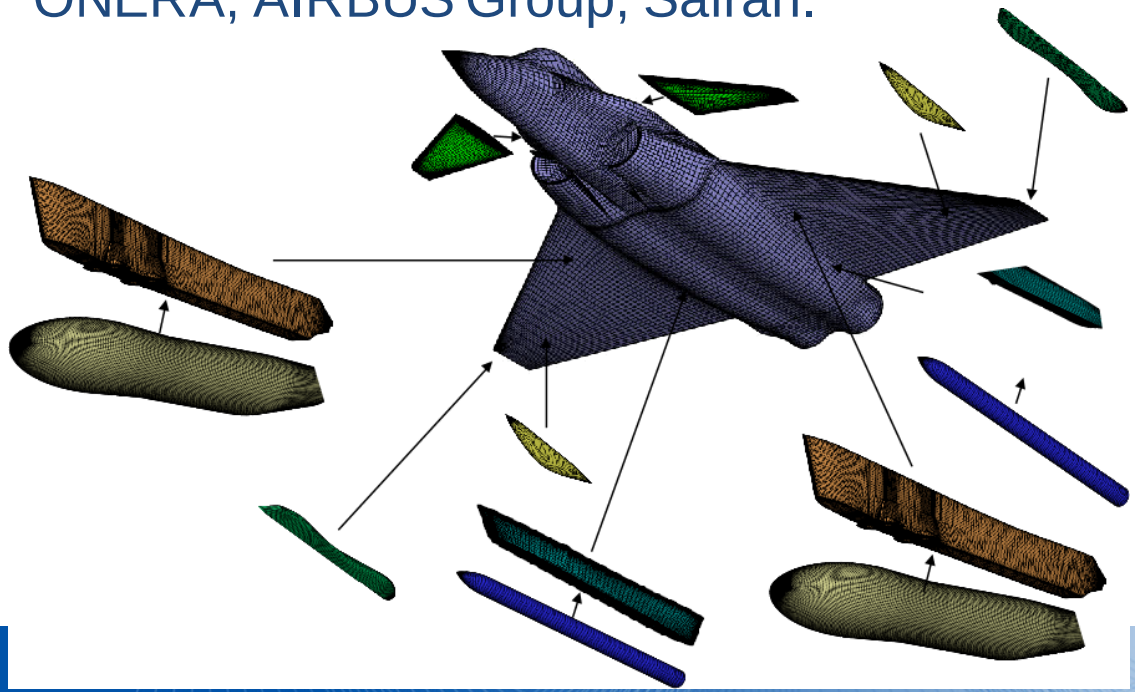
G. Billonnet, J.-C. Boniface, L. Castillon, C. François, J.-L. Hantrais-Gervois, D. Hue, A. Le Pape, L. Wiart

ONERA

THE FRENCH AEROSPACE LAB

Introduction

- Overset grid technology developed at ONERA since the 1990s
- 1990s: within FLU3M code for missiles applications (Guillen et al.)
- 2000s: within elsA code for helicopter flow simulations (ONERA-DLR CHANCE program)
- 2010s: overset grid preprocessing achieved by Cassiopee package, update of the solution at interpolated points within the solver (*elsA*)
- Mainly used for simulations with structured solver of *elsA* by ONERA, AIRBUS Group, Safran.



Dassault Rafale
Courtesy of P. Thorigny
ONERA/DAAP/MHL

Cassiopee & overset grids: initial motivation

- Cassiopee modules initiated with overset grid pre and post-processing functions
  - Overset grid assembly inside the elsA CFD solver became tricky
- Advantages:
  - Simplification of the overset grid assembly
  - Other codes/purposes possible (e.g. immersed boundaries…)
  - Check the assembly (orphan point detection,…)
  - If the mesh is modified: same script can be re-run

Overset grid technology with Cassiopee

- Pre-processing tools:
  - Hole-cutting techniques
  - Doubly defined boundary conditions
  - Overset grid connectivity computation
- Mesh generation tools
- Post-processing tools

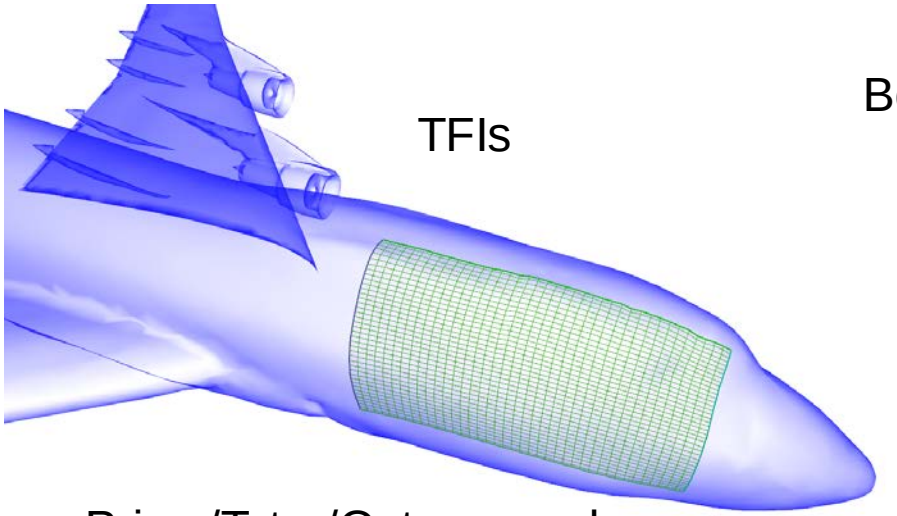
Cassiopee: contents

- Cassiopee: set of Python modules
  - CGNS/Python data representation (M. Poinot)
  - Independent compilation/setup of modules
  - Capitalizes CFD pre and post-processing methods
  - Pre/post-processing of *elsA* CFD solver and other in-house codes
  - Since 2008
  - Upgradeable/collaborative
- Distribution:
  - Fully distributed and installed with elsA solver
  - > 95% of open source code (since 12/2013) under GPL3 license

Réf: C. Benoit, S. Péron, S. Landier, « Cassiopee: a CFD pre- and post-processing tool », Aerospace Science & Technology, vol 45, pp. 275-283, 2015

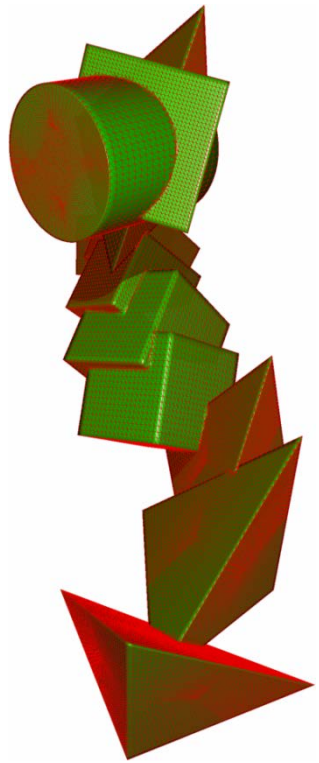
Cassiopee package: Generator module

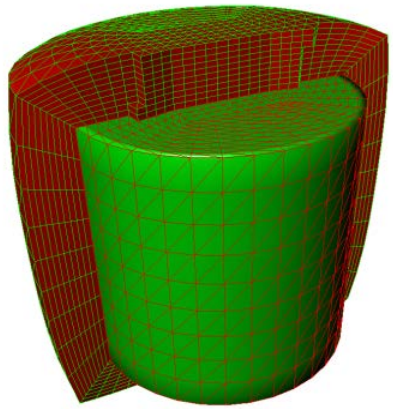
- Mesh generation functions (2D/3D)



TFIs

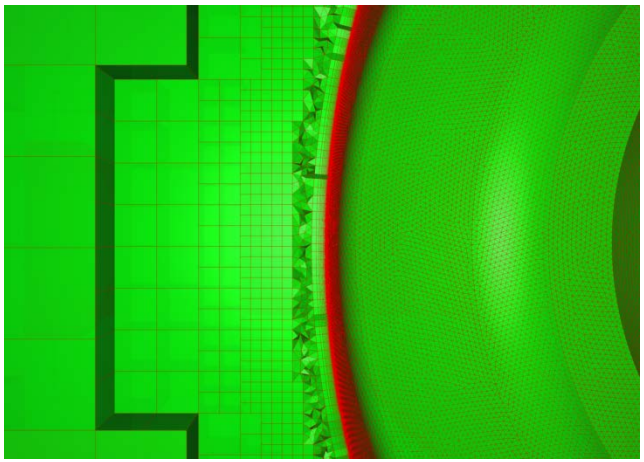
Boolean operators





Extrusion

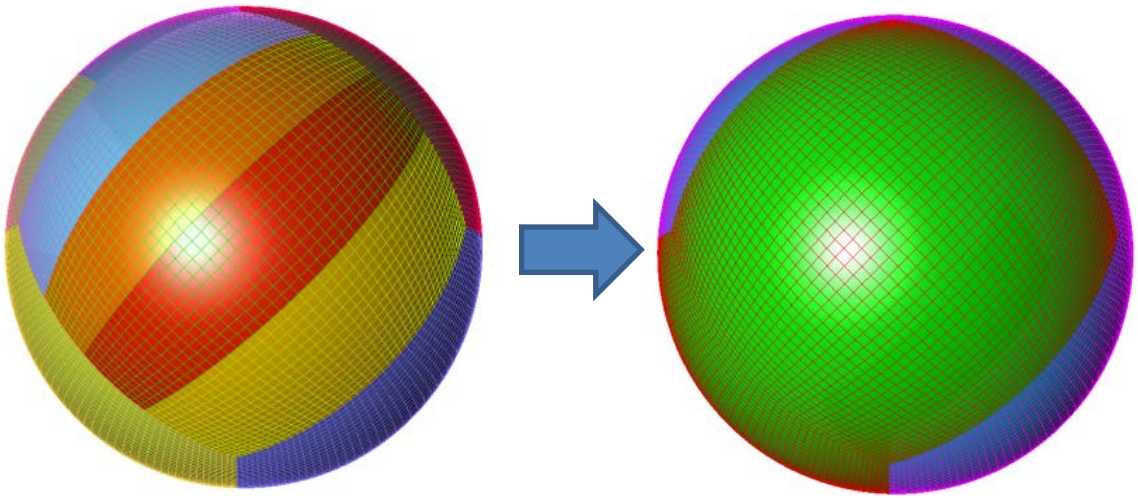
Prism/Tetra/Octree mesh



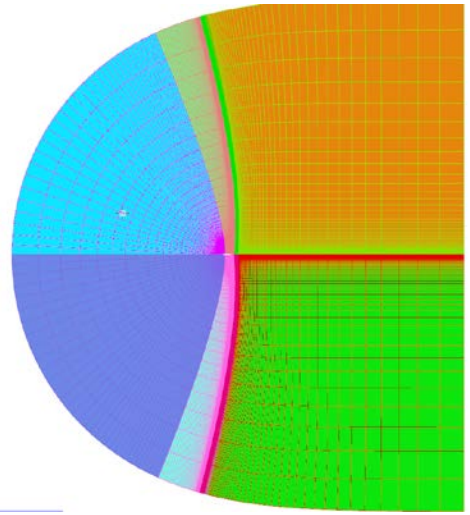
Cassiopee package: Transform module

- Mesh transformations

Merging



Splitting

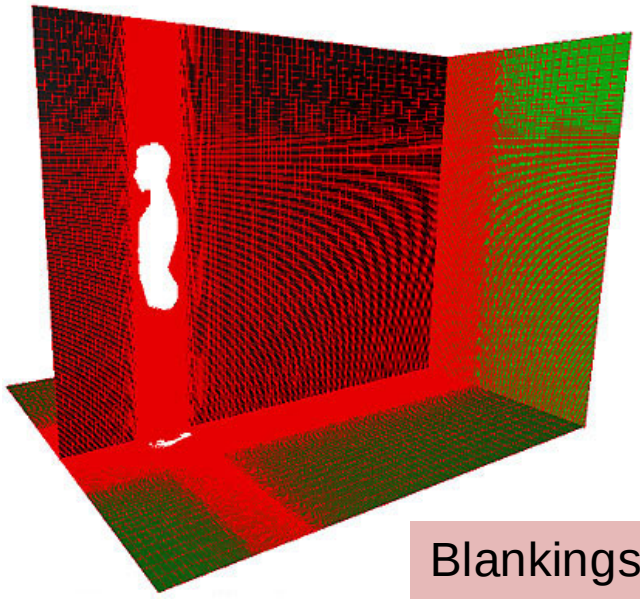


Modification of the mesh + BCs+ connectivities + solutions

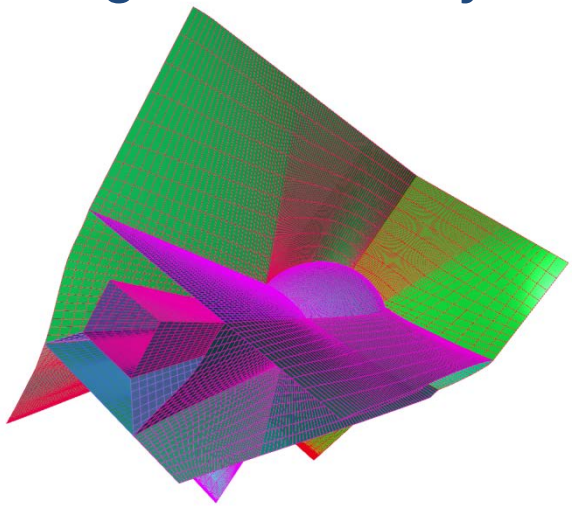
Rotations, symmetries, projections, reordering…

Cassiopee package: Connector module

- Connectivities between grids & overset grid assembly



Blankings

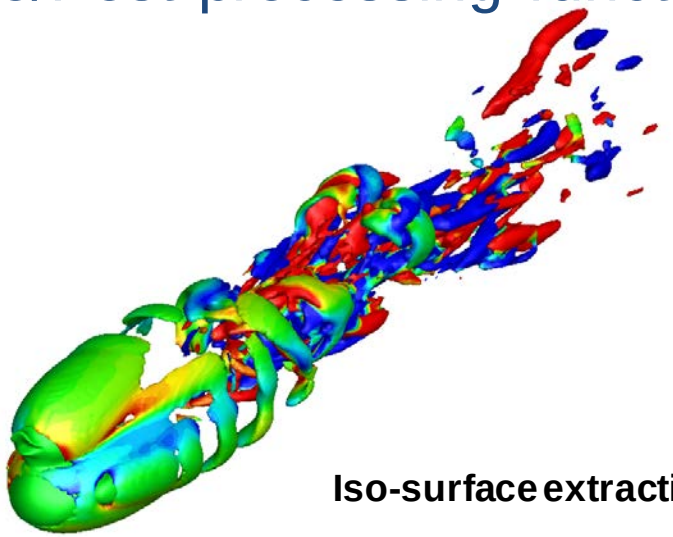


Abutting grid connectivity

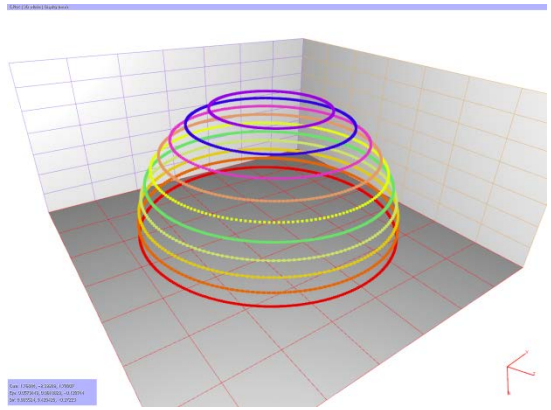
and overlap optimization, Chimera donor cell search, IBM preprocessing,…

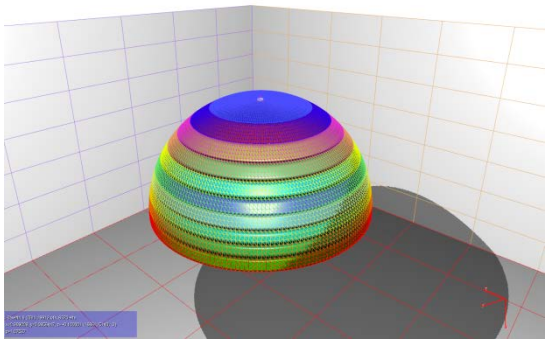
Cassiopee package: Post module

- Co/Post-processing functions

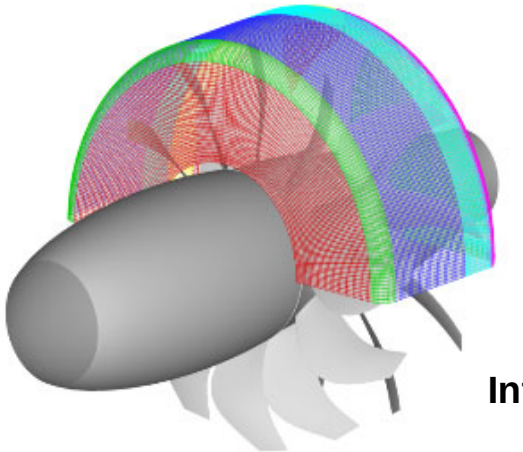


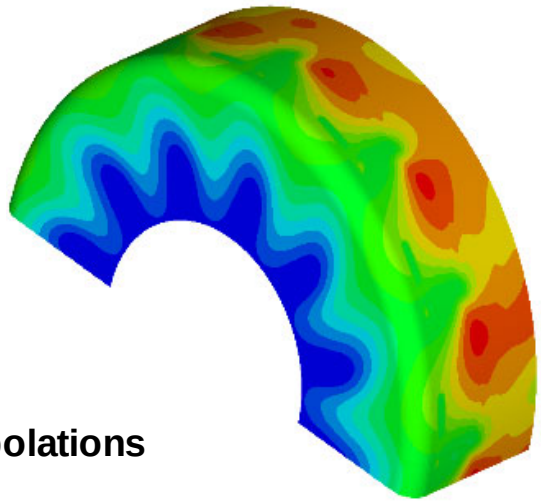
Iso-surface extraction





Integrations on curves/slices/surfaces

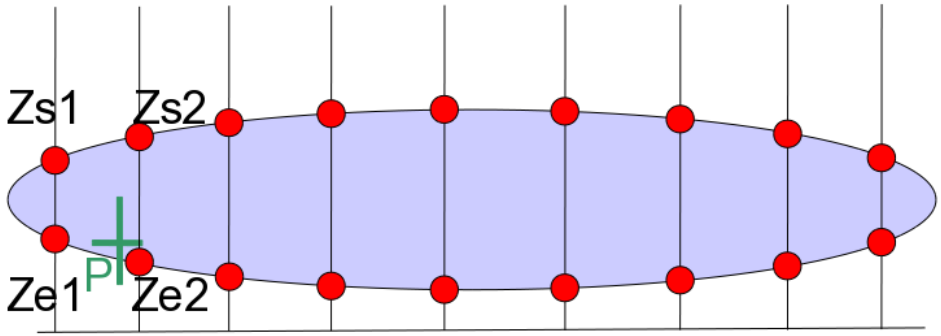




Interpolations

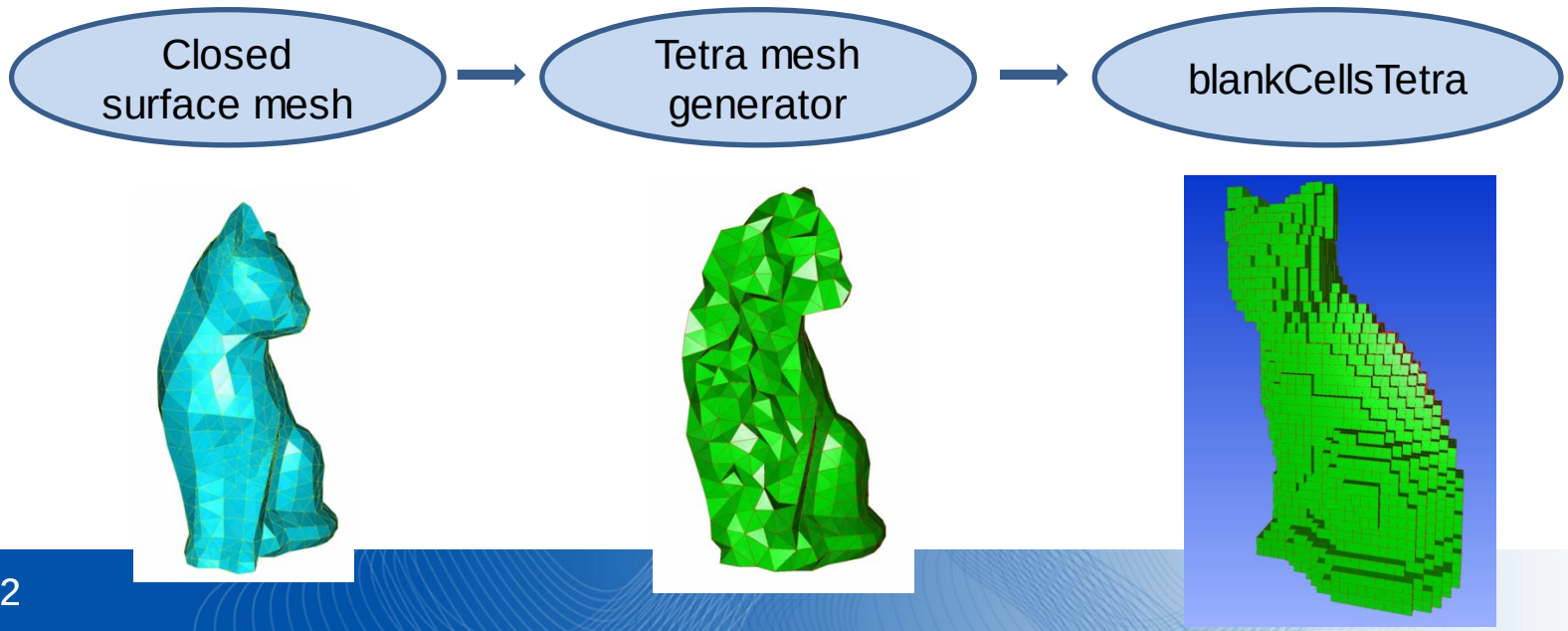
Hole cutting techniques (1): X.blankCells

- X.blankCells: original X-Ray hole-cutting technique (Meakin, 2001)
  - Requires closed and watertight 2D meshes to describe bodies
  - Rays are cast from the (X,Y) plane only in our implementation
  - Available at cell vertices & centers



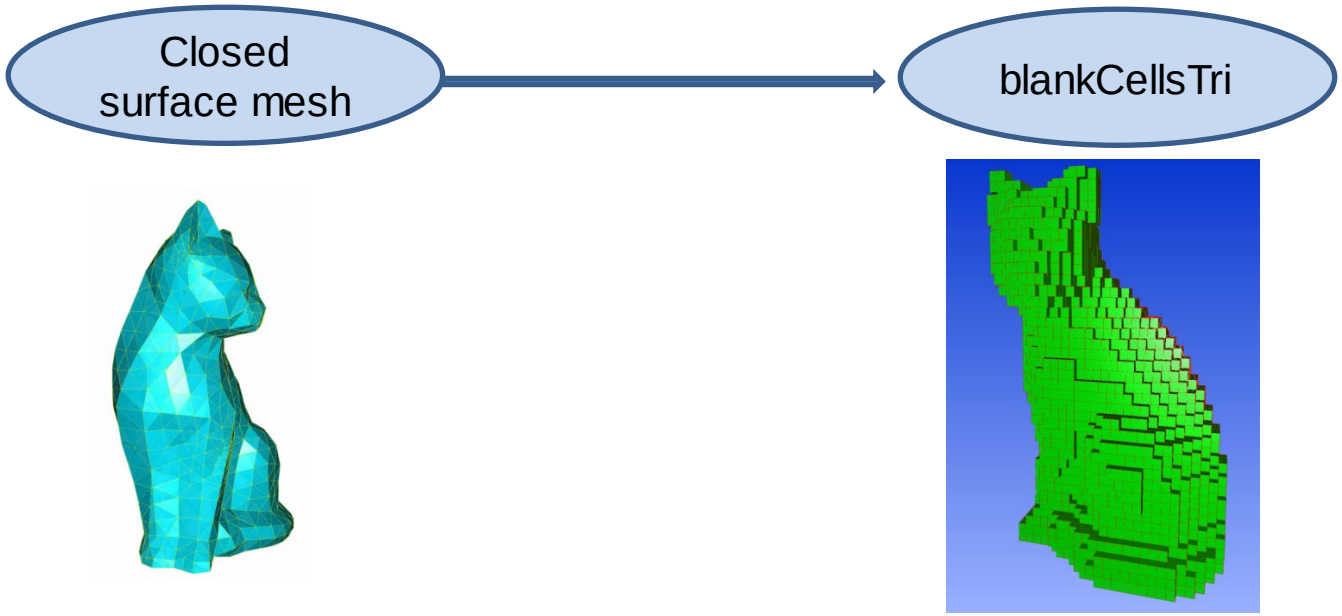
Hole cutting techniques (2): blankCellsTetra

- blankCellsTetra:
  - Body represented by a set of tetrahedra
  - Points are blanked if inside one tetra
  - Pros/Cons:
    - Accurate & fast (4 determinants computed per node)
    - But requires a TETRA mesh of the solid



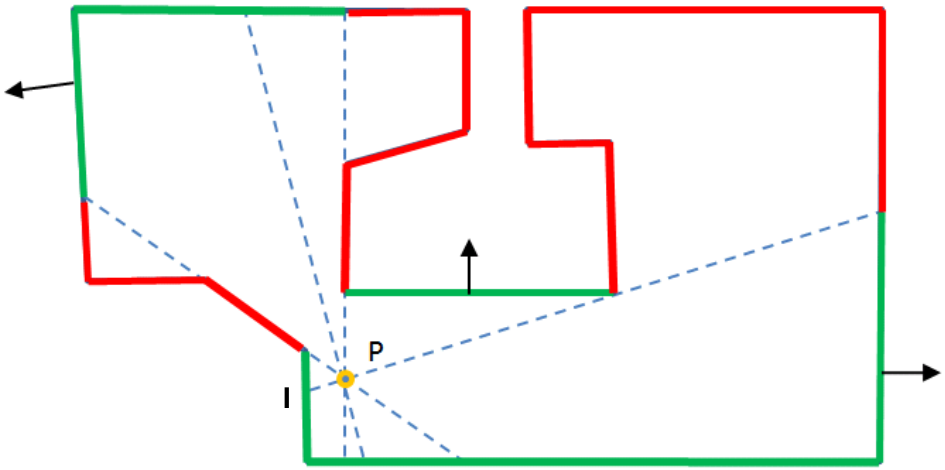
Hole cutting techniques (3): blankCellsTri

- blankCellsTri:
  - Body represented by a triangular mesh
  - Surface mesh must be closed (overlapping surfaces OK)



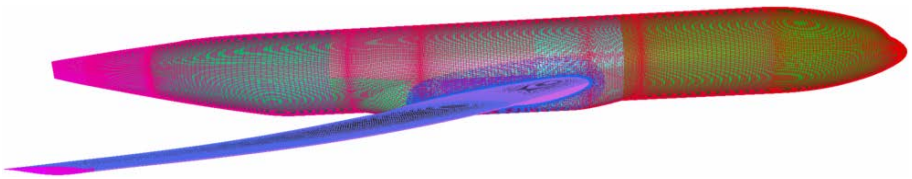
Algorithm

A node P can always be located in relation to a surface as one can always find a triangle T of the surface visible from that point P.

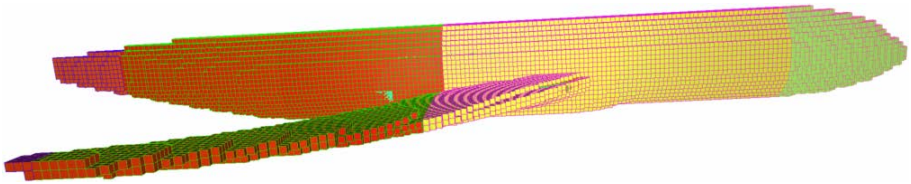


- Storage of the surface nodes in a k-d tree
- Find the closest node I to P in the k-d tree
- Find the closest intersecting triangle with the segment [PI]
- Sign of the normals vs sign of [PI]

Hole cutting techniques: comparisons



Fuselage/Wing/Collar configuration
137 structured zones, 27 MPts



2 algorithms compared:

- Cell blanked if cell center inside the solid
- Cell blanked if intersected by the solid

	center_in	cell_intersect
X-Ray	19 s	24 s
blankCellsTetra	12 s	17 s
blankCellsTri	22 s	25 s

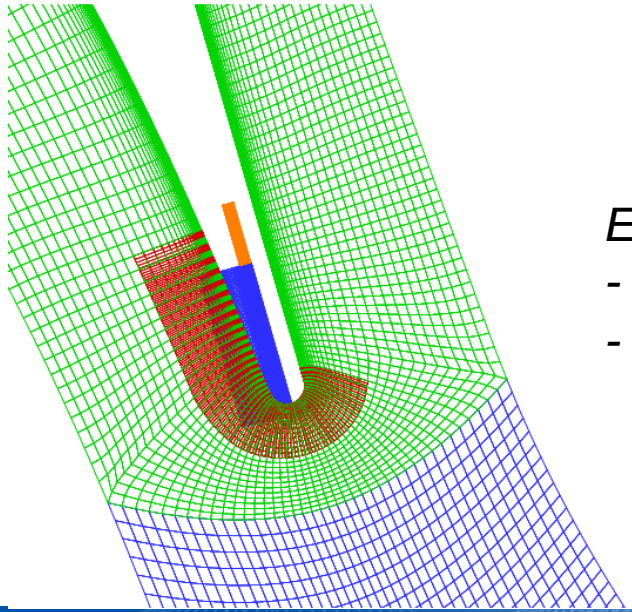
16 threads OpenMP on SGI Altix

blankCellsTetra/Tri accurately mark blanked cells/nodes

- ✓ Useful for high-lift devices
- ✓ Other purposes: blanking for Immersed Boundary Method

Composite definition of surfaces

- "Doubly defined" BCs
- Useful to add slots in a mesh arbitrarily:
  - Slot borders do not necessarily intersect the background mesh at a node ⟹ borders are defined by the physical boundary and overlap
  - If a point on a doubly defined BC is interpolable from a donor zone, meaning that it is a fluid point, then it is interpolated. Otherwise, the physical BC will be applied.



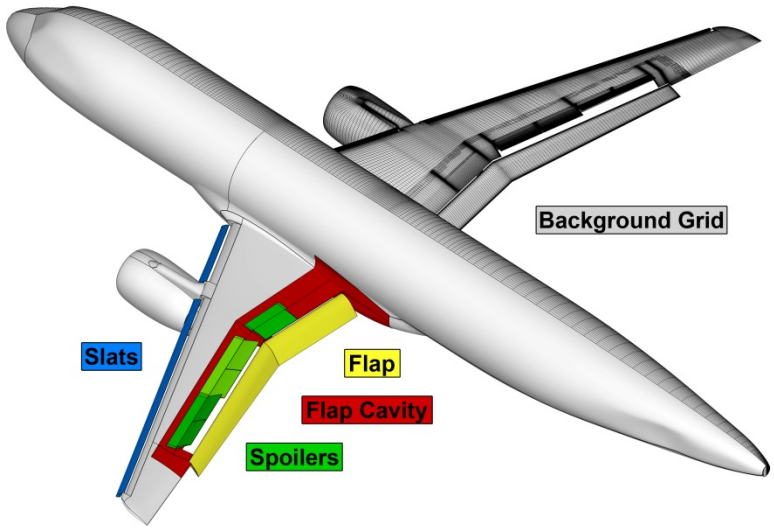
Example of doubly defined BCs:
- on the wall boundary of the blade (green)
- on the lateral borders of the slot (blue)

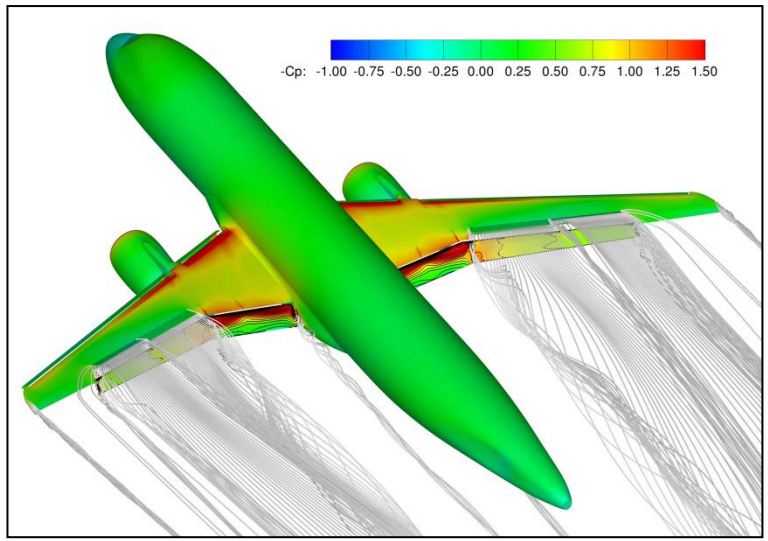
```
t=X.setDoublyDefinedBC(t)
```

Overlap optimization

- Based on the PEGASUS algorithm (Suhs, Rogers)
- Selection based on volume or on a priority to a Chimera component over another component
- Projection algorithm when grids overlap at a wall border
- Work in sequential (OpenMP) and distributed (OpenMP/Mpi)

```
t=X.optimizeOverlap(t)
```





Overset grid connectivity

- Different interpolation methods w.r.t. to donor mesh type:
  - Structured: 2nd order, 3rd and 5th order Lagrange, Moving Least Squares (MLS, 3rd order)
  - Tetra: 2nd order, MLS
  - Polyhedral: MLS
- Projection algorithm when grids overlap at a wall border
- Optimized donor cell search:
  - Oriented Bounding Boxes to reduce the number of candidate donor zones
  - ADT for donor cell search in a candidate donor zone
- Information stored in the pyTree for transfers

```
t=X.setInterpData(t)
```

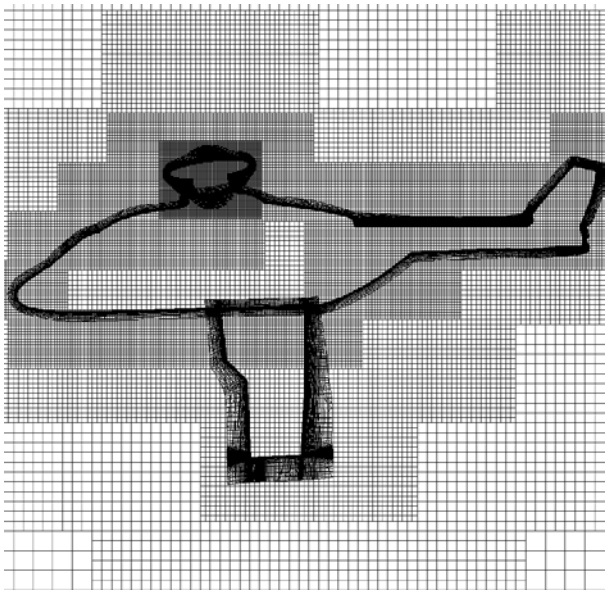
Transfers

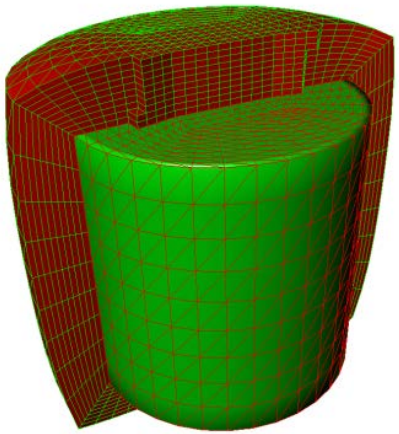
- Use of data stored in the pyTree (donor indices, coeffs, interpolation type)
- Works in OpenMP and OpenMP/Mpi
- Generalized: update of the solution for abutting connectivities and IBM points

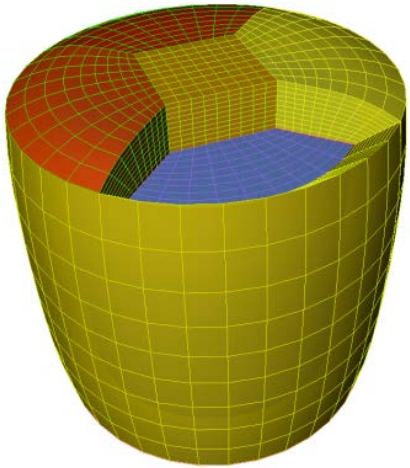
```
t=X.setInterpTransfers(t)
```

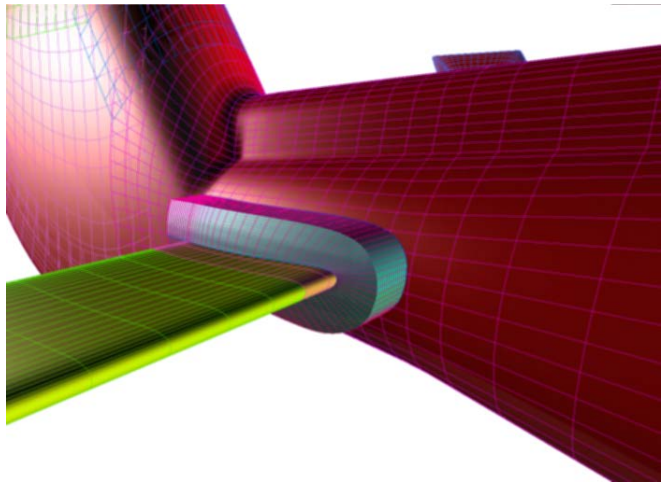
Cassiopee: some useful mesh generation functions

- 3D mesh generation functions
  - Extrusion
  - Off-body Cartesian grids
  - Collar grids
  - Others…



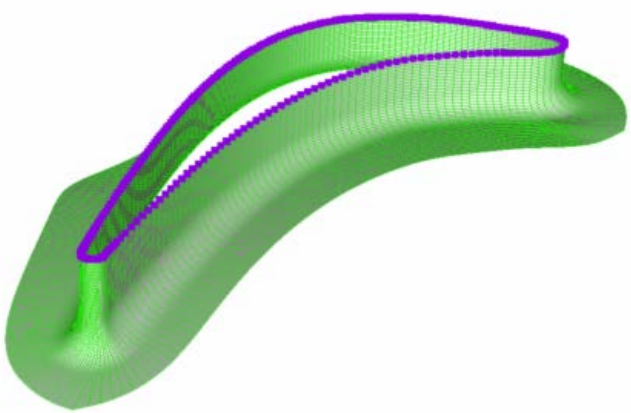


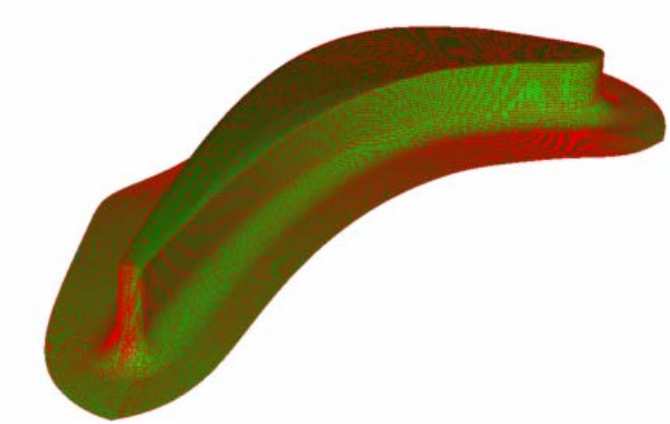




Cassiopee: surface mesh operations (1)

- Useful to create bodies for hole-cutting/post-processing:
  - Close holes in a triangular mesh

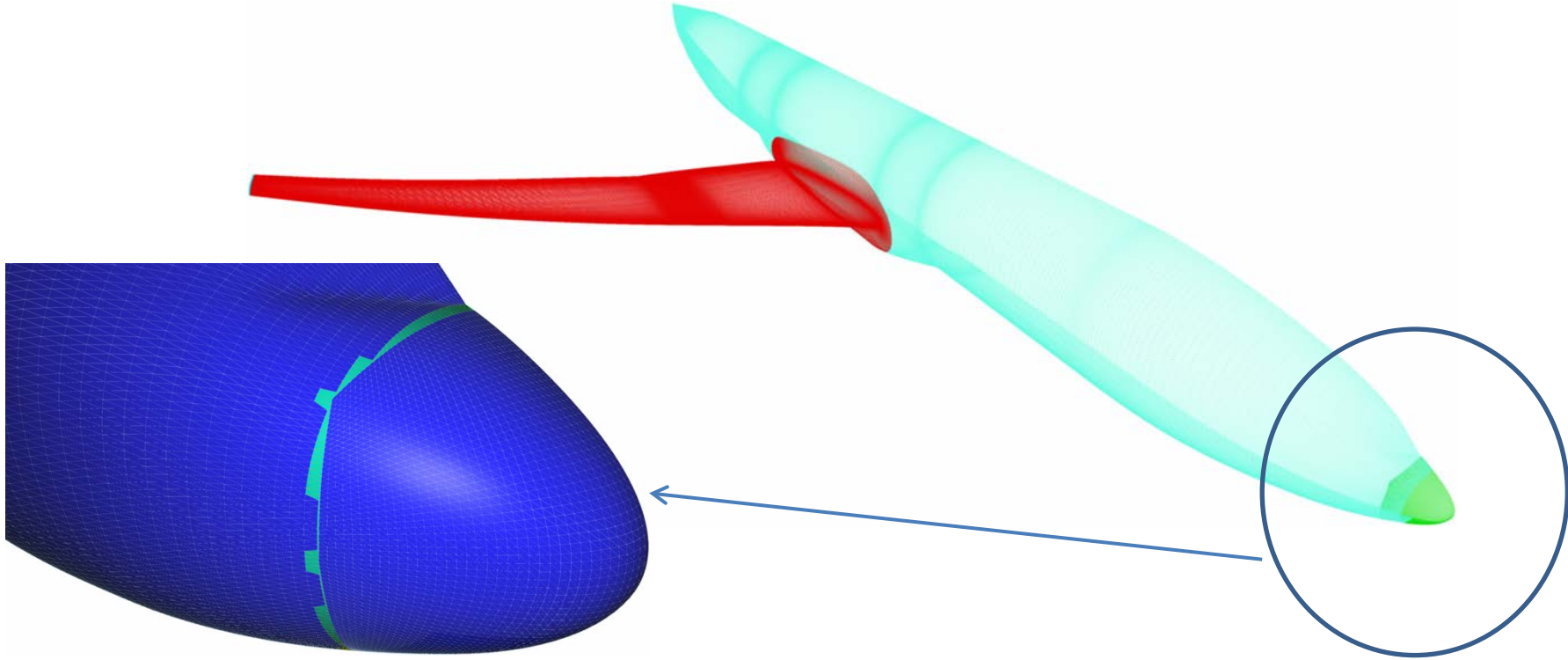




```
t=G.gapFixer(c)
```

Cassiopee: surface mesh operations (2)

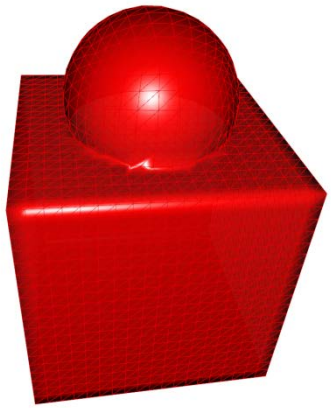
- Useful to create bodies for hole-cutting/post-processing:
  - Unique surface reconstruction from overlapping surfaces



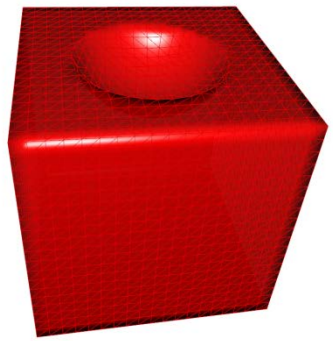
```
t=G.gapsmanager(t)
```

Cassiopee: surface mesh operations (3)

- Useful to create bodies for hole-cutting/post-processing:
  - Surface mesh starting from intersecting surface meshes:
    - Closed input surfaces: 2D boolean operators (union, difference,intersection)



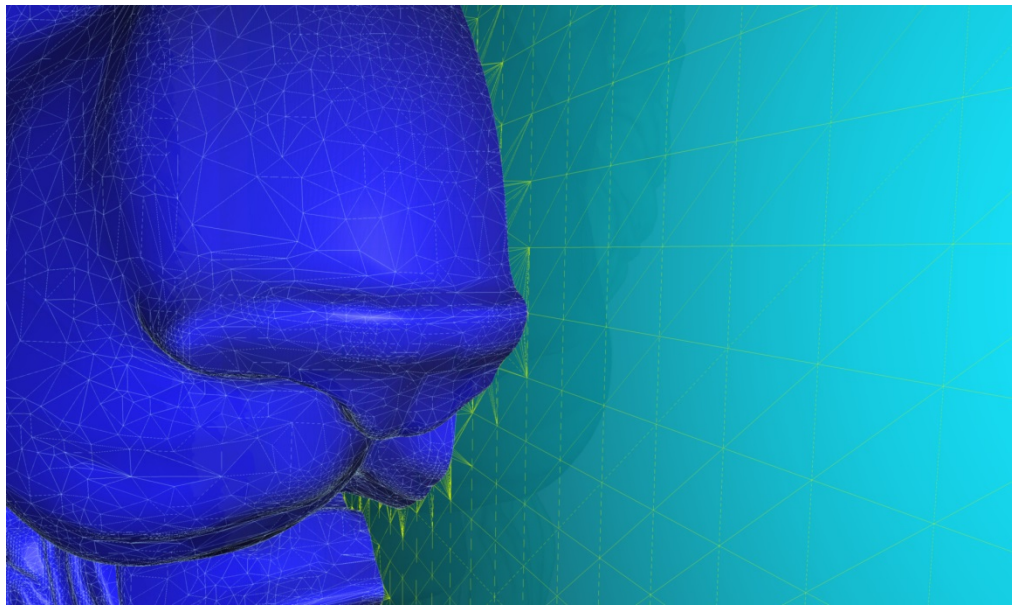
```
t=G.booleanUnion(a,b)
```



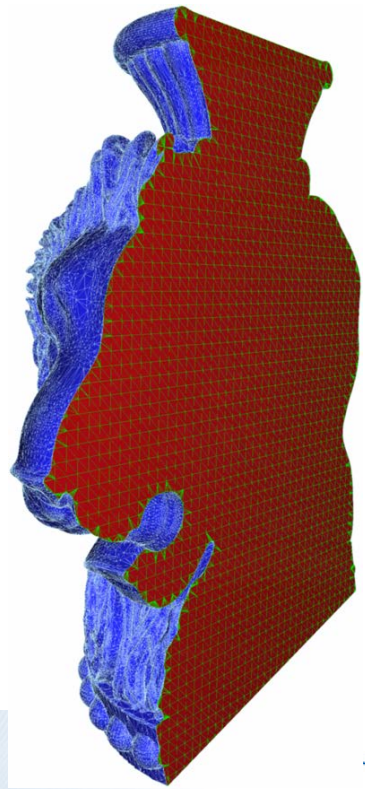
```
t=G.booleanMinus(a,b)
```

Cassiopee: surface mesh operations (4)

- Useful to create bodies for hole-cutting/post-processing:
  - Surface mesh starting from intersecting surface meshes:
    - Closed input surfaces: 2D boolean operators (union, difference,intersection)
    - **Open input surfaces: conformization + splitting of manifold portions**

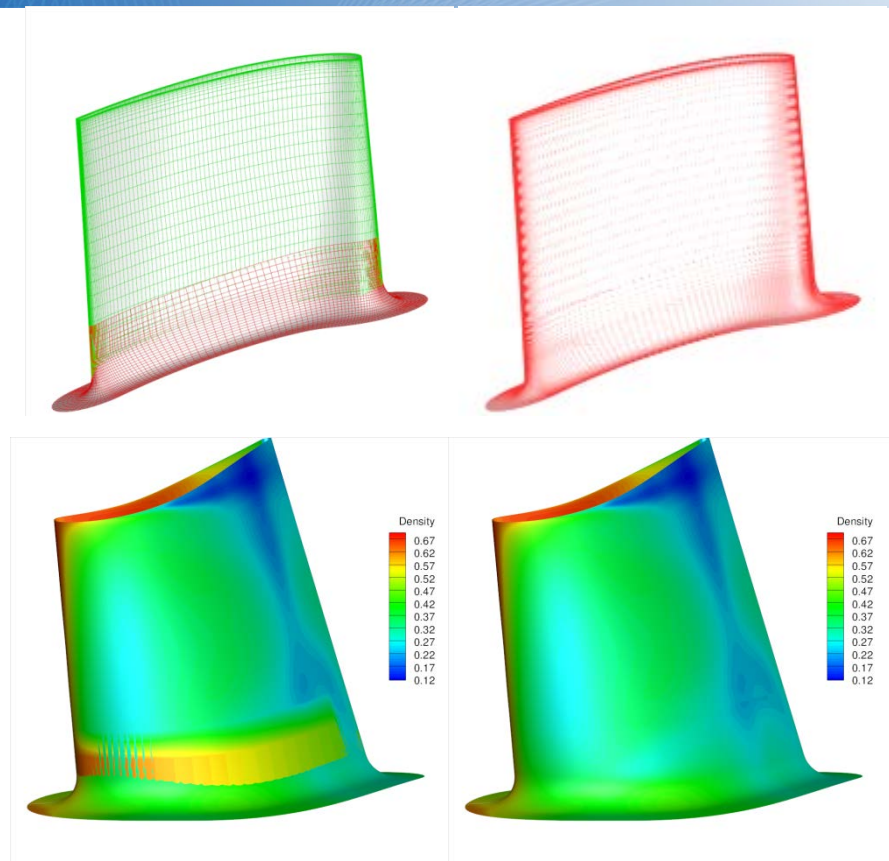


```
z = G.conformizeUnstr(a)
zones = T.splitManifold(z)
```



Cassiopee: post-processing & overset grids

- Chimera nature of points taken into account
  - Weighted integrations and loads
  - Slices, isosurfaces, isolines
  - Interpolation of the solution on geometrical features (planes, points, 3D mesh)



Overset *Reconstructed*

Unique surface reconstruction: `G.gapsmanager`
Then interpolation: `P.extractMesh`
Integration of pressure: `P.integ`

Applications

Overset grid assembly for aerodynamics applications

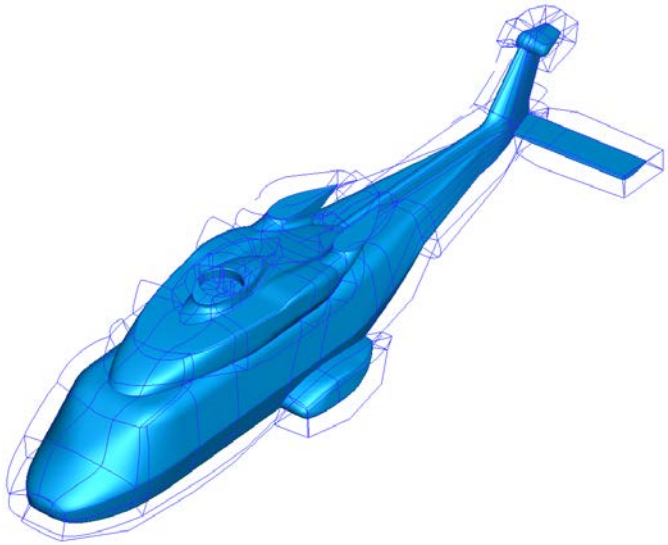
- Applications achieved with structured solver of *elsA*
- Bodies in relative motion
- Geometrical details addition
- Accurate capture of wakes
- Parametric studies & flow control

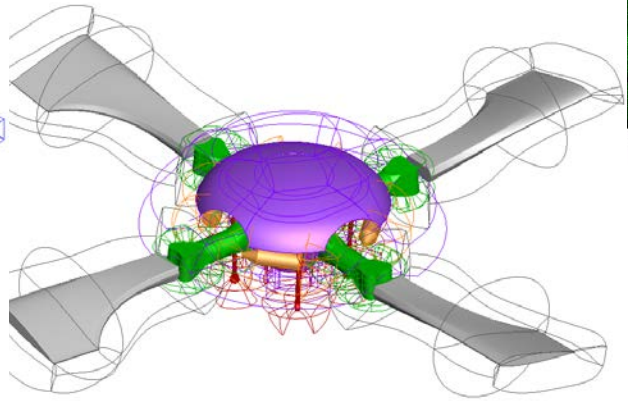
1. Bodies in relative motion

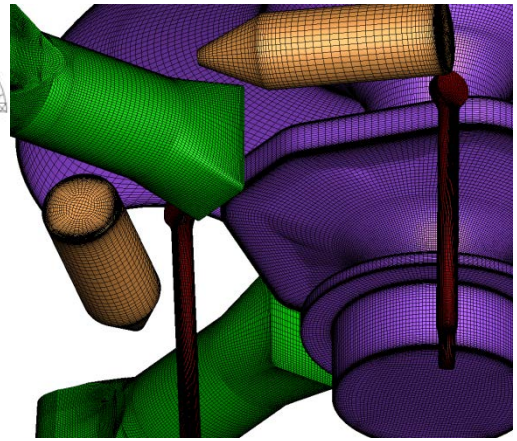
Ref: T. Renaud, A. Le Pape, S. Péron, « Numerical analysis of hub and fuselage drag breakdown of a helicopter configuration », CEAS Aeronautical Journal, 2014

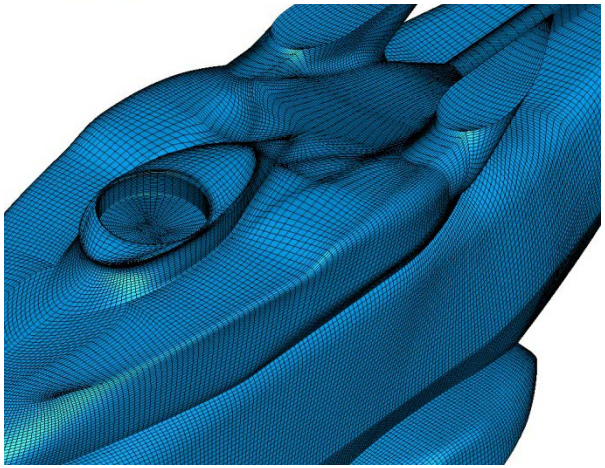
Helicopter fuselage & rotor head

- Evaluation of the influence of the rotor head on the drag









Fuselage: 8M pts

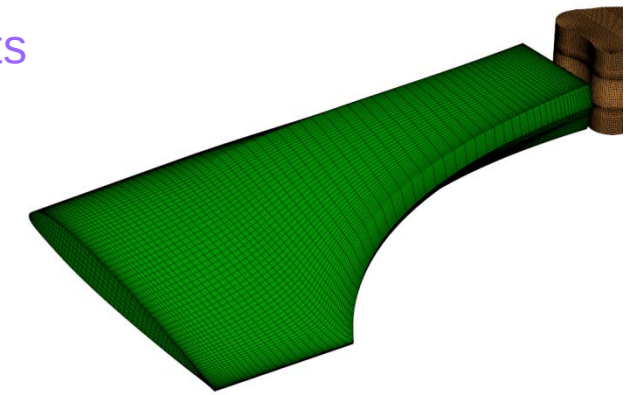
Rotor mast + cap: 12.6M pts

Attachs: 4x1.6M pts

Dampers: 4x0.25M pts

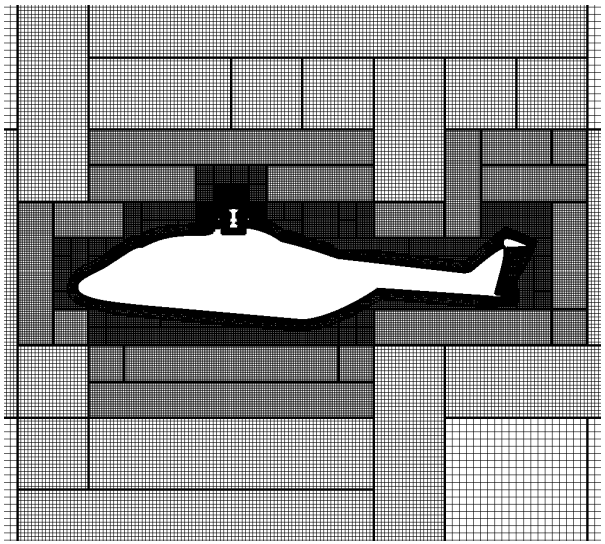
Rods: 4x1.1M pts

Stubs: 4x0.94M pts

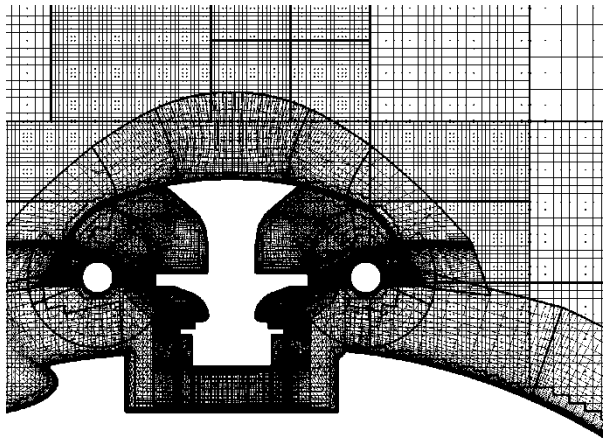


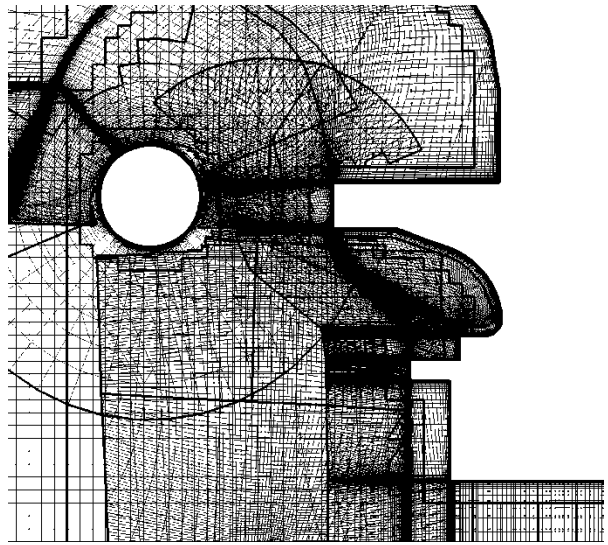
Helicopter fuselage & rotor head

- Near-body grids = 31.8M pts (pitch rods removed)
- Cartesian off-body mesh = 26.8M pts, refined in the vicinity of the rotor head



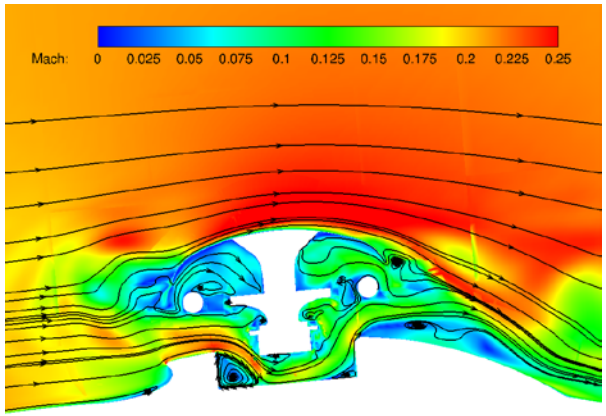
View of overset grids near the fuselage

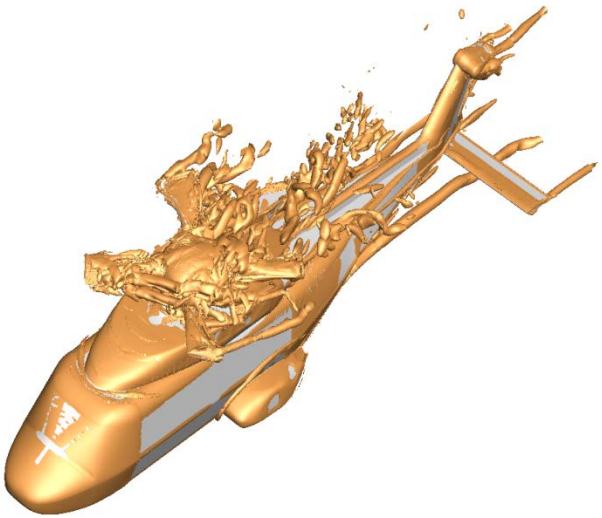




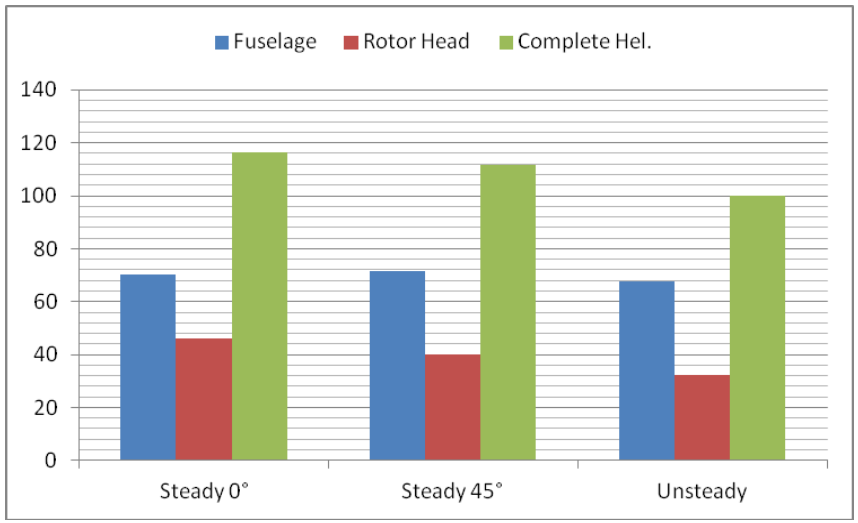
Detailed view of the hole-cutting near the rotor head

Unsteady simulation of the complete configuration





- X-Ray hole-cutting:
  - During pre-processing for Chimera components of same motion
  - At each time step if relative motions
- 2nd order interpolations achieved within elsA solver



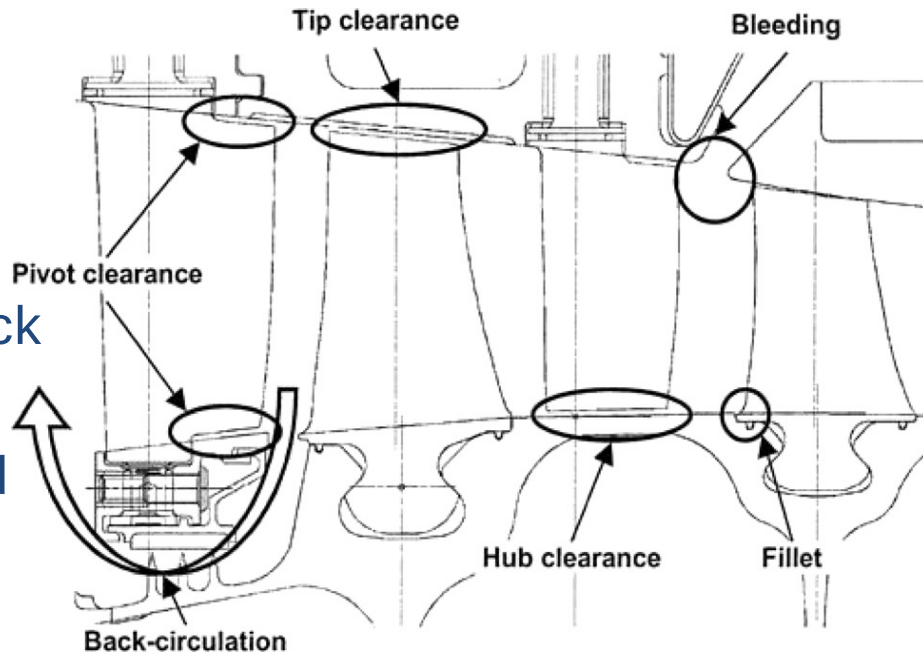
Rotor head drag: 35% of the total drag

2. Parametric studies

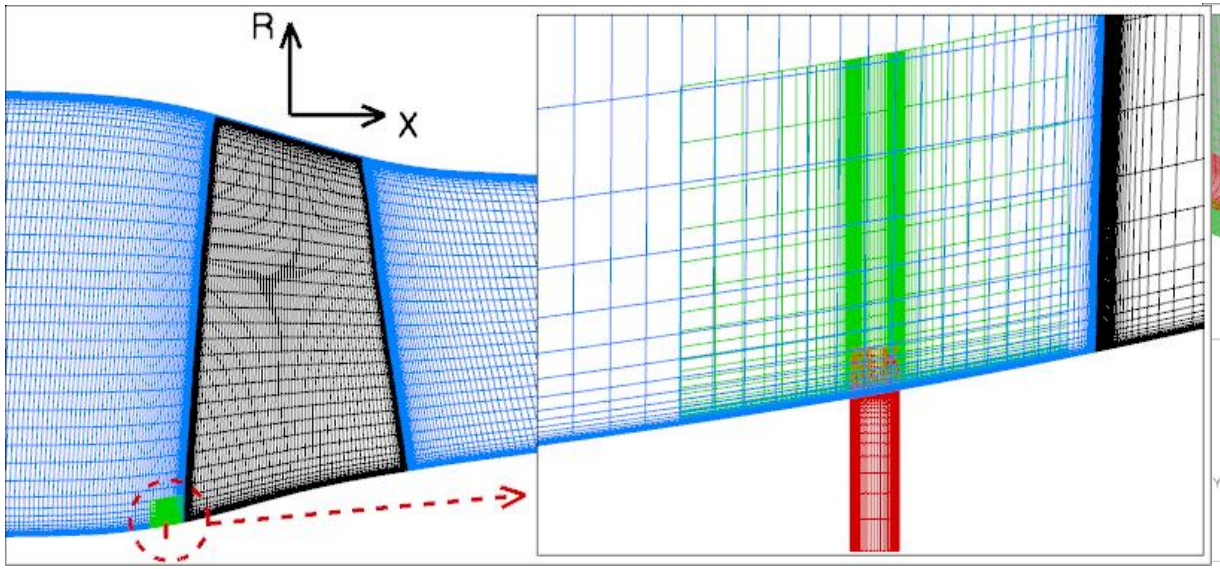
Refs:
L. Castillon, G. Billonnet, J. Riou, S. Péron, C. Benoit, « A Technological Effect Modeling on Complex Turbomachinery Applications With an Overset Grid Numerical Method », Journal of Turbomachinery, 2014,
L. Castillon, G. Legras, « Overset Grid Method for Simulation of Compressors with Nonaxisymmetric Casing Treatment », Journal of Propulsion & Power, 2013.

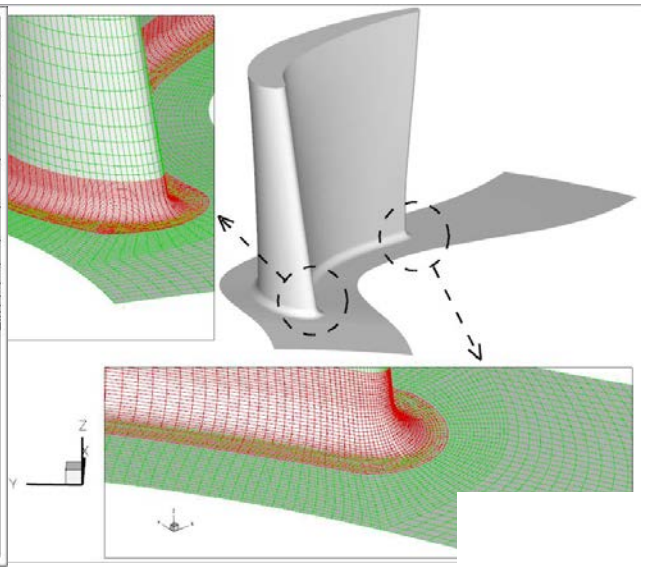
Turbomachinery applications

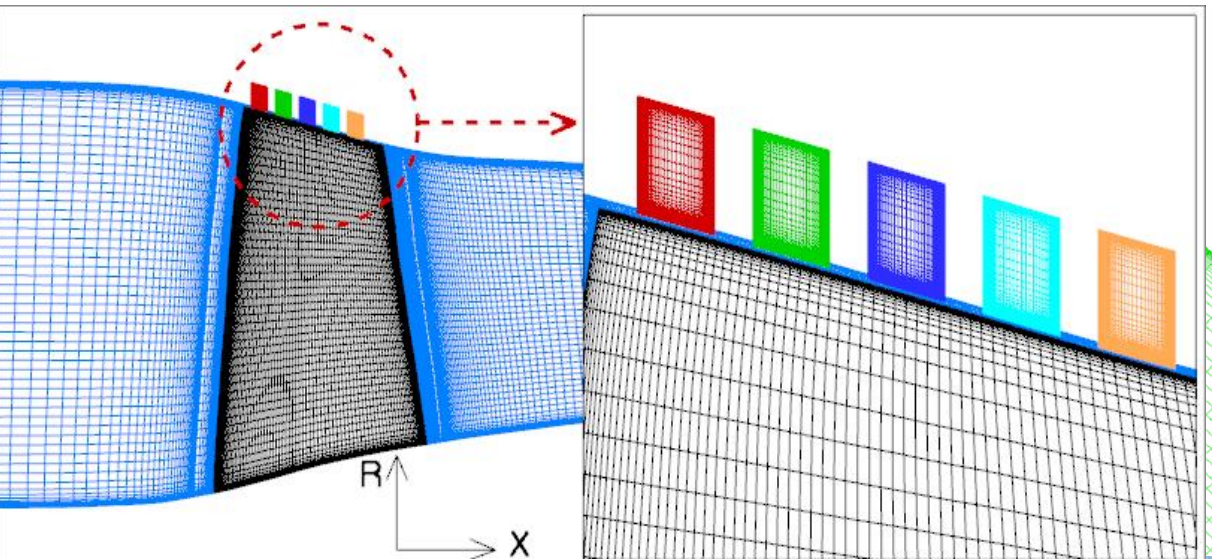
- Overset grids used to deal with technological effects
  - Simplifies strongly the mesh generation
  - Improved node distribution compared to structured multiblock approaches
  - Parametric studies are simplified (local grid modifications)
- But: acceptable conservation losses must be ensured

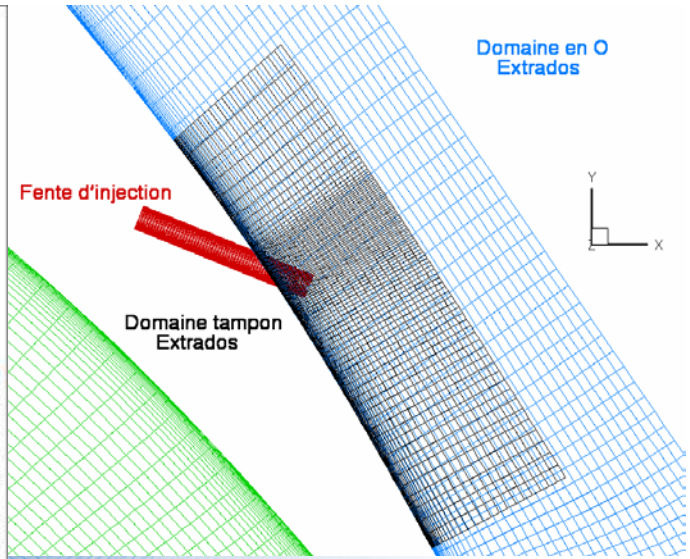


Examples


R
X

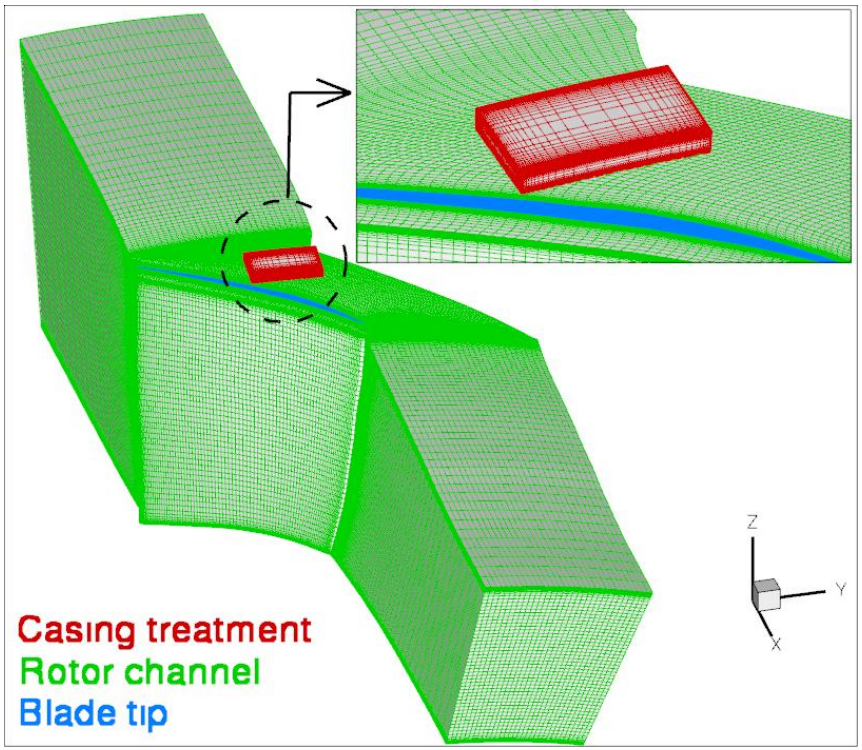


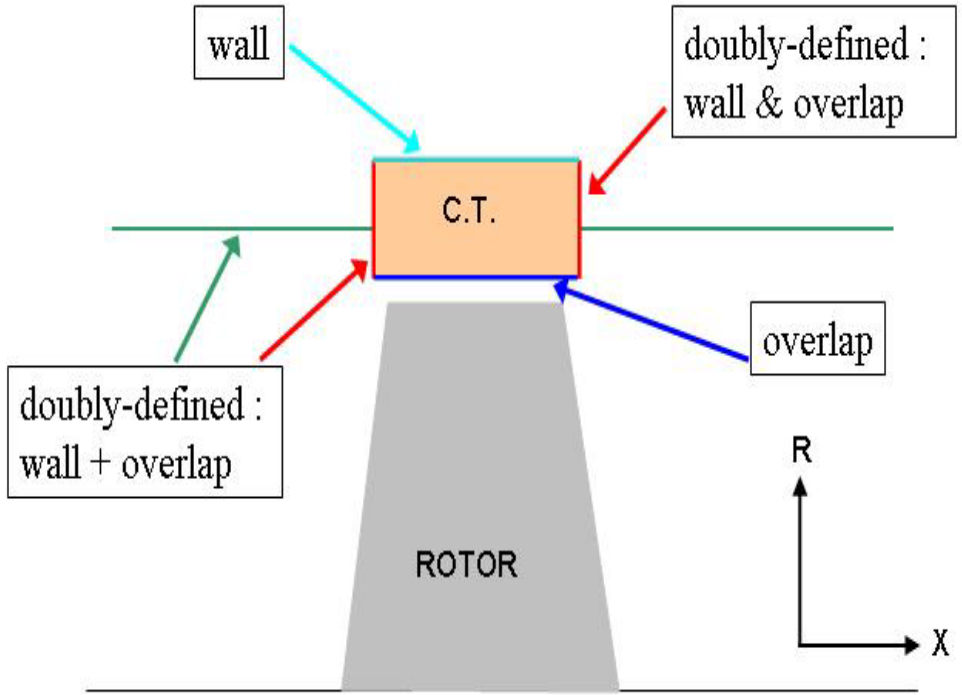

R
X


Domaine en O
Extrados
Fente d'injection
Domaine tampon
Extrados

Non-circumferential casing treatments

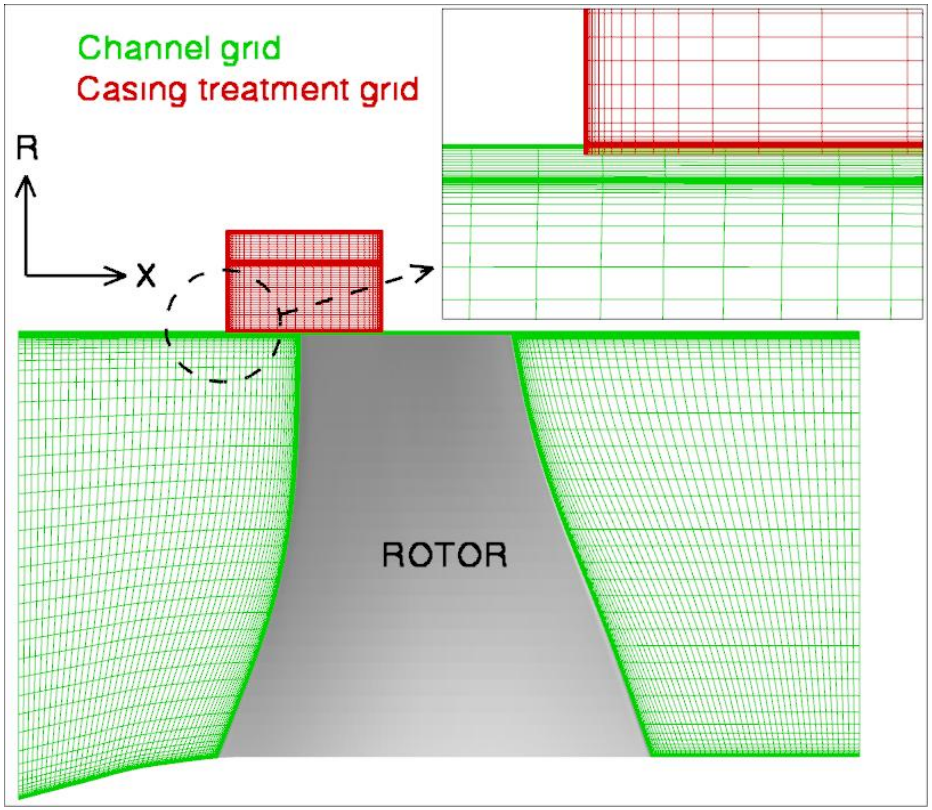
- Casing treatment defined by overset grids
- Unsteadiness of the flow: use of chorochronic method (periodic flow assumption) to reduce the computational domain to one channel (1 rotor / 1 casing treatment)

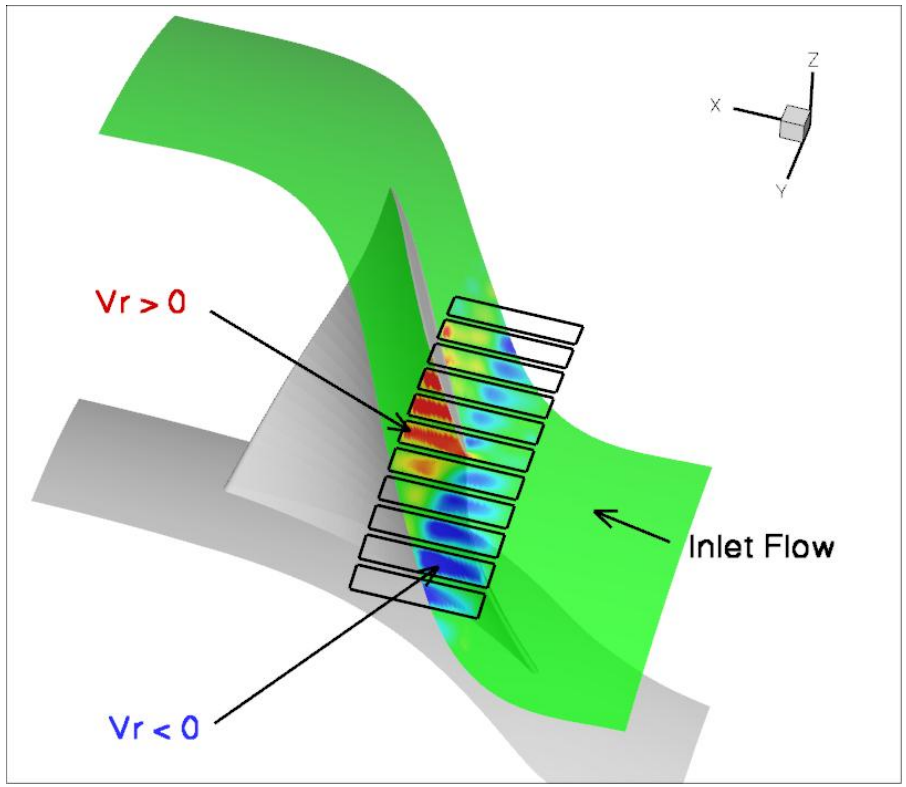




Influence of the effect of the CT on the stall

- BUAA configuration: 17 blades, 9 slots per blade passage
- 1 rotor blade channel: 13 blocks, 3.10^6 points
- y+ ~ 1 , Wilcox k-ω turbulence model



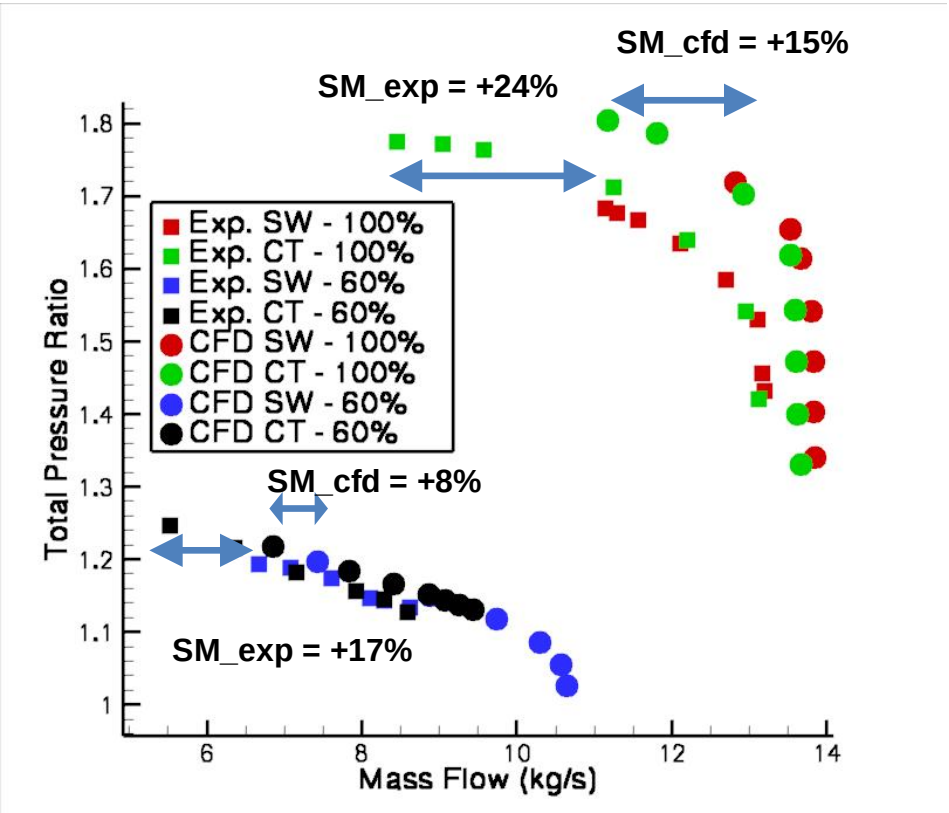


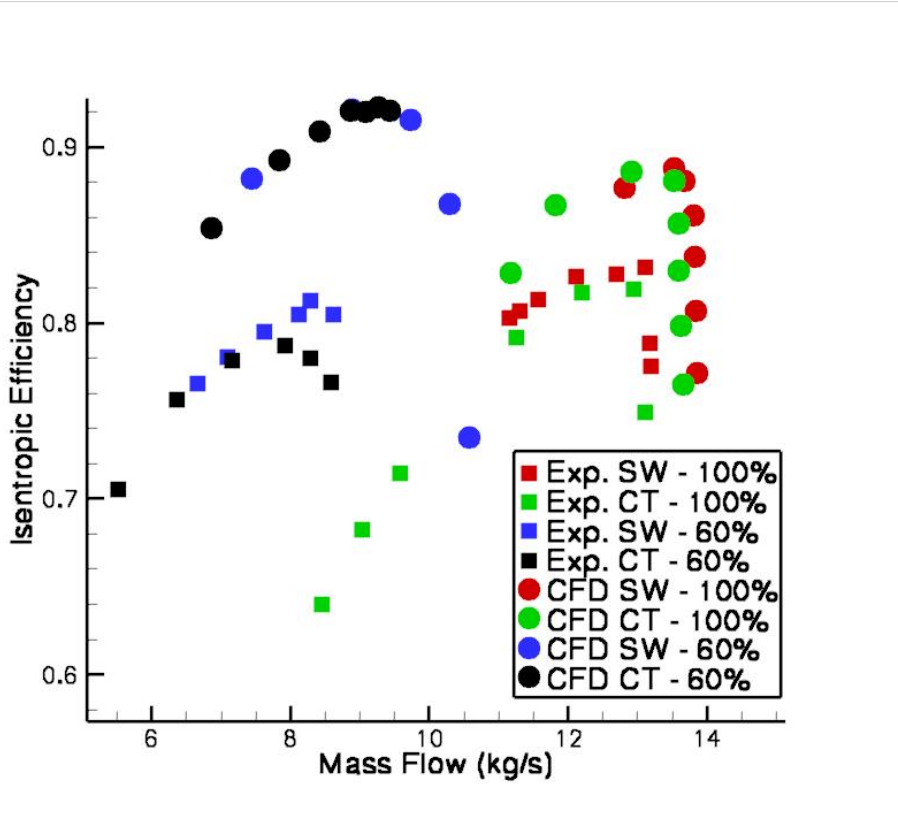
Casing treatment: compressor map results

- 2 rotation speed investigated : 100% and 60%
- Increase of stall margin with casing treatments
- Stall margin increase underestimated in CFD

$SM=(m_{solid\ wall}-m_{casing\ treatment})\ /m_{solid\ wall}$

Stall Margin	60%	100%
CFD – 100%	+8%	+15%
EXP – 100%	+17%	+24%



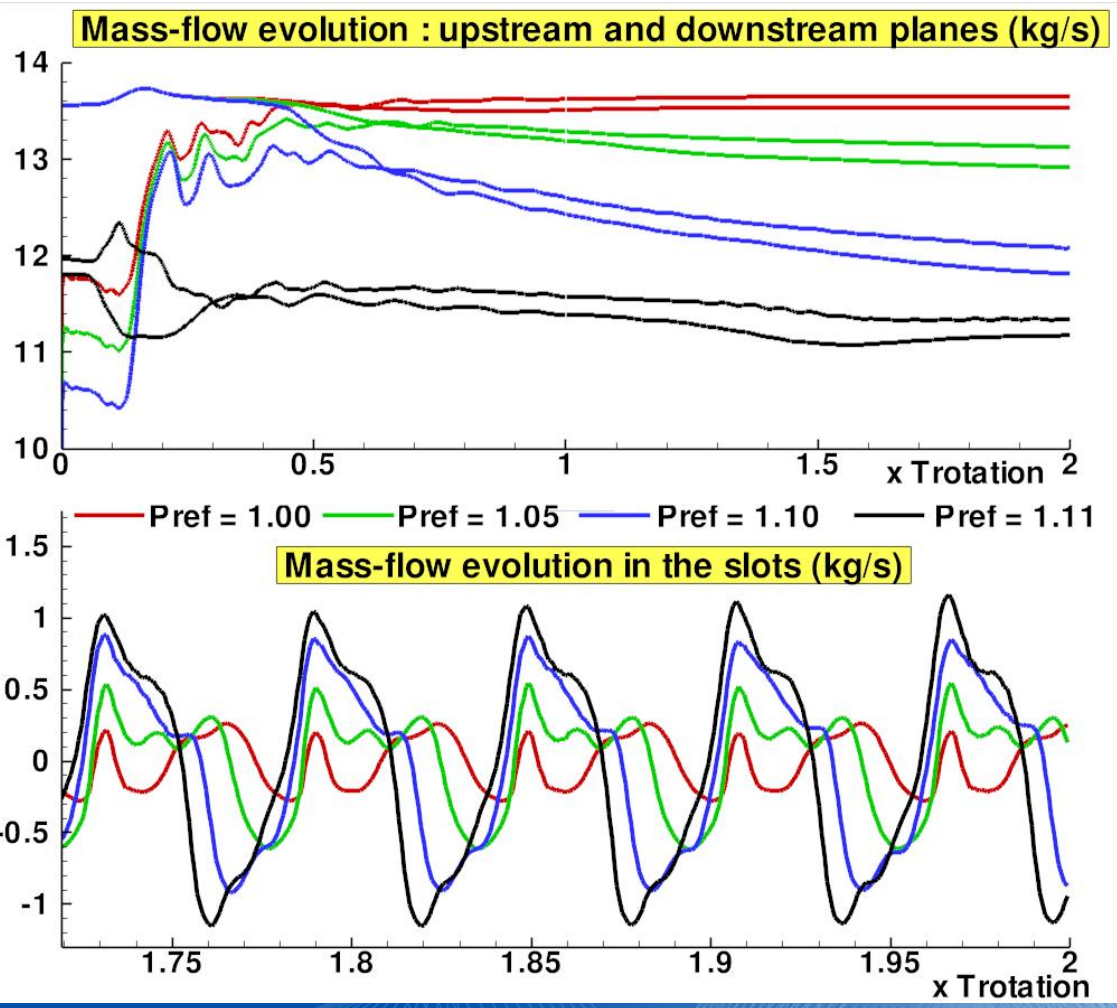


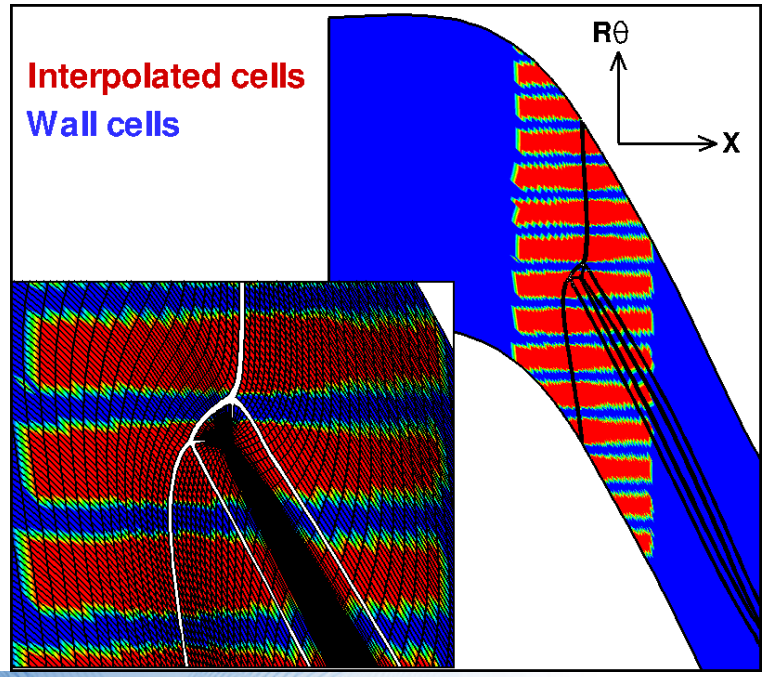
Mass flow evolution and conservation losses

Conservation losses of 2%

$\Delta m=(m_{upstream}-m_{downstream})\,/m_{upstream}$

p_{ref}	Δm
1.00	$0.9\ 10^{-2}$
1.05	$1.6\ 10^{-2}$
1.10	$2.2\ 10^{-2}$
1.11	$1.5\ 10^{-2}$

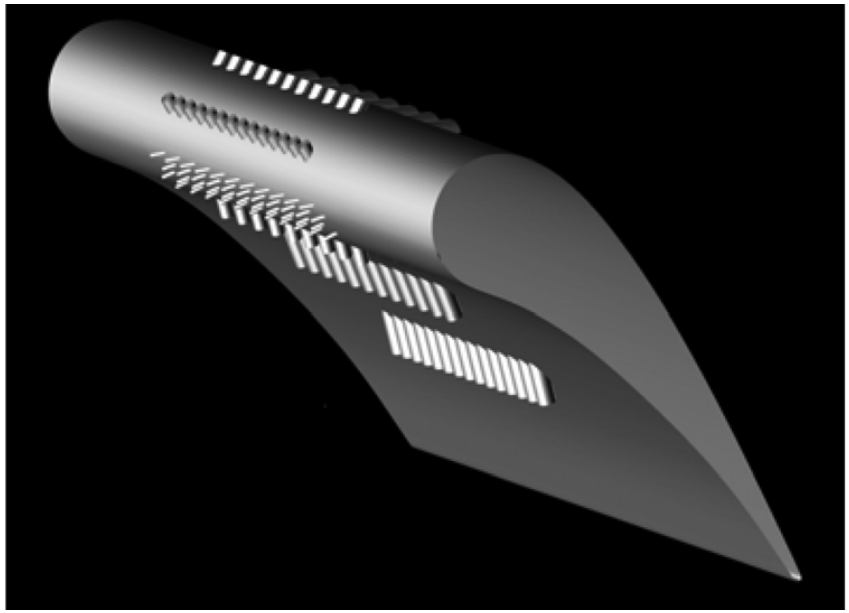




ONERA
THE FRENCH AEROSPACE LAB

Film cooling

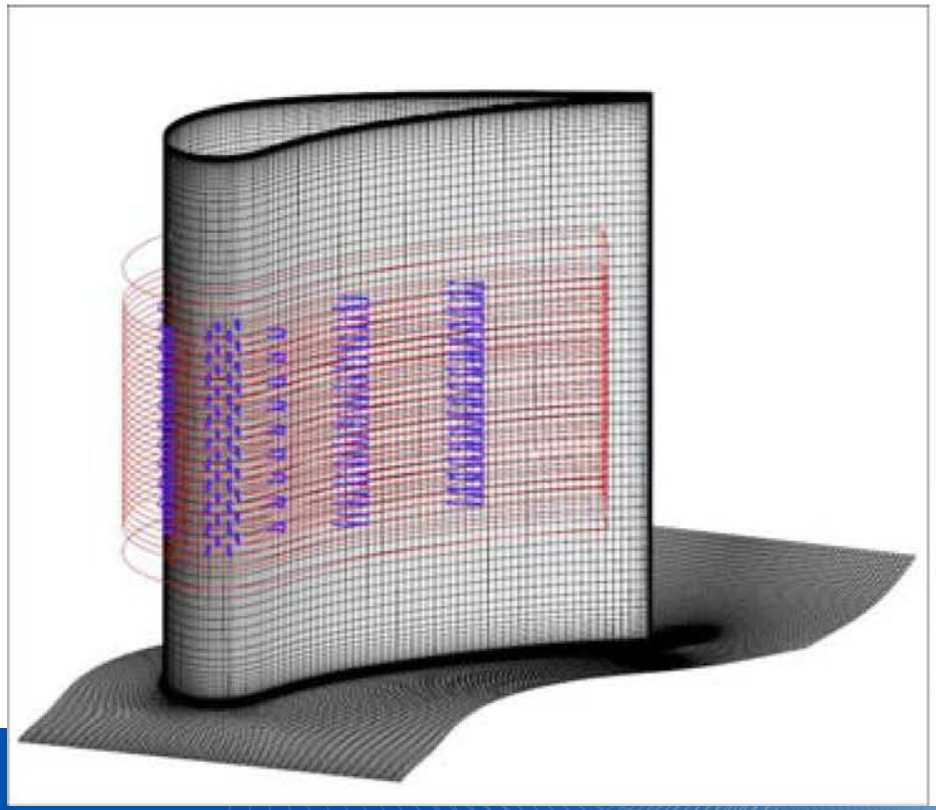
- Film-cooled nozzle guide vane (Jonsson & Ott, EPFL)
  - 113 holes on 10 rows
  - Very small holes ~ size of blade mesh cells

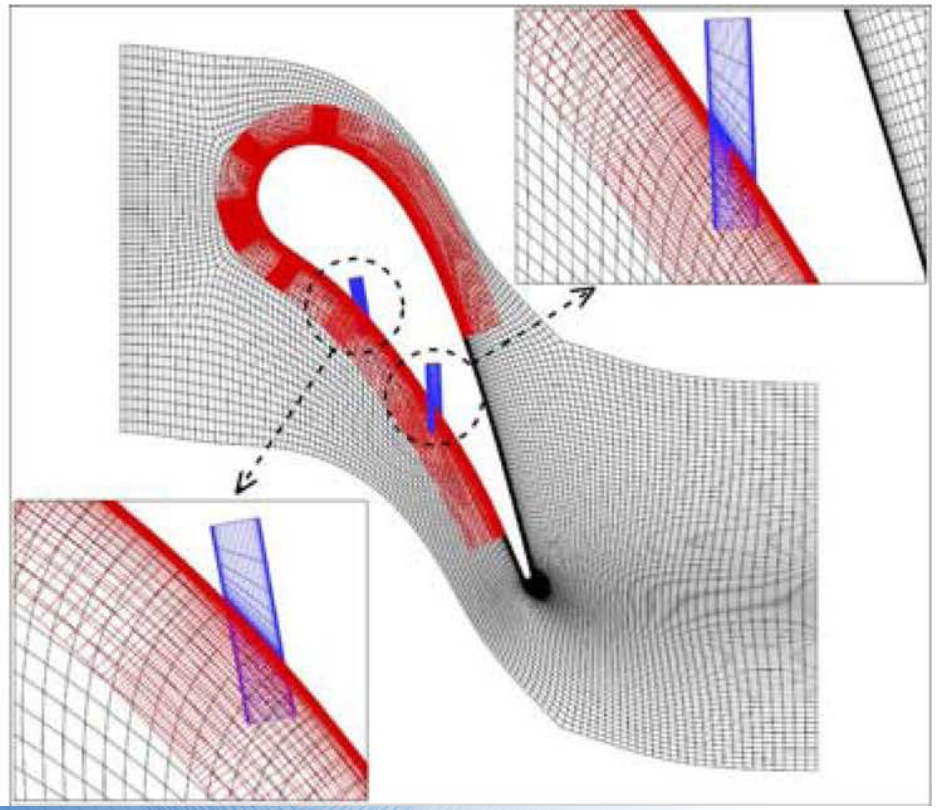


Meshes

- Overset approach
  - Double discretization of the wall
  - Doubly defined BCs
  - Overlap optimization

	# of points
Channel grid	1,6 M
Cooling holes	885,920
Buffer grid 1	7 M
Buffer grid 2	29 M





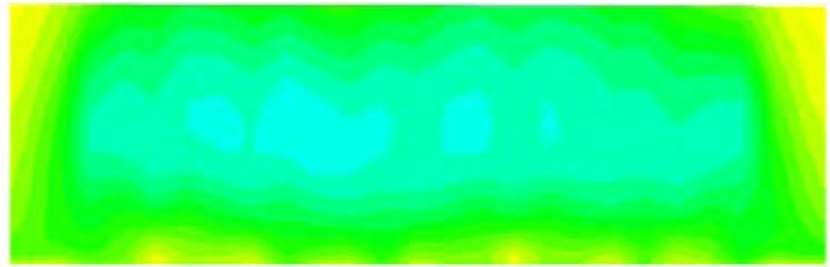
Conservation losses

- Dense buffer grids enable to capture the transport of the cooler flow and the interaction effects between the cooling flow and the main flow

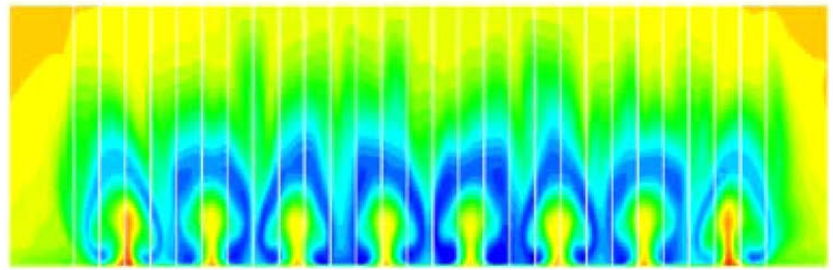
	Mass flow conservation
No buffer grid (a)	0.01865
Buffer grid 1 (b)	0.00241
Buffer grid 2 (c)	0.00111

(c) : vortical structures highlighted

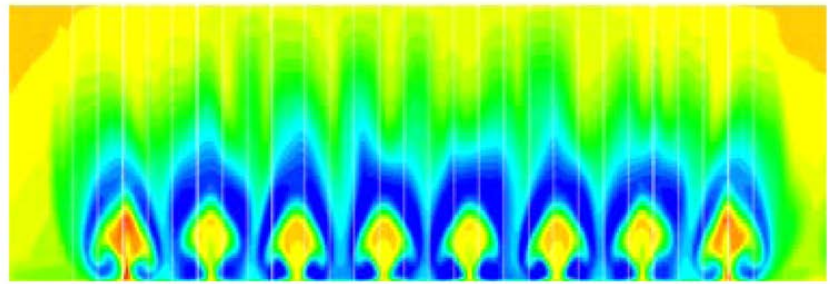
(*a*)



(*b*)



(*c*)



Plane normal to the wall on suction side.
Static temperature distribution

3. Aerodynamic shape design

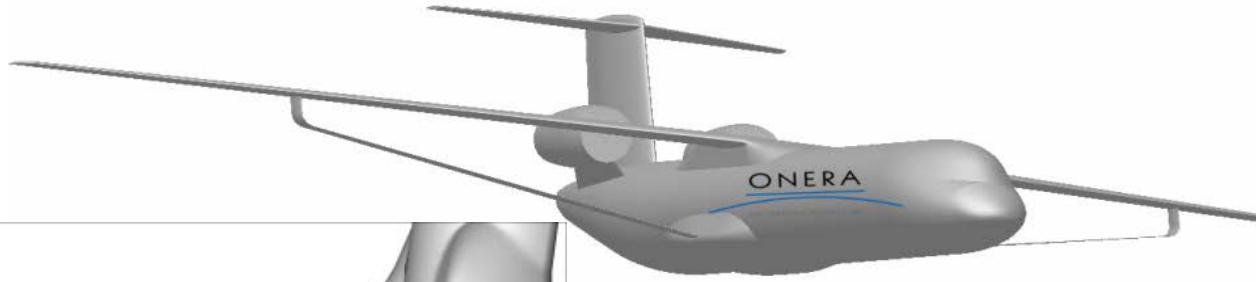
Ref: G. Carrier, O. Atinault, S. Dequand, J.-L. Hantrais-Gervois, C. Liauzun, B. Paluch, A.-M. Rodde, C. Toussaint, « Investigation of a strut-braced wing configuration for future commercial transport », ICAS, 2012, paper 597

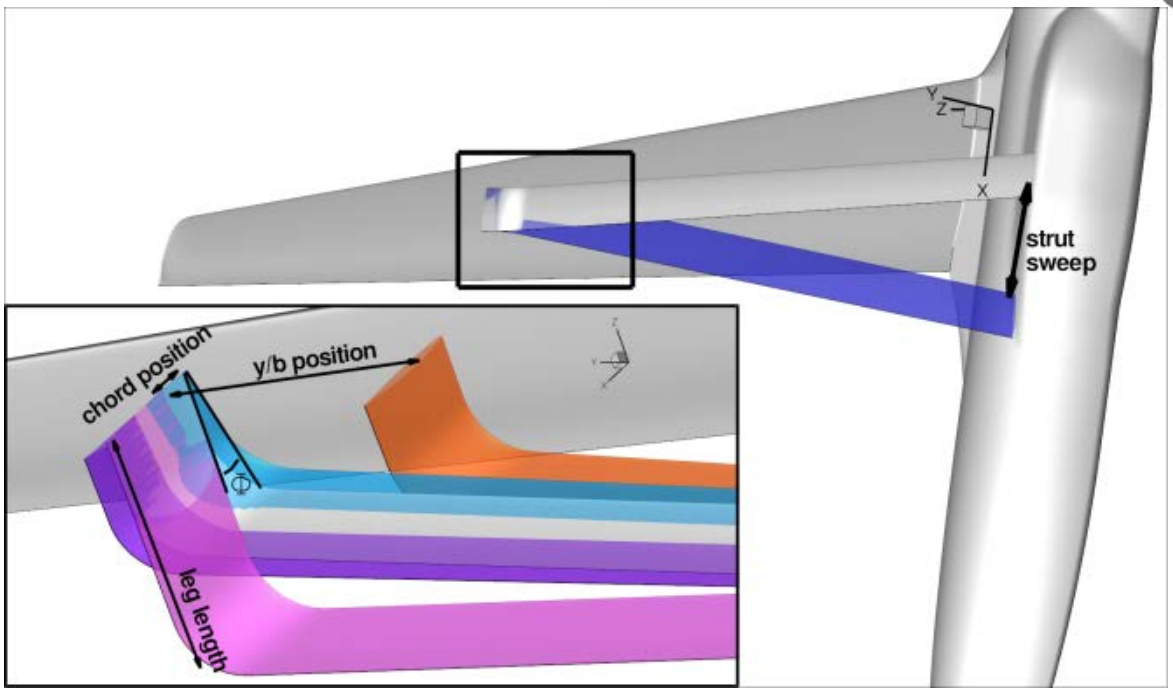
Aerodynamic shape design

- Optimisation with structured meshes
  - High quality CFD
  - Meshing / remeshing: complex task
  - Usually restricted to detailed design with mesh deformation
- Objective = Direct Operating Cost minimisation
  - Best aero-structure compromise
- Tools to derive the DOC
  - Overall Aircraft Design through analytical formulas: engine, low speed performances, mission…
  - High fidelity RANS CFD: transonic cruise performance
  - Low fidelity structure design (beam model)

Example: MDO strut-braced wing optimisation

- Wing & strut optimisation
  - Strut position (wrt the wing and the fuselage)
  - Wing span
  - Wing thickness

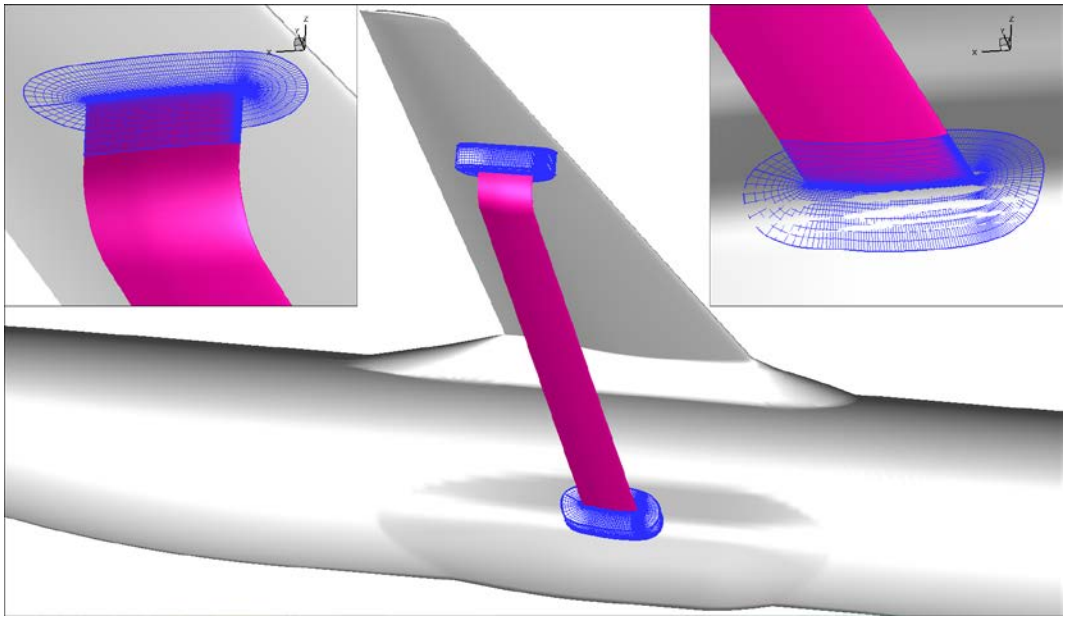




ALBATROS configuration

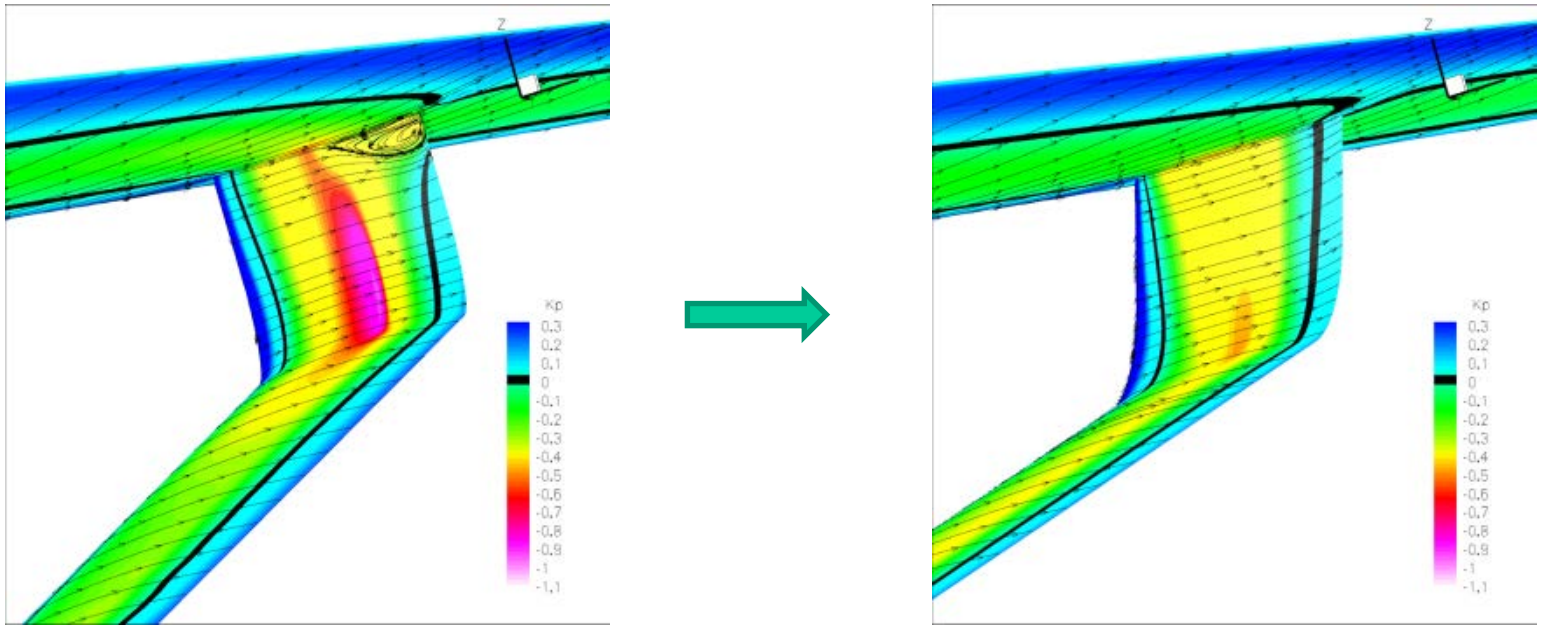
Automatic workflow for optimization

- Python script relying on Cassiopee functions:
  - Use of automatic collar grid generation:
    - ✓ No CAD handling
    - ✓ Boolean union operation detects the geometrical intersection
  - Overset grid assembly using Cassiopee/Connector functions:
    - ✓ Fuselage / Strut / Strut-wing collar / Strut-fuselage collar



ALBATROS configuration: results

- Strut geometry design thanks to high-fidelity CFD
- Analysis of the compromise between aerodynamics & structure
  - Strut attached at 60% of the wing span
  - Wing with moderate aspect ratio
  - Thin wing



4. Flow control

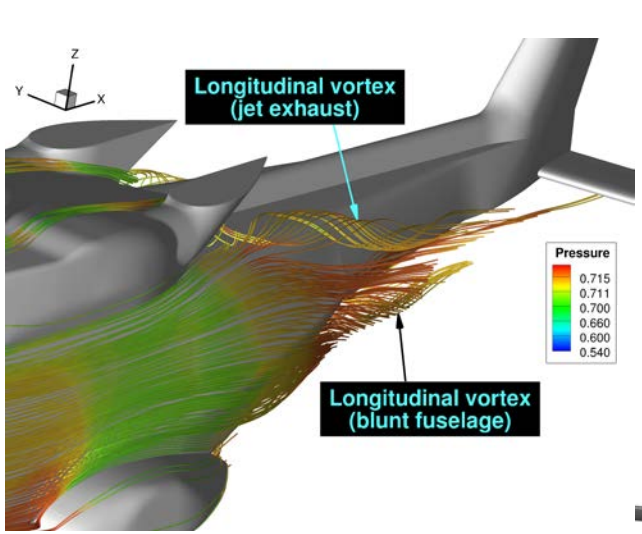
Refs:

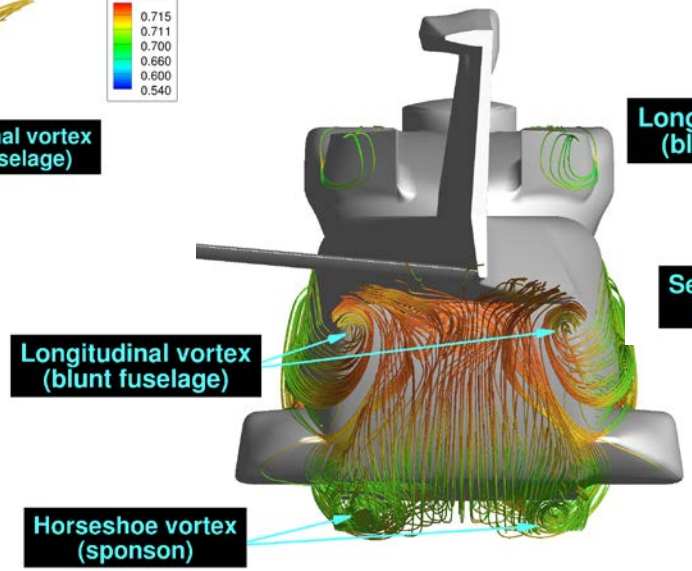
[1] G. Gibertini, J.-.C. Boniface, A. Zanotti, G. Droandi, F. Auteri, R. Gaveriaux, A. Le Pape, « Helicopter Drag Reduction by Vortex Generators», Aerospace Science & Technology, vol. 47, pp.324-339, 2015

[2] Alex Zanotti, Giovanni Droandi, Giuseppe Gibertini, Franco Auteri, Jean-Christophe Boniface, Robert Gavériaux, Arnaud Le Pape, « Wind Tunnel Assessment of the Computational Framework for Helicopter Fuselage Drag Reduction Using Vortex Generators», 72th American Helicopter Society Forum, 2016

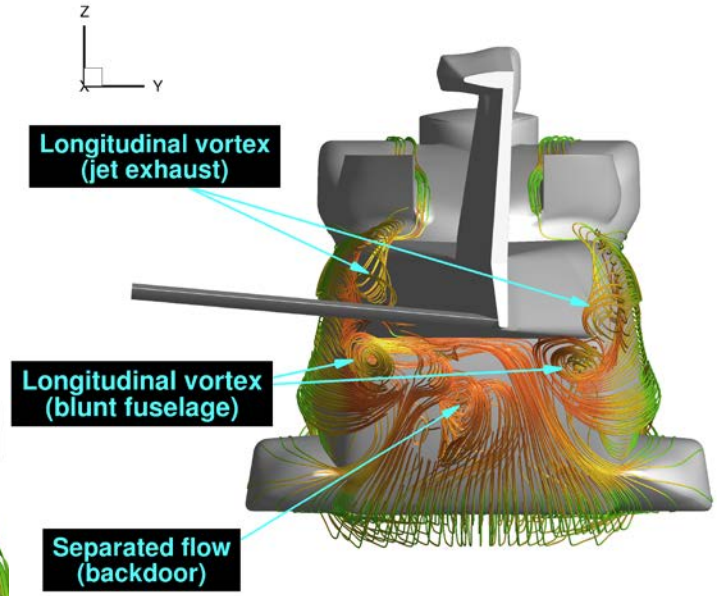
Target: reduction of the backdoor separation with VGs

Parametric studies of VG configurations operating in a complex 3D environment, which may dramatically impact the effectiveness of VGs





Near wake at cruise speed for AOA = -12°

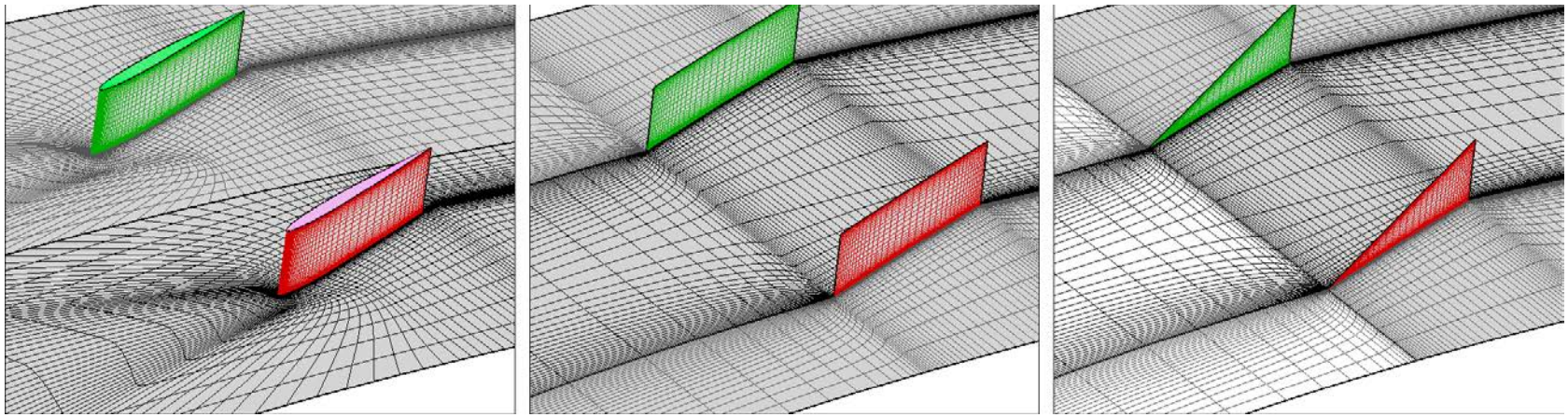


Near wake at cruise speed for AOA = -2°

VGs: overset approach for pre-test CFD simulations

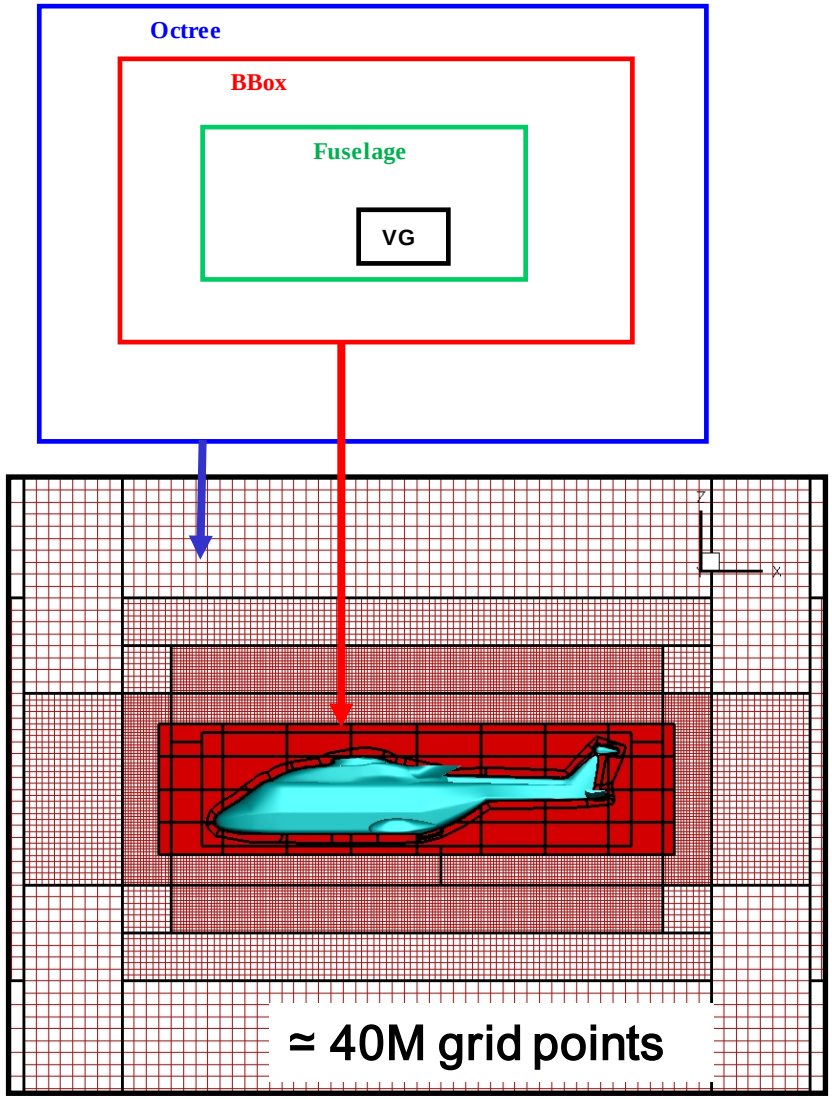
VG explicitly discretized in the CFD mesh as independent overset structured grids, allowing for parametric investigations:

- Location, size with typically VG height ≈ δ (local boundary layer thickness)
- VG pitch angle with respect to the local velocity field
- Configuration: pairs/arrays of co-rotating or counter-rotating VG
- Planform/thickness (to improve a primary design)

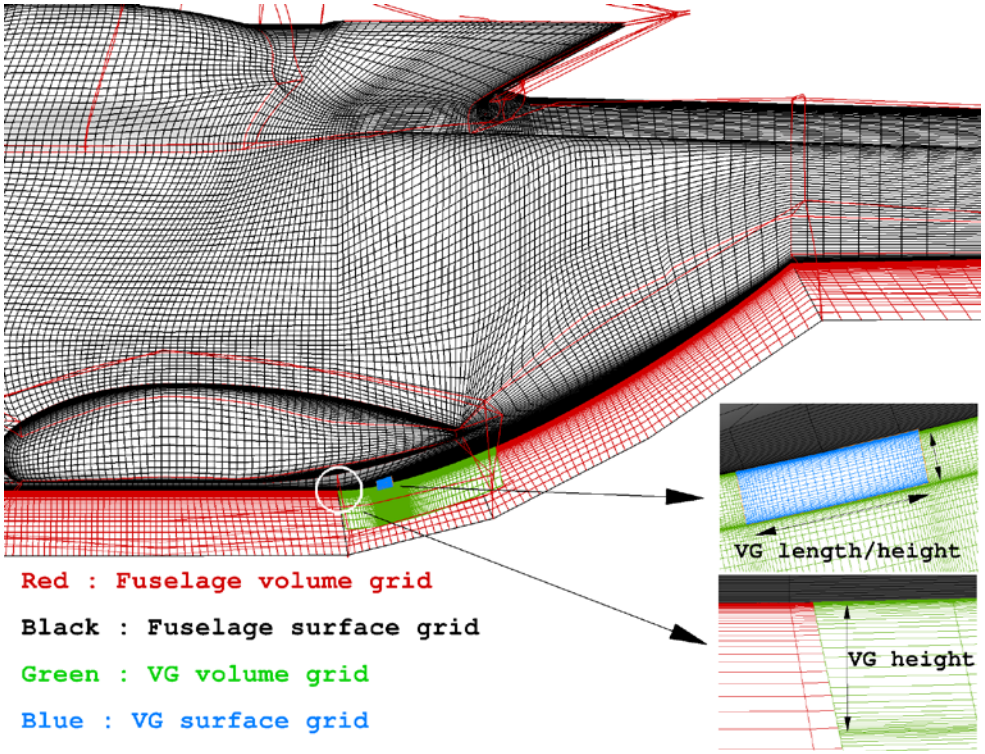


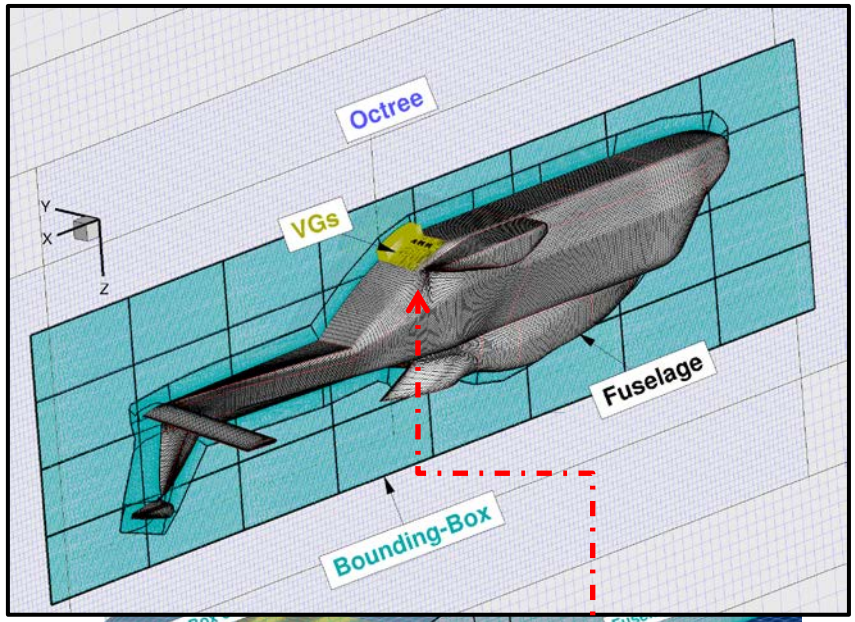
Computational methodology

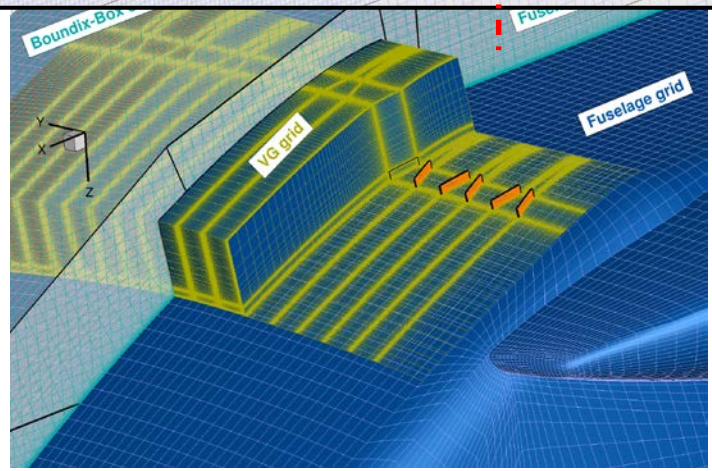
- Overset grids: 4 hierarchical mesh components
  - VGs
  - Fuselage
  - Intermediate Cartesian grid
  - Off-body Cartesian mesh
- Python scripts (Cassiopee):
  - VG mesh generation
  - overset grid assembly



Example of overset grids for a VG array mounted at the back-ramp





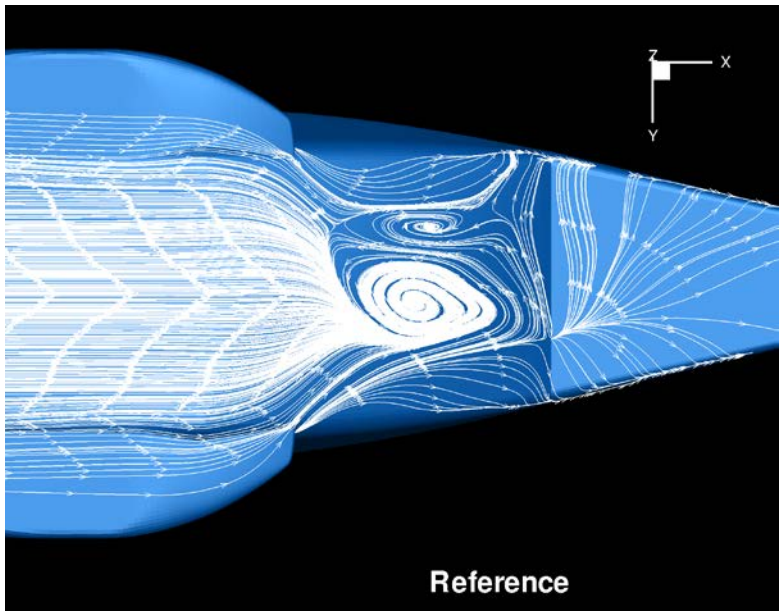


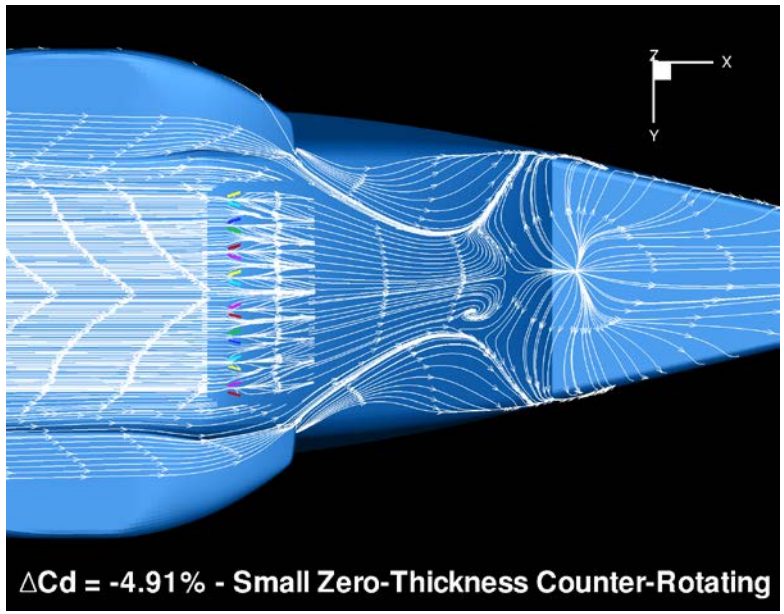
High-density meshes generated with use of Cartesian grids in the near and farfield (≈ 40M grid points)

⇨**Test matrix for 8 VG configurations**

Pre-test numerical investigations

Efficient VG configurations achieve **up to 5% drag reduction** by cumulative effects of flow reattachment, limited device drag, and static pressure recovery

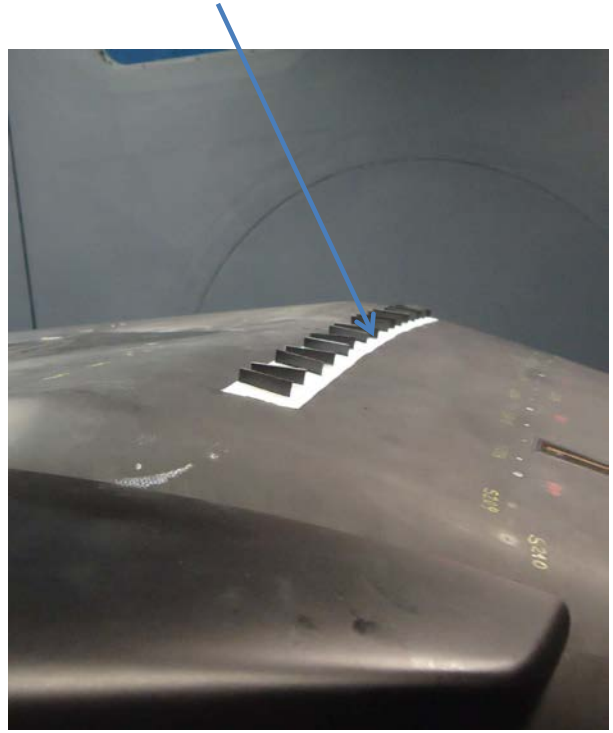




Most promising layouts: optimal position found slightly downstream of the upsweep line

Experimental setup: Helicopter model in the WT

4 co- and counter-rotating VG arrays on the back-ramp: layout and position optimized by CFD

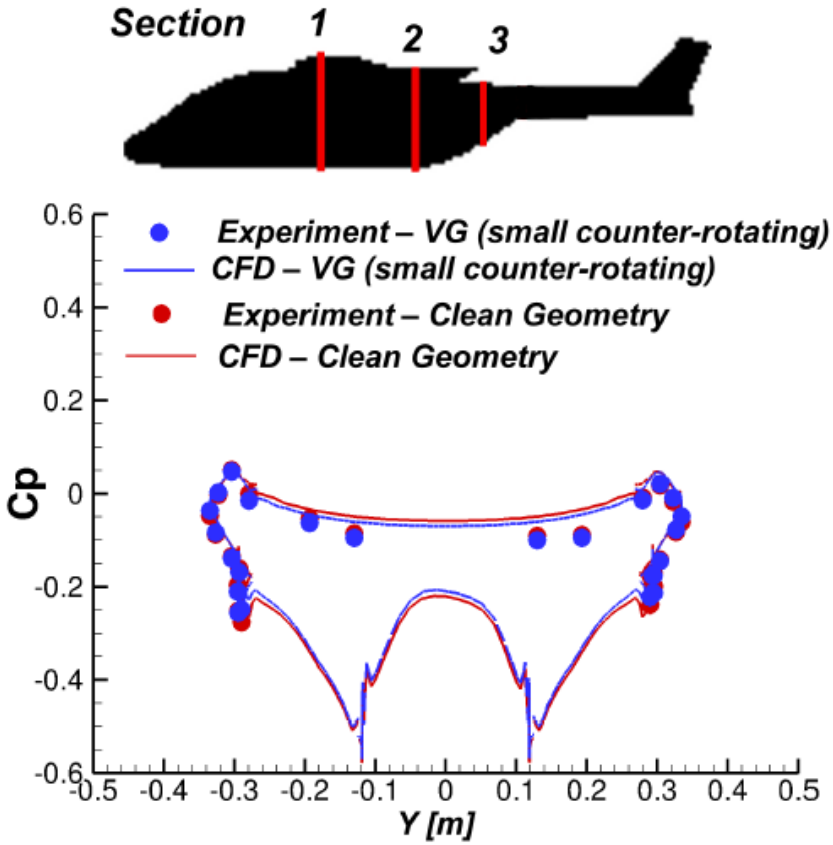


VG Configuration	α_{VG}	L_{VG}	H_{VG}
Small co-rotating	$\pm 15°$	3.6δ	δ
Large co-rotating	$\pm 15°$	4δ	2δ
Small counter-rotating	$\pm 15°$	3.6δ	δ
Large counter-rotating	$\pm 15°$	4δ	2δ

4 configurations tested in WT with small (δ scale) and large (2 δ scale) co-rotating and counter-rotating VG layouts (α_{VG} = ± 15°)

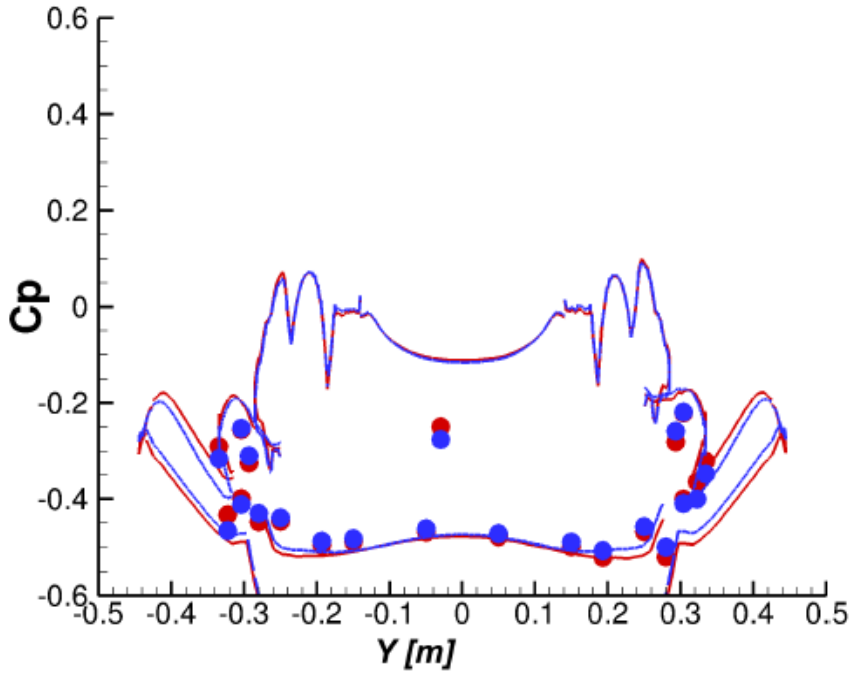
Comparison CFD vs. exp: pressure distribution upstream of the VGs

Angle-of-attack = 2°



(a) Section 1 - X = 1.805 m

- ✓ A very good agreement between CFD and experiments was found
- ✓ On the sections, located upstream the VGs, their effects are very small

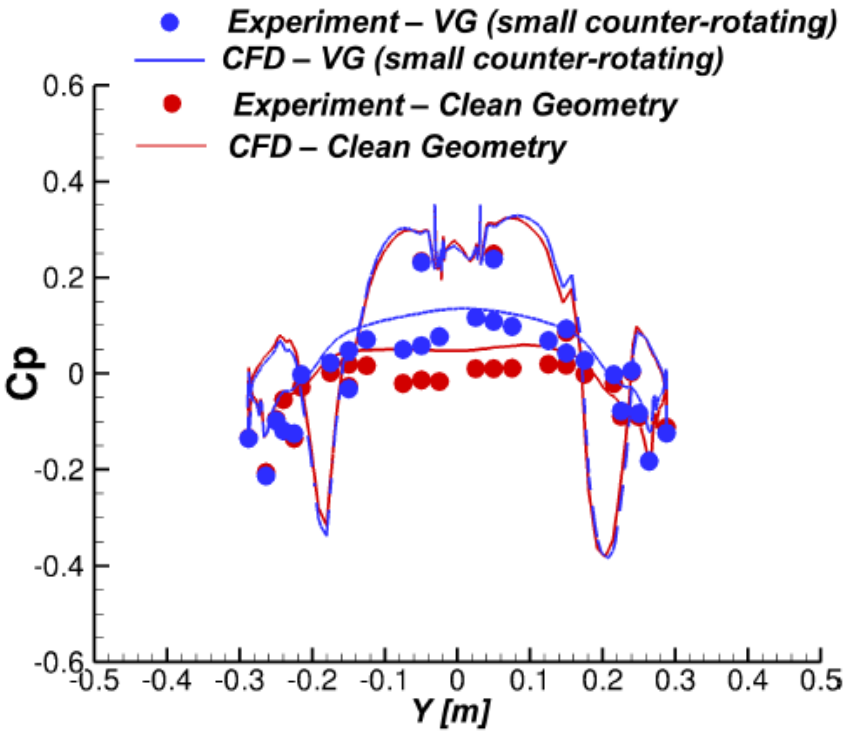


(b) Section 2 - X = 2.468 m

ONERA
THE FRENCH AEROSPACE LAB

Comparison CFD vs. exp: pressure distribution downstream of the VGs

Angle-of-attack = 2°



(c) Section 3 - X = 2.945 m

- ✓ On the section intersecting the back-ramp surface downstream the VGs, both experiments and CFD show an apparent increase of pressure
- ✓ The measured pressure rise is quite similar to CFD calculation

The pressure distribution analysis confirms the benefit introduced by VGs in reducing drag by locally increasing the static pressure field, thus reducing the suction effect responsible for pressure drag rise

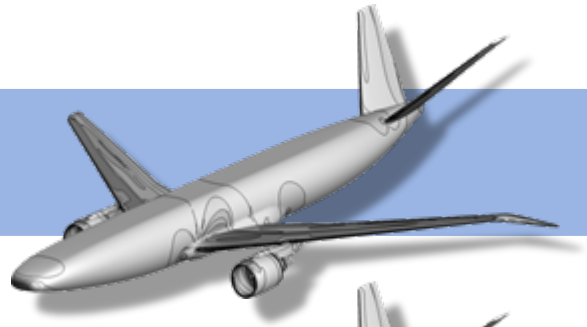
4. Novel configuration design

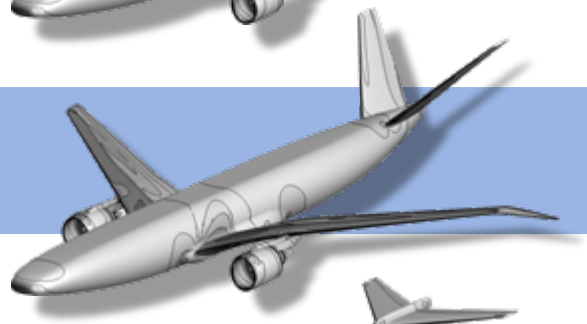
Ref: L. Wiart, O. Atinault, D. Hue, R. Grenon, B. Paluch
« Development of NOVA Aircraft Configurations for Large Engine »,
33rd AIAA Applied Aerodynamics Conference, 2015

Targeted architectures

Baseline

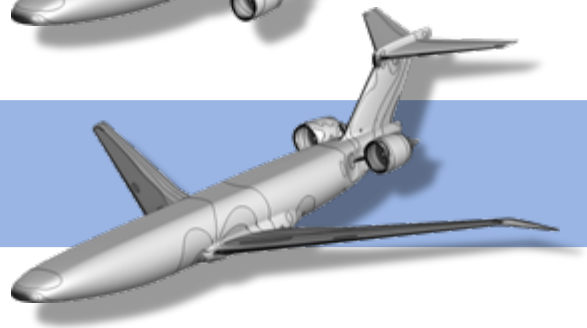
Wide lifting fuselage
High AR wing
Downward oriented winglets
UHBR engine



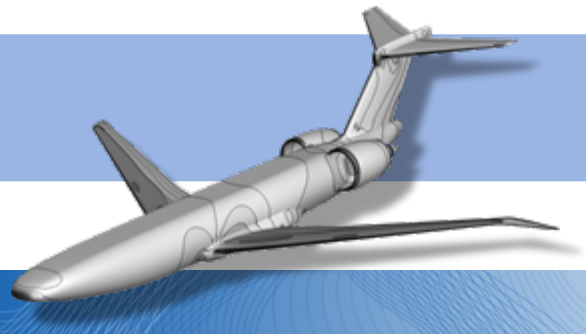


- Gull wing
  - Increased inner wing dihedral to limit landing gear length

- Podded
  - Engines mounted on aft fuselage side

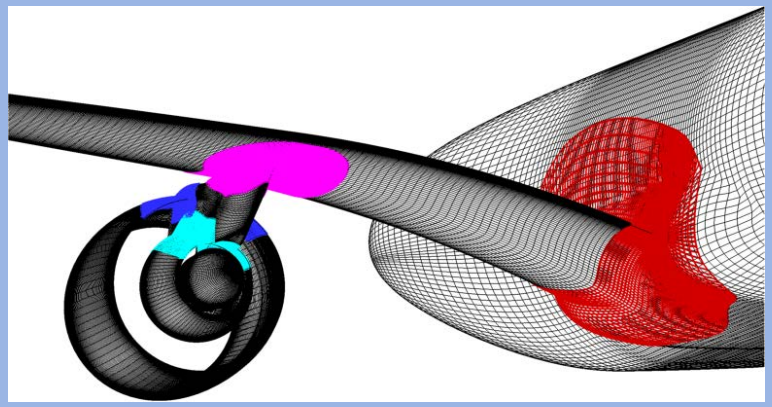


- BLI
  - Engine inlet ingesting the fuselage boundary layer



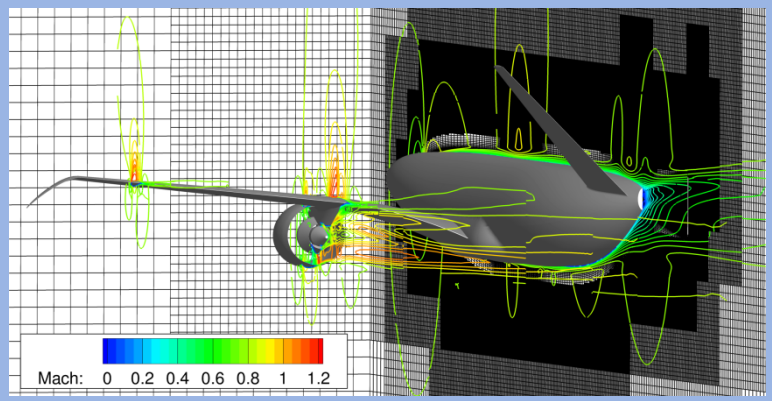
CFD mesh generation

- Use of structured grids for their capability to accurately predict flow features
- Use of advanced grid generation methods to reduce the mesh generation effort



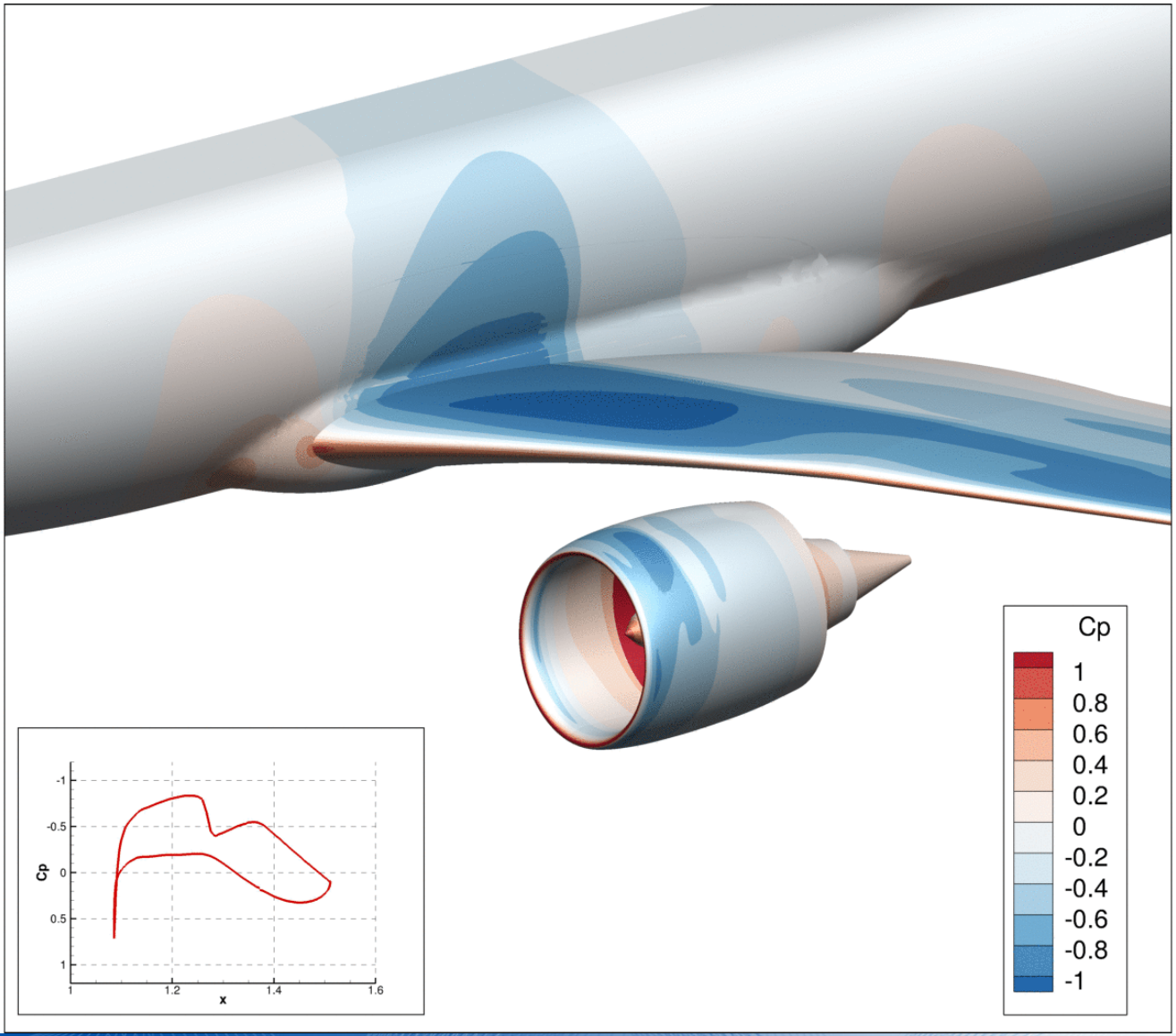
- A complex geometry is split into simpler elements (ex: fuselage/wing/nacelle/pylon…)
- Each element is meshed separately (overset grid approach)
- Collar grids are used to mesh the junction regions

- Off-body adaptive Cartesian grids are then automatically generated around the geometry using an octree approach
- Near body and off-body grids are finally assembled using Cassiopee/Connector
- CFD calculations are carried out with the in-house ***elsA*** flow solver



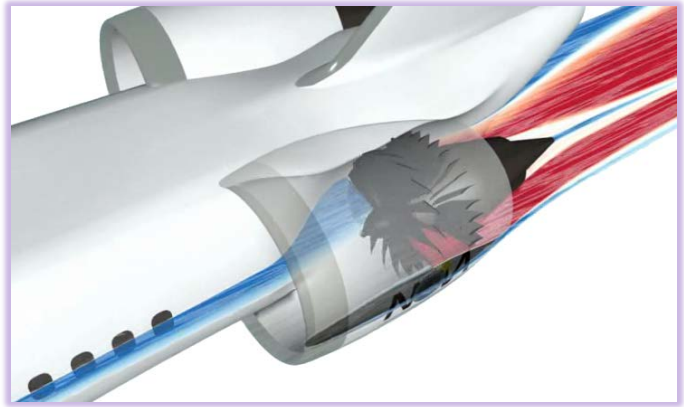
Hue, D., Péron, S., Wiart, L., Atinault, O., Gournay, E., Raud, P., Benoit, C., and Mayeur J., "Validation of a near-body and off-body partitioning methodology for aircraft aerodynamic performance prediction," Computers and Fluids, 2015

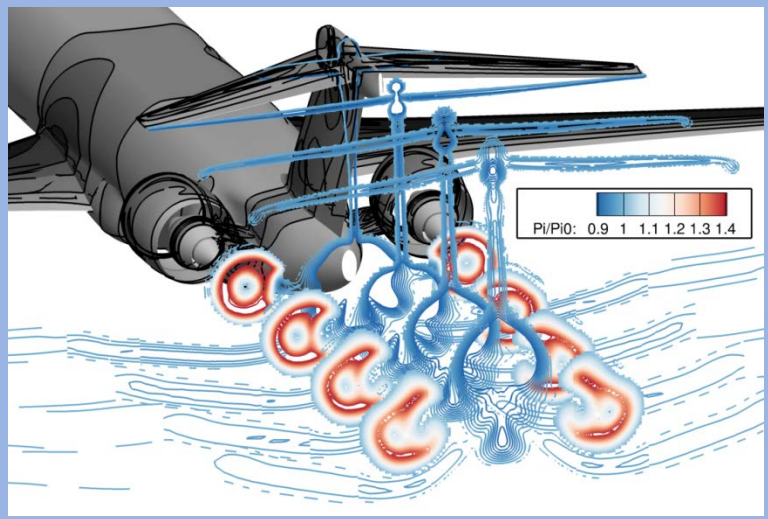
Influence of engine position on skin pressure

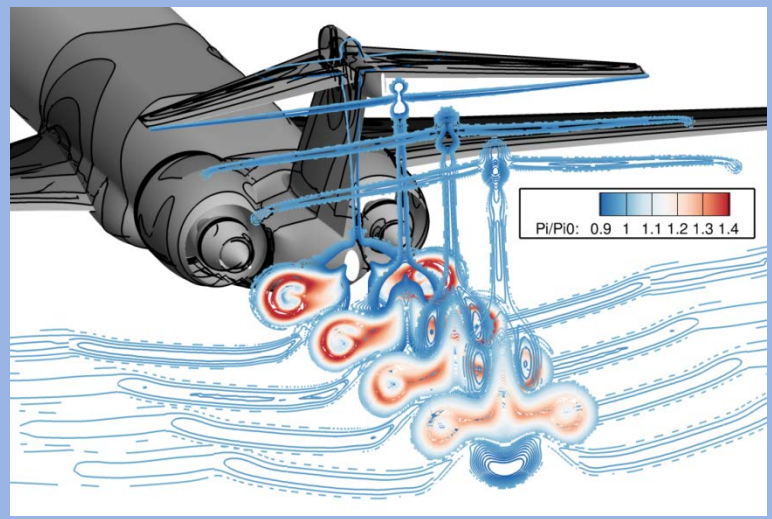


BLI configuration

- By embedding the engine into the fuselage, savings in fuel (due to reduced wetted area and jet/wake losses) and mass are expected
- Deliberately « agressive » design:
  - engine~40% buried
  - short inlet (inlet length/fan diameter ratio~1)







When ingesting the fuselage boudary layer, the engines tend to minimize the aircraft print in the surrounding airflow, indicating better thrust-drag balance

Arntz, A., "Civil Aircraft Aero-thermo-propulsive Performance Assessment by an Exergy Analysis of High-fidelity CFD-RANS Flow Solutions", PhD Thesis, Lille 1 University, 2014.

Perspectives

- Methods
  - Overset mesh intersection by Boolean operators:
    - CFD calculation on a polyhedral mesh
    - (More) “conservative” interpolation coefficients
  - Mixing overset/IBM approaches
- Applications
  - Overset structured & unstructured grids

Cassiopee package

- Documentation/download: http://elsa.onera.fr/Cassiopee
  - Reference doc of Python functions
  - Tutorials
  - Gallery
  - Forum (help, bug reports, suggestion box)
  - Mailing list: dmfn-cassiopee@onera.fr

Thanks for your attention !