A python pre-processing module for Chimera assembly

S. Péron, C. Benoit
ONERA, CFD and Aeroacoustics Dept.

ONERA
THE FRENCH AEROSPACE LAB

return on innovation

Outline of talk

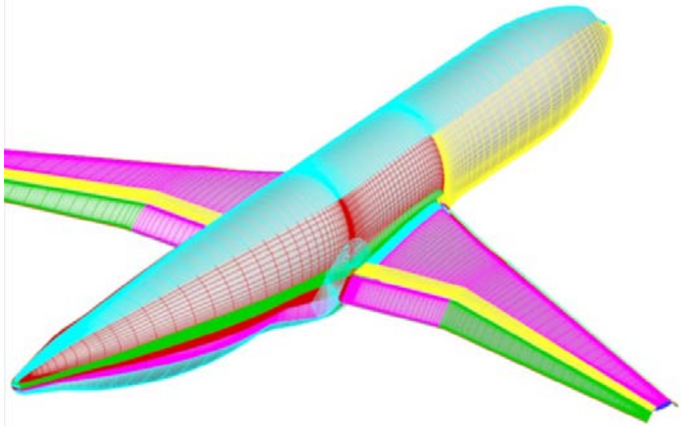
- Motivation
- Principle of the Chimera assembly
- Basic examples
- Implementation in Connector module
- Applications
- Future work

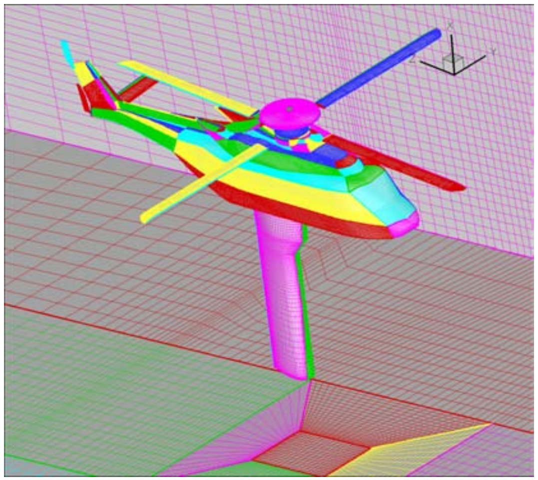
ONERA

THE FRENCH AEROSPACE LAB

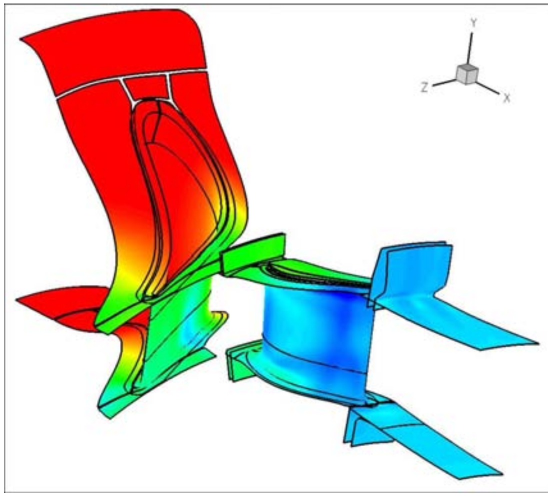
Chimera method at ONERA

Developed in elsA CFD solver for many years
Applications: helicopters, aircrafts, turbomachines, CROR, missiles, launchers

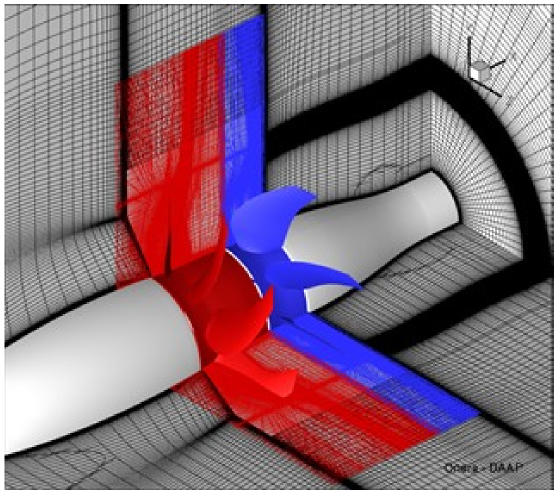




M. Costes (ONERA/DAAP)



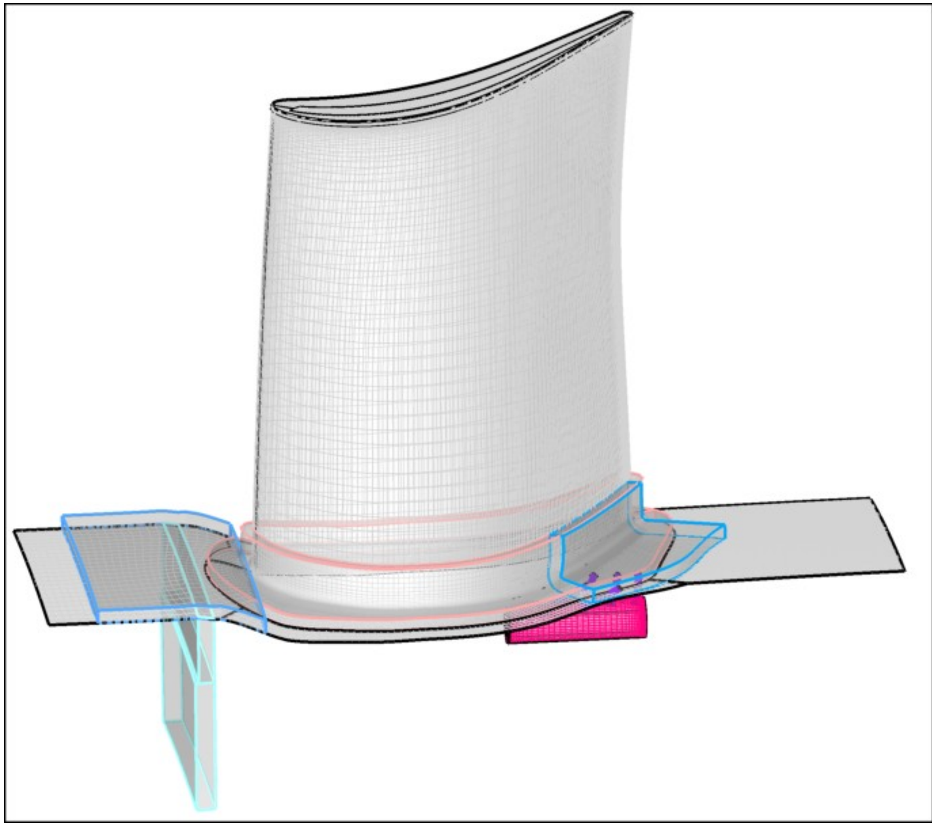
L.Castillon (Onera/DAAP), grid provided by A. Paillassa (Turbomeca)



R. Boisard (ONERA/DAAP)

Motivations

- Increasing complexity of applications
  - Ex: add/remove technological details
- Simplification of Chimera grid assembly:
  - formalization of the relationships between grids
  - development of an external python tool for Chimera pre-processing
  - simplification of setup of configurations
- Tool and data independent from the CFD solver: CGNS-compliant



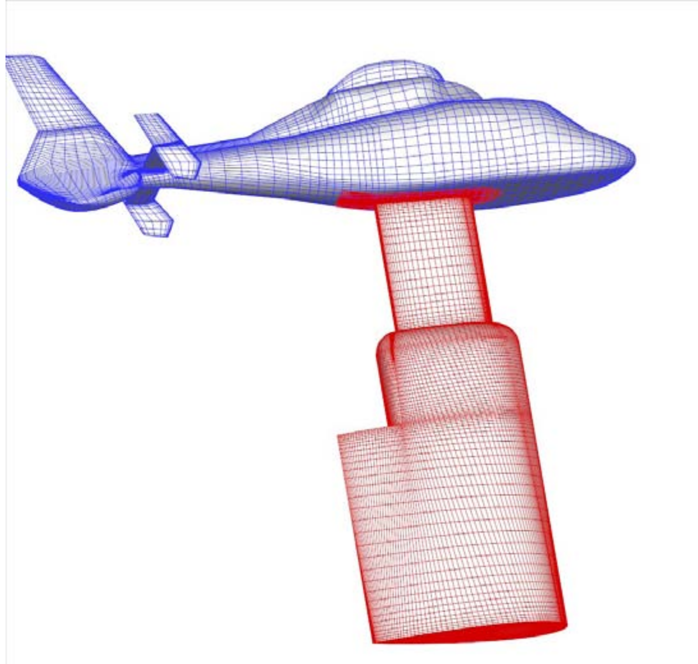
TATEF2 rotor: cooling slots on the hub, blade fillet, tip clearance (court. of L. Castillon)

Outline of talk

- Motivation
- Principles of the Chimera assembly
- Basic examples
- Implementation in Connector module
- Applications
- Future work

Chimera assembly: component

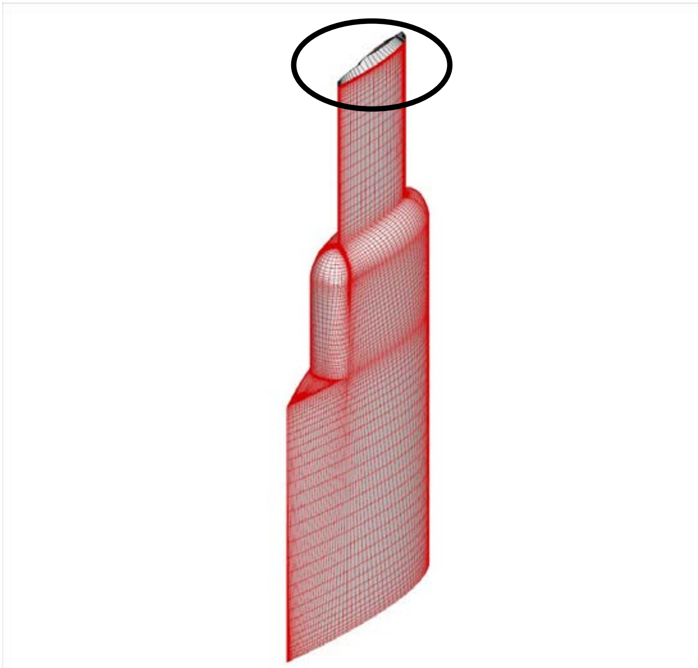
- Component: set of grids describing a body or a piece of body
  - Example: fuselage, strut



ONERA

Chimera assembly: component

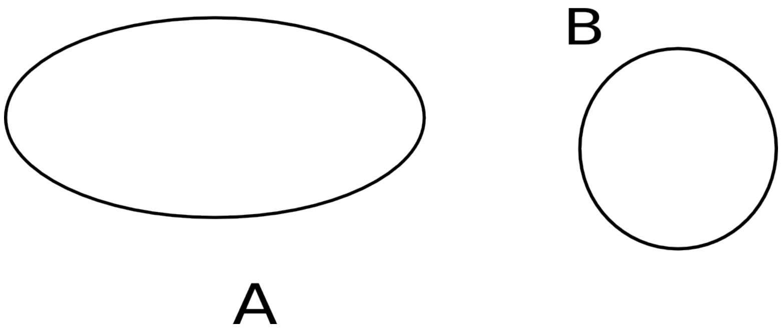
- Component: set of grids describing a body or a piece of body
  - Example: fuselage, strut
- Definition of the body surface starting from the walls of the component
  - Must be closed and watertight
- For each component: one body (except for the background grids) defining one blanking surface



Closure (in black) of the strut surface

1st type of assembly: neutral ANB

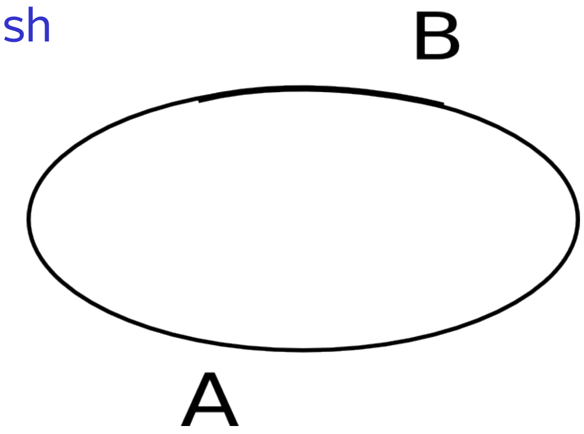
- Neutral assembly ANB:
  - Separated bodies
  - No assembly at wall borders
- Blanking:
  - A body blanks B grids
  - B body blanks A grids
  - A and B bodies may blank other components (background grids…)



ONERA

2nd type of assembly: inactive A 0 B

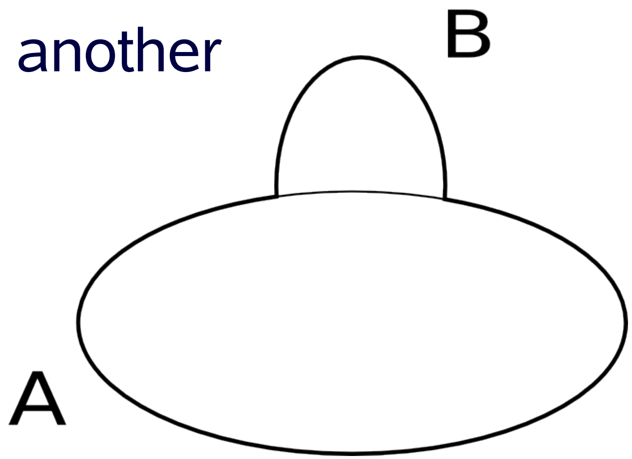
- Inactive assembly A 0 B:
  - components define the same body (e.g. buffer mesh onto a wall)
  - Not symmetric



- Blanking:
  - A body does not blank B
  - B body does not blank A
  - Only A body may blank other components (background grids…)

3rd type of assembly: union A+B

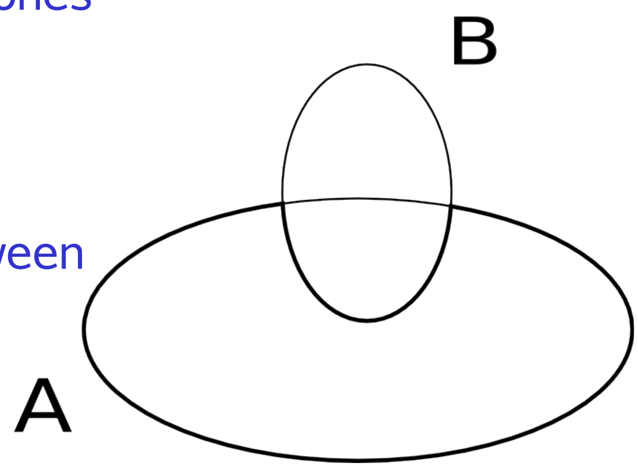
- Union A+B: addition of a body onto another body
  - Not symmetric: A+B different from B+A
  - B grids lie on the wall of A
  - Example: spoiler onto a wing



- Blanking:
  - B body blanks A grids
  - A body does not blank B grids
  - A and B bodies may blank other components (e.g. background grids)
  - A+B describes 'virtually' a new body which blanks other components by applying both blankings

4th type of assembly: difference A-B

- Difference A-B: digging of a body
  - Not symmetric
  - Example: slot in a blade
  - Wall borders are also defined as overlap borders
- Donor zones for Chimera transfers due to the doubly defined BC:
  - Grids from B (resp. A) must be defined as donor zones to apply the overlap BC on A (resp. B)
- Blanking:
  - A (resp. B) does not blank B (resp. A)
  - Blanking surface results from CSG difference between A surface and B surface



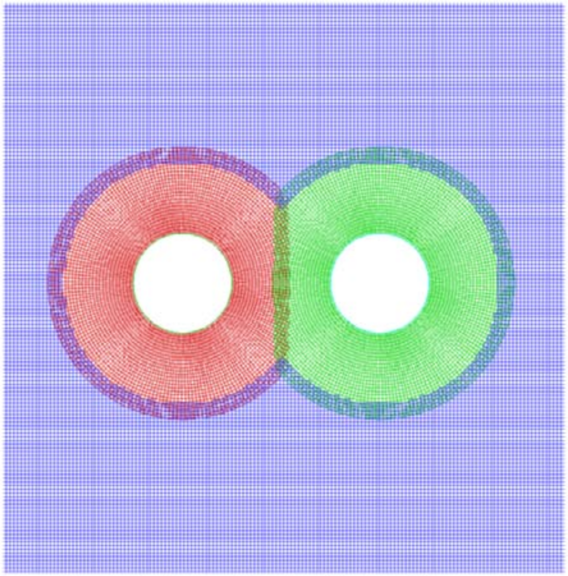
ONERA
THE FRENCH AEROSPACE LAB

Outline of talk

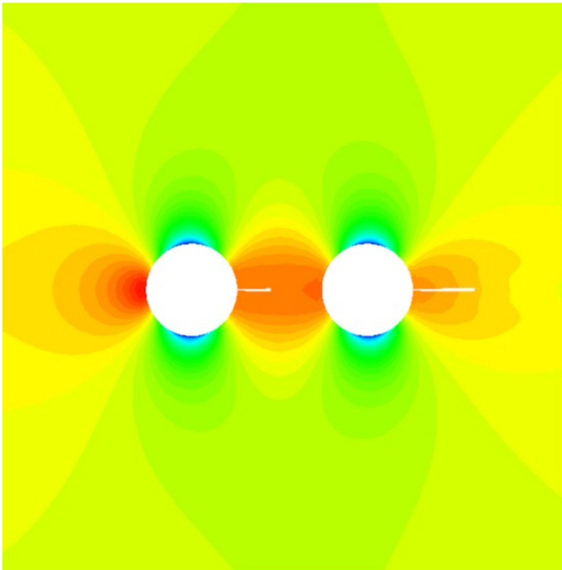
- Motivation
- Principles of the Chimera assembly
- Basic examples
- Implementation in Connector module
- Applications
- Future work

Example of neutral assembly (ANB)NC

A: red grid, B: green grid, C: blue background grid



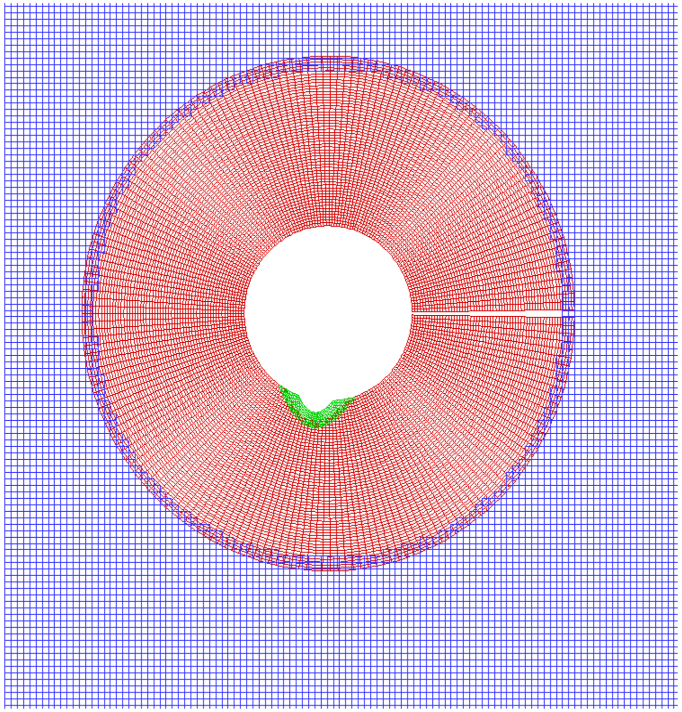
View of the mesh after blanking and overlap optimization (blanked cells not visualized)



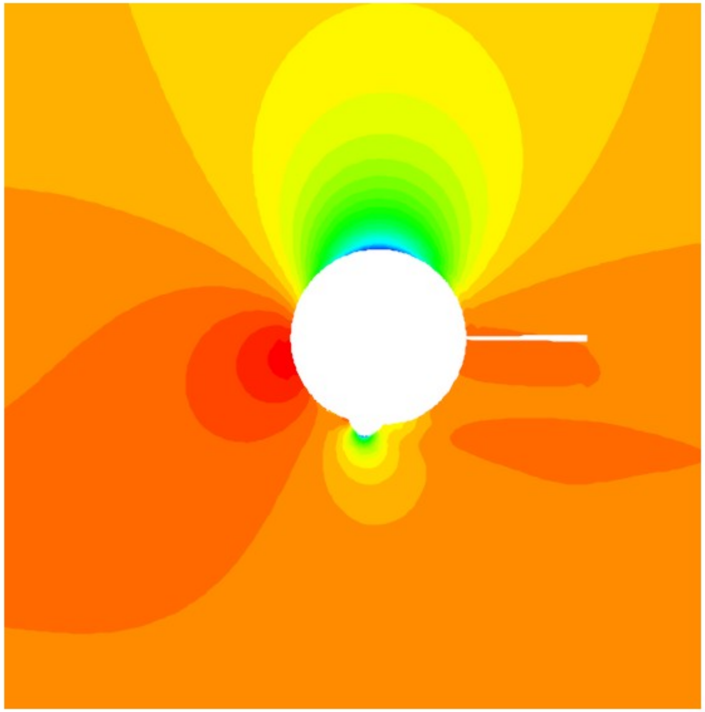
Density contours, Euler computation, Mach = 0.2

Example of assembly (A+B)NC (union+neutral)

A: red grid, B: green grid, C: blue background grid



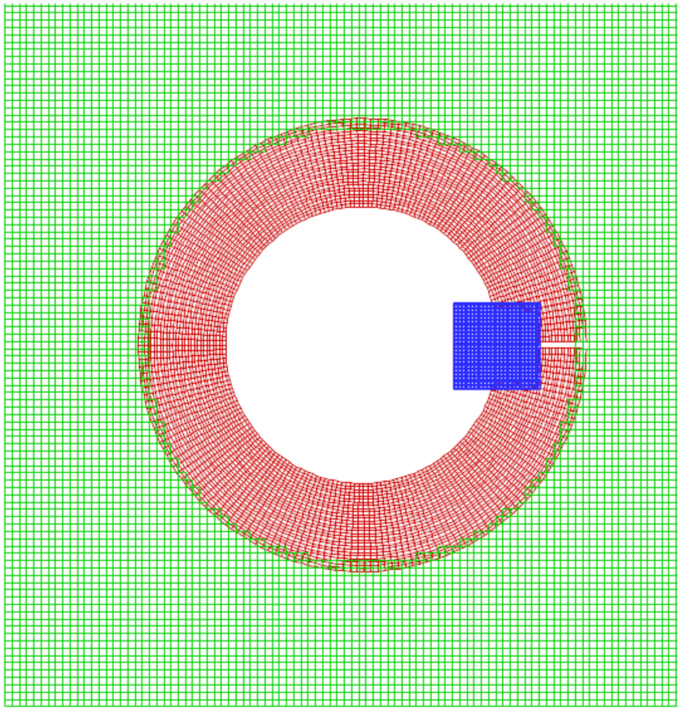
View of the mesh after blanking and overlap optimization (blanked cells not visualized)



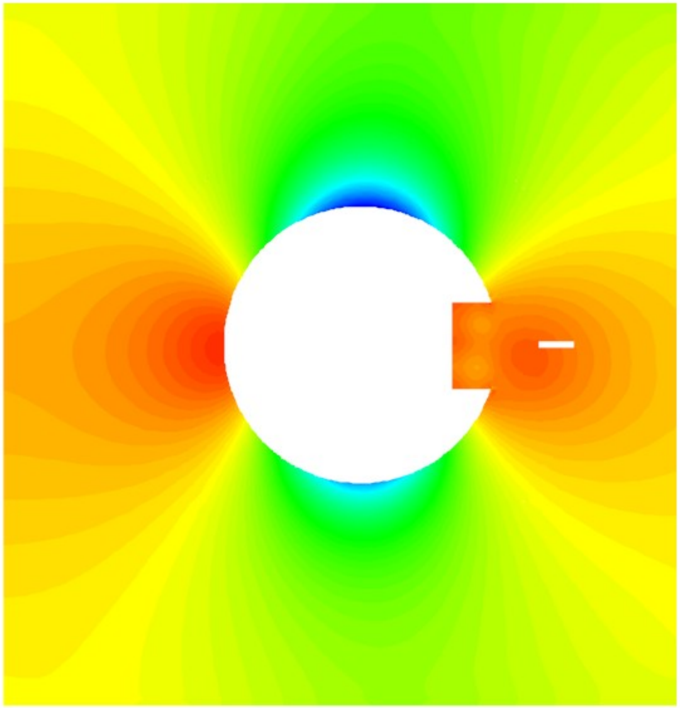
Density contours, Euler computation, Mach = 0.2

Example of assembly (A-B)NC (difference+neutral)

A: red grid, B: blue grid, C: green background grid



View of the mesh after blanking and overlap optimization (blanked cells not visualized)



Density contours, Euler computation, Mach = 0.2

Outline of talk

- Motivation
- Principles of the Chimera assembly
- Basic examples
- Implementation in Connector module
- Applications
- Future work

Chimera assembly module: Connector

- Python module with functions dedicated to specific Chimera functionnalities (blanking, overlap optimization…)

- Input/output data: python CGNS trees:
  - Input: a python CGNS tree (or pytree) describing all the zones of the problem. Each base defines a Chimera component
  - Output: the pytree, for which each blanked zone contains the CGNS OversetHoles node, providing the indices of blanked cells

Chimera assembly module: functions

- Python module with several functions:
  - blankCells : blanking
  - optimizeOverlap: overlap optimization
  - applyBCOverlaps: definition of interpolated points defined by overlap connectivity
  - setDoublyDefined: determination of cells which are defined by a physical BC or an overlap connectivity on a border
  - testInterpolations: test of the Chimera connectivity (interpolated, extrapolated or orphan cells)

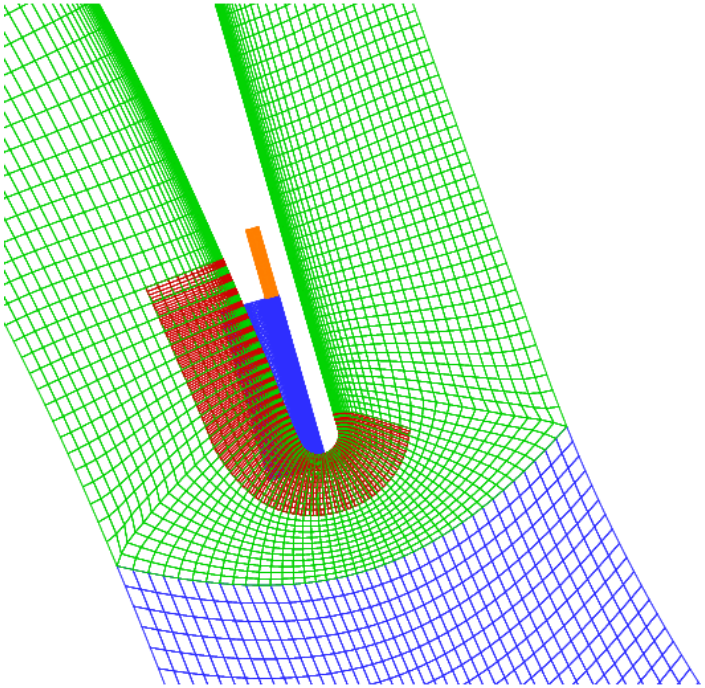
Steps

- Identify the components and their relationships
- Define mesh, BCs, connectivity in the pyTree
- Define doubly defined BCs
- Identify bodies for blanking (and close if necessary the bodies)
- Blanking and/or overlap optimization
- Creation of OversetHoles node for blanked zones

Define doubly defined BCs: composite definition of surfaces

- For a border defined by both a physical BC and an overlap BC
  - Cells interpolable from one donor zones are set to be interpolated
  - Physical BC applied if cell not interpolable

- Useful to add slots in a geometry

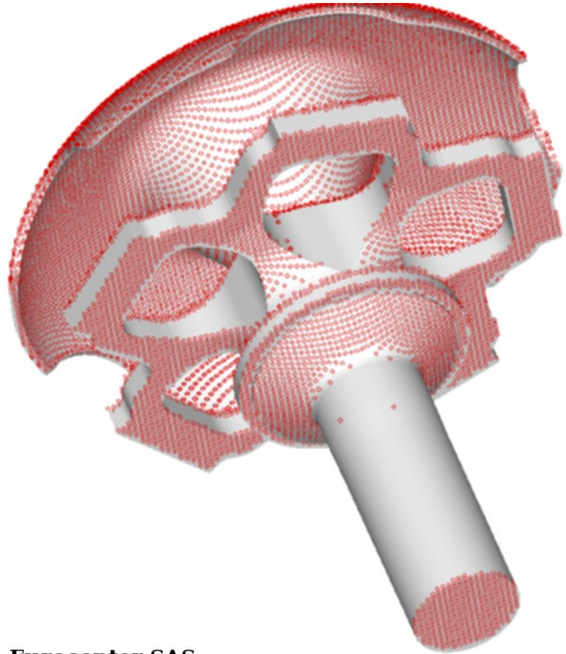


Example of doubly defined BC:
- on the wall boundary of the blade (green)
- on the lateral borders of the slot (blue)

blankCells: blanking function (1)

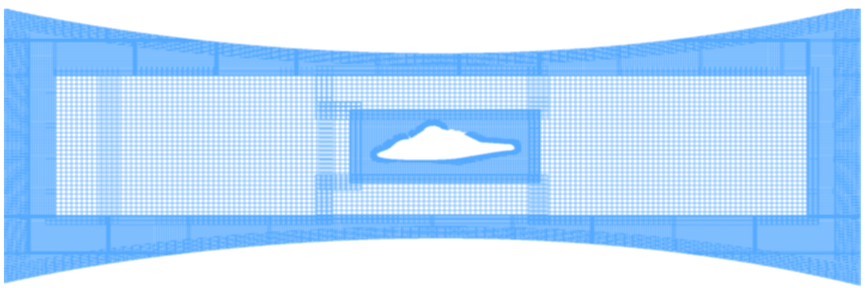
- Use of "Object X-Rays" technique introduced by Meakin (2001):
  - rays cast through bodies, providing intersection points with the body
  - cells lying between intersection points are blanked
  - most accurate hole-cutting technique

- Inverted blanking is possible to blank outside a body (e.g. for internal flows)



Eurocopter SAS

View of the X-Ray intersection points with a rotor hub



Blanking of Cartesian grids out of the curved boundaries

blankCells: blanking function (2)

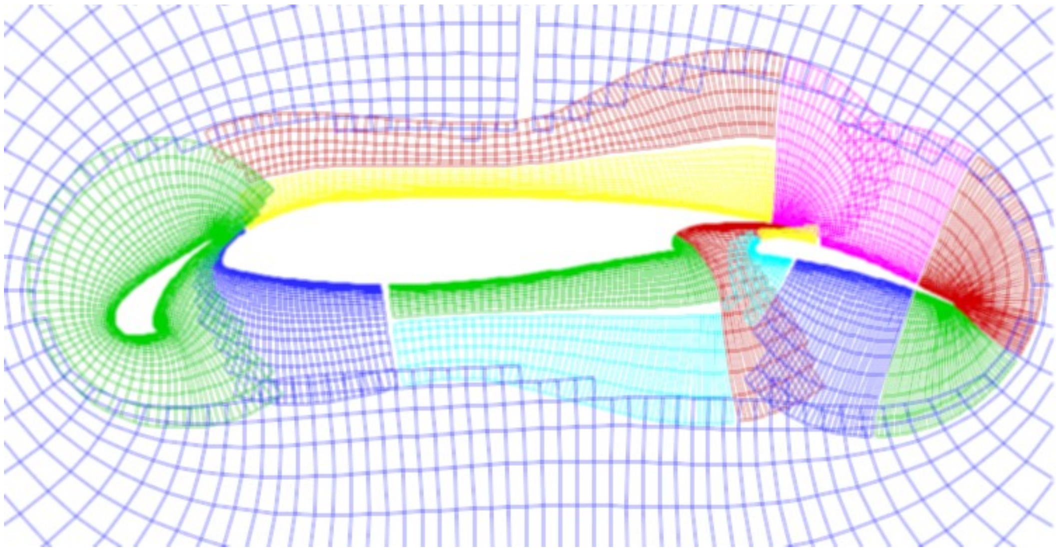
- Input data:
  - the pytree defining the mesh
  - the objects (bodies, parts of bodies, closed if necessary) into which cells are blanked
  - the relationships between the different components, rendered into an assembly matrix

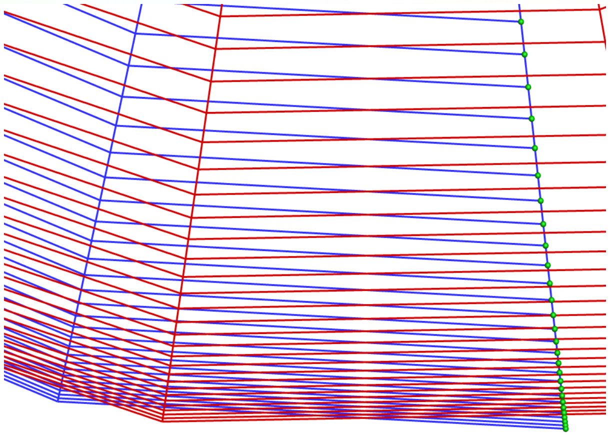
blankCells: blanking function (3)

- Input data:
  - the pytree defining the mesh
  - the objects (bodies, parts of bodies, closed if necessary) into which cells are blanked
  - the relationships between the different components, rendered into an assembly matrix
- Advantages:
  - components can be added or removed easily for other simulations
  - one can defined objects/blanking surfaces separately → blanking open to other usages (post-processing…)

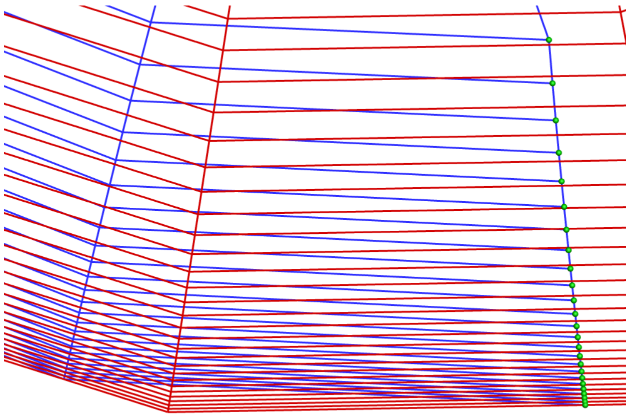
optimizeOverlap: overlap optimization

- Overlap optimization based on PEGASUS algorithm (2002)
- Multiple wall definition possible, if several body grids overlapping on the surface





Interpolated points (green)



Interpolated wall point is projected onto the other wall, the other interpolated points are shifted, with a damping

Outline of talk

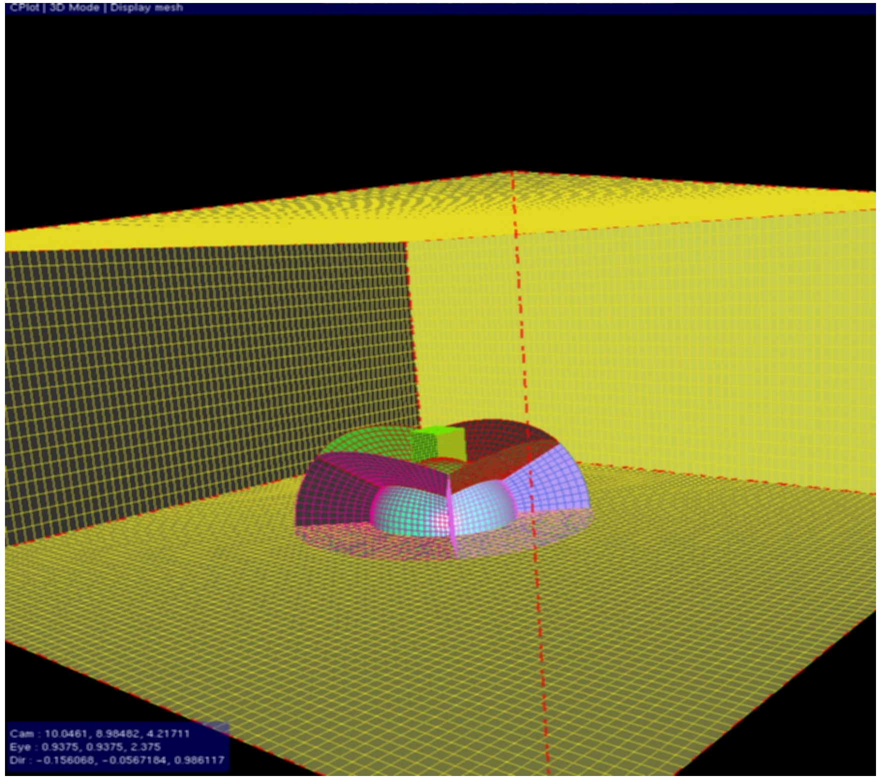
- Motivation
- Principles of the Chimera assembly
- Basic examples
- Implementation in Connector module
- Applications
- Future work

Application to a configuration defined by a composite surface (1)

- 3 components:
  - the sphere (5 grids)
  - the background (1 grid, yellow)
  - the slot (1 grid, green)

- The slot is defined by doubly defined boundary conditions on its lateral borders

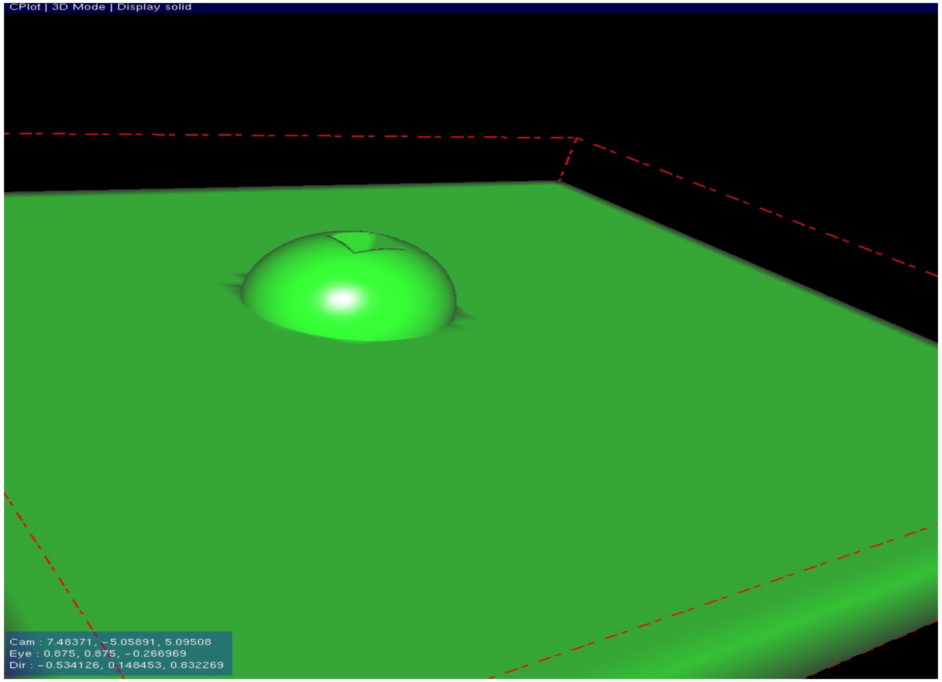
- Objective: blanking of the background grid using a composite surface



ONERA
THE FRENCH AEROSPACE LAB

Application to a configuration defined by a composite surface (2)

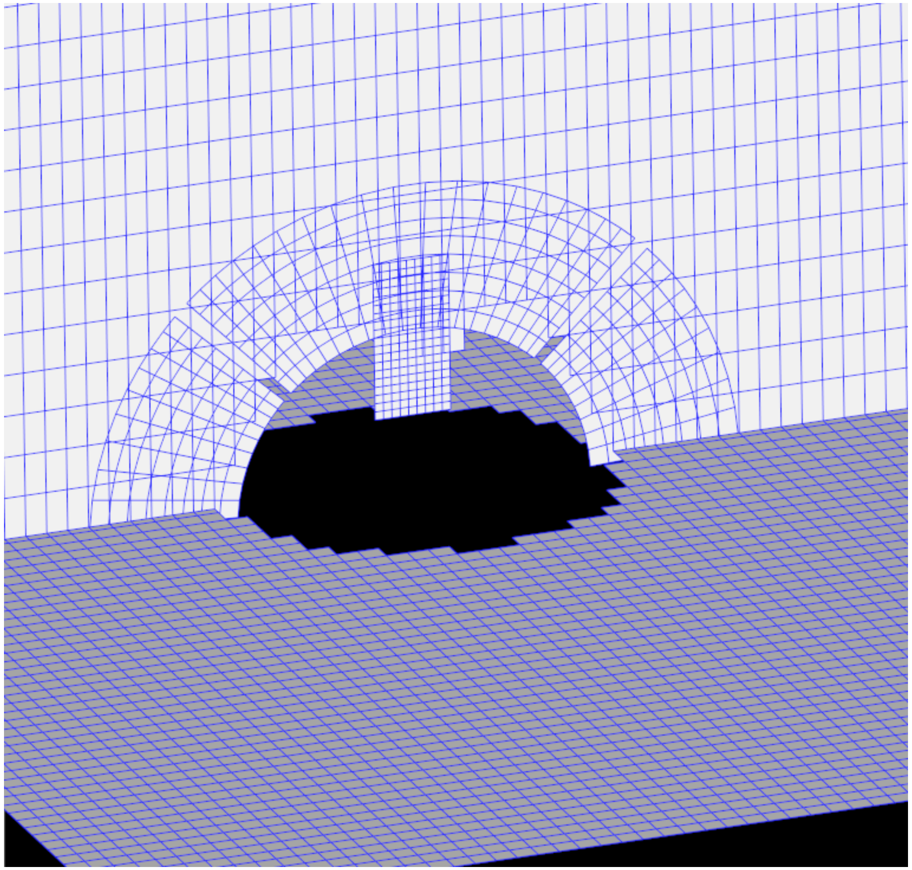
- Composite surface composed by half a sphere, pierced by a slot, lying on a flat surface
- Results from CSG boolean operations:
  - Union between the half-sphere and the flat surface
  - Difference between the slot and the sphere



View of the composite body used for blanking

Application to a configuration defined by a composite surface (3)

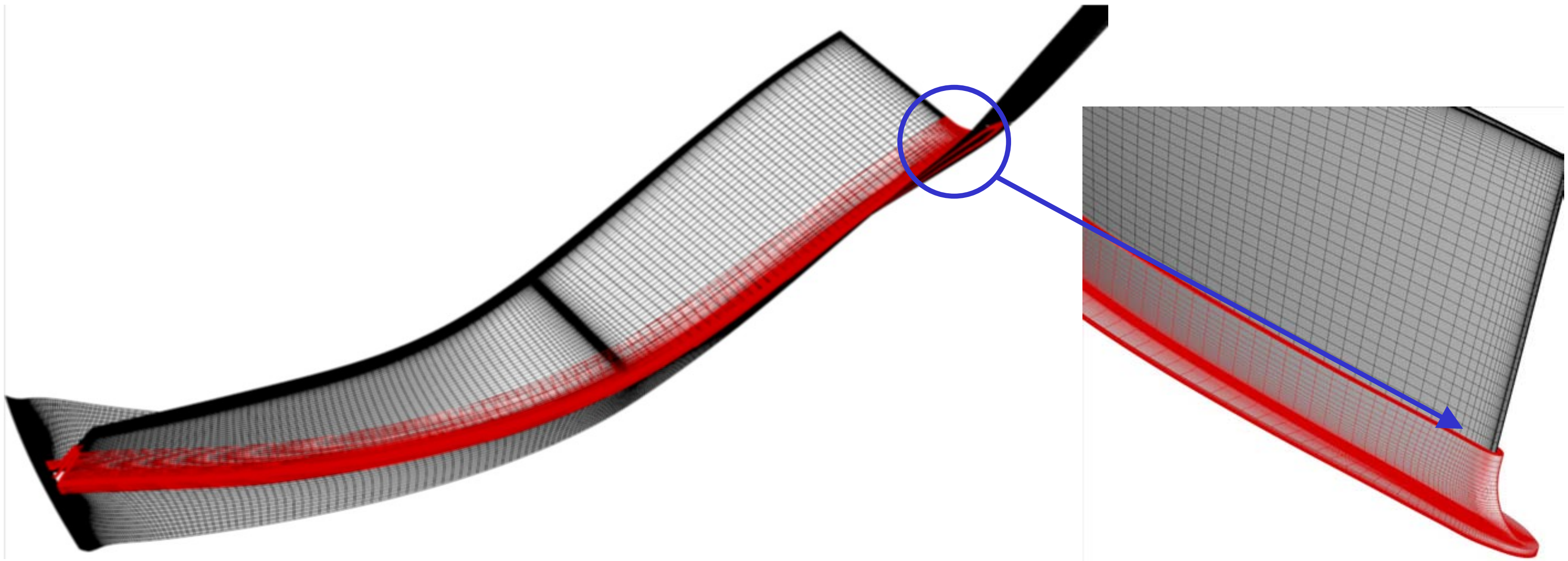
- Composite surface composed by half a sphere, pierced by a slot, lying on a flat surface
- Results from CSG boolean operations:
  - Union between the half-sphere and the flat surface
  - Difference between the slot and the sphere



View of the composite body used for blanking

Application to the blanking under blade fillets

- Assembly A+B: B body blanks A grids
- Component A: rotor blade, O-H type topology
- Component B: fillet, O-type mesh, connecting the hub and the blade

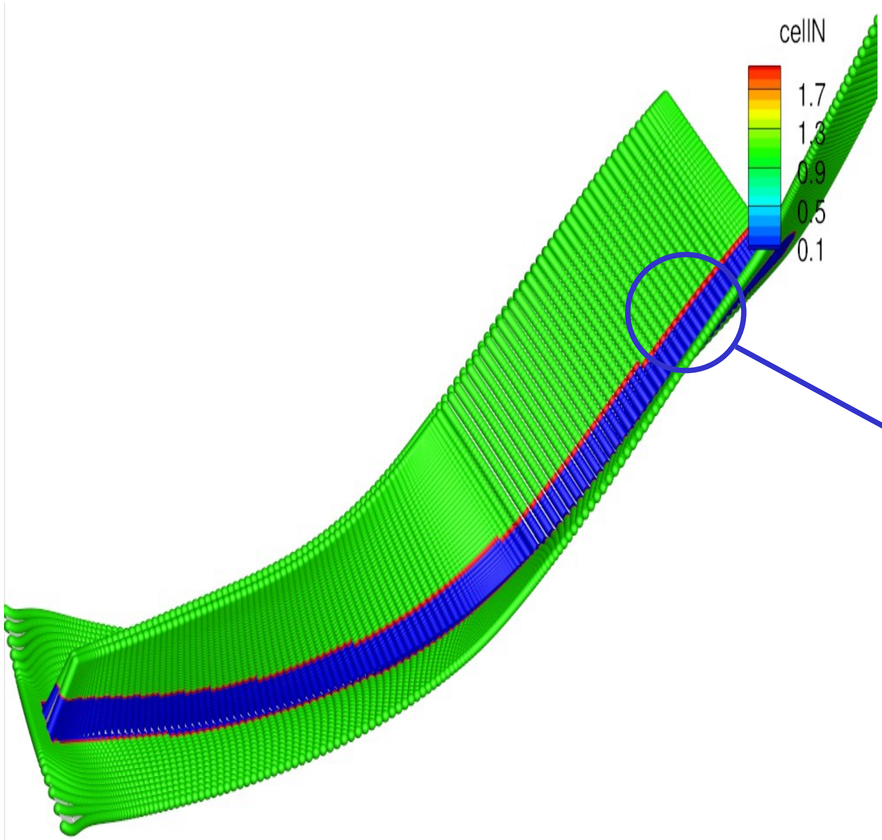


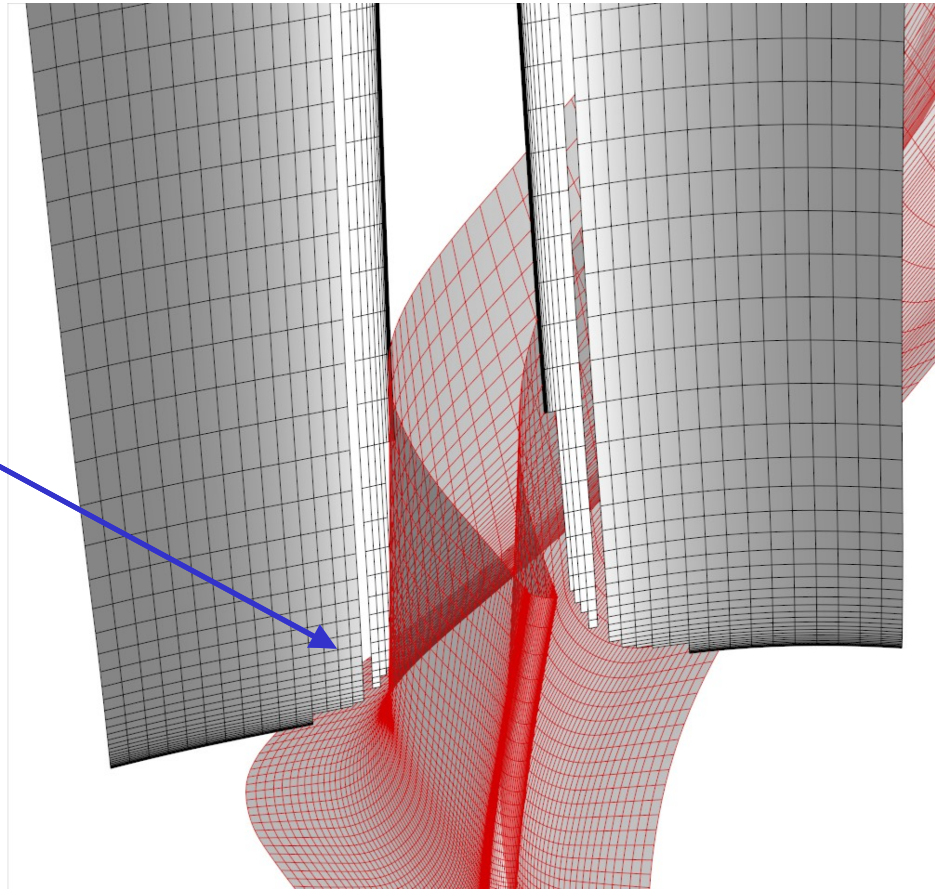
Blade leading edge

Configuration and mesh provided by Turbomeca

Application to the blanking under blade fillets

View of the cell nature on the blade surface (j=1 of the blade O-mesh)
(blue: blanked, red:interpolated, green: computed)

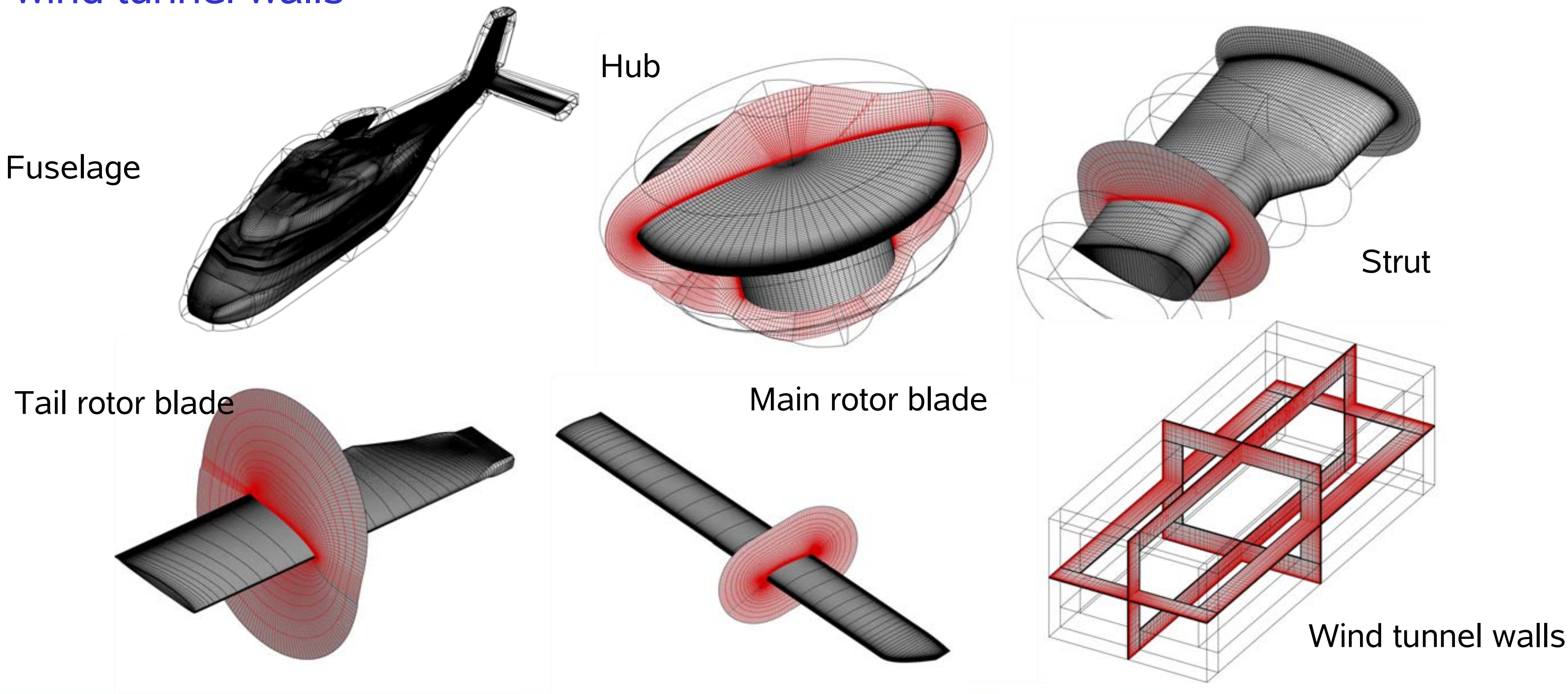




Slice in the blade mesh centers after blanking by the fillet
Blanked cells not represented

Application to the GoAhead configuration (1)

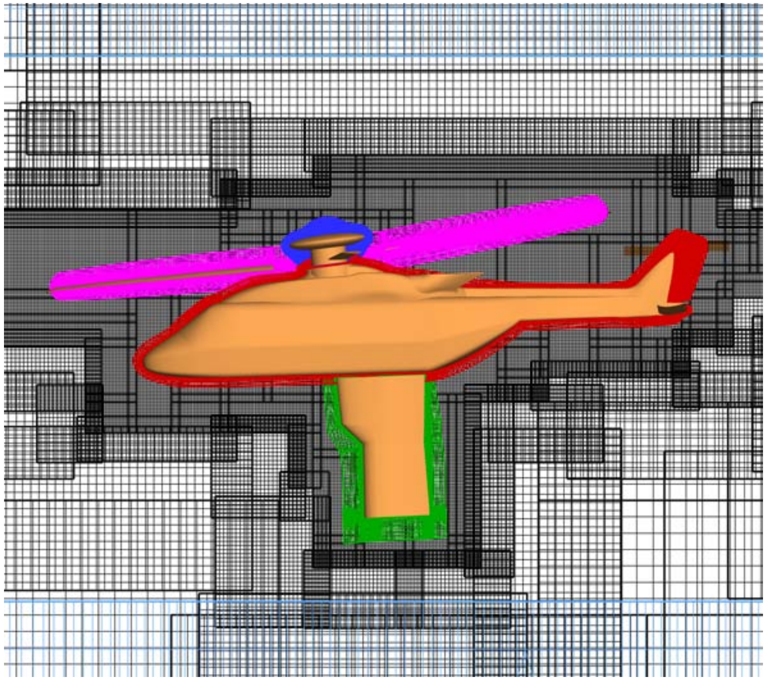
- NH90 helicopter in a wind tunnel
- Blanking performed at each iteration for moving bodies (main and tail rotor) into the solver
- Blanking performed by Connector for the fuselage, the strut, the hub and wind tunnel walls



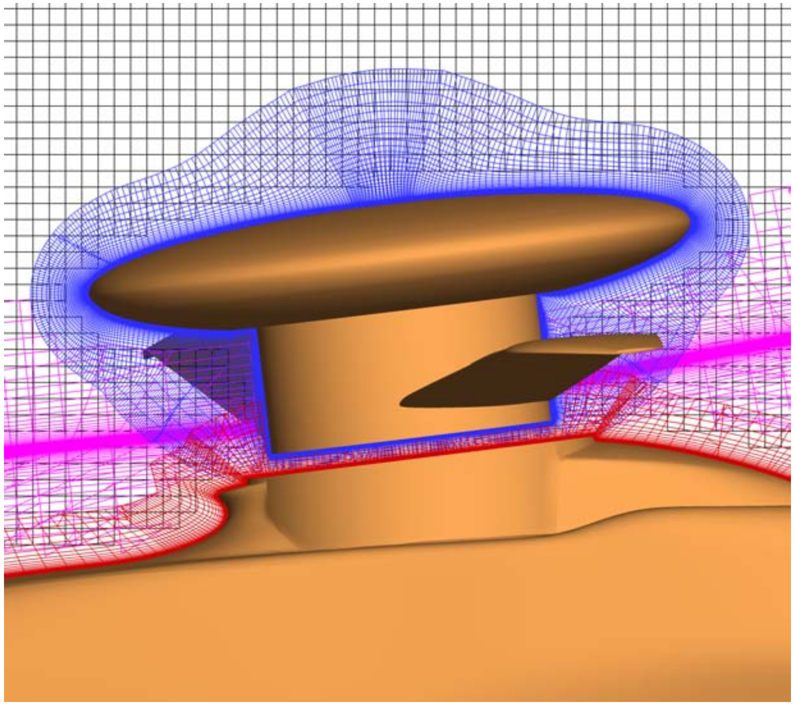
ONERA

Chimera assembly of the different parts of the GoAhead configuration (2)

- Blanking starting from the bodies (strut body is closed before)
- Neutral assembly: fuselage + hub, fuselage + background Cartesian grids, hub + background Cartesian grids
- Union: fuselage + strut



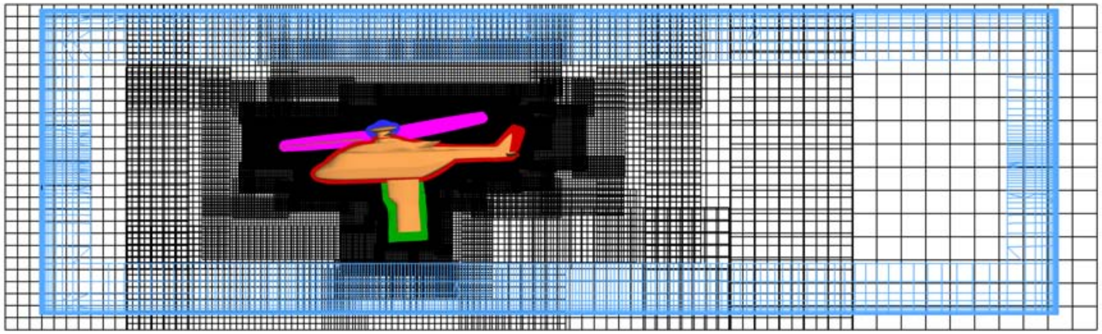
View of overset grids near the fuselage



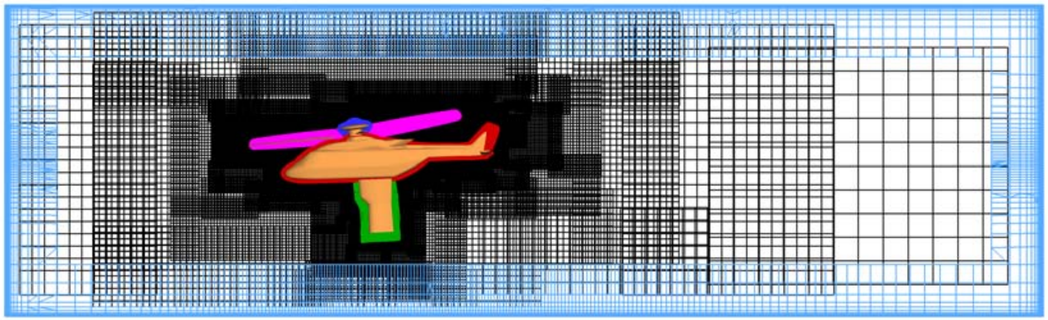
Detailed view of the blanking near the rotor hub

Application to the GoAhead configuration (3)

- Neutral assembly between the wind tunnel walls and the Cartesian background grids
- Background grids out of the computational domain: inverted blanking

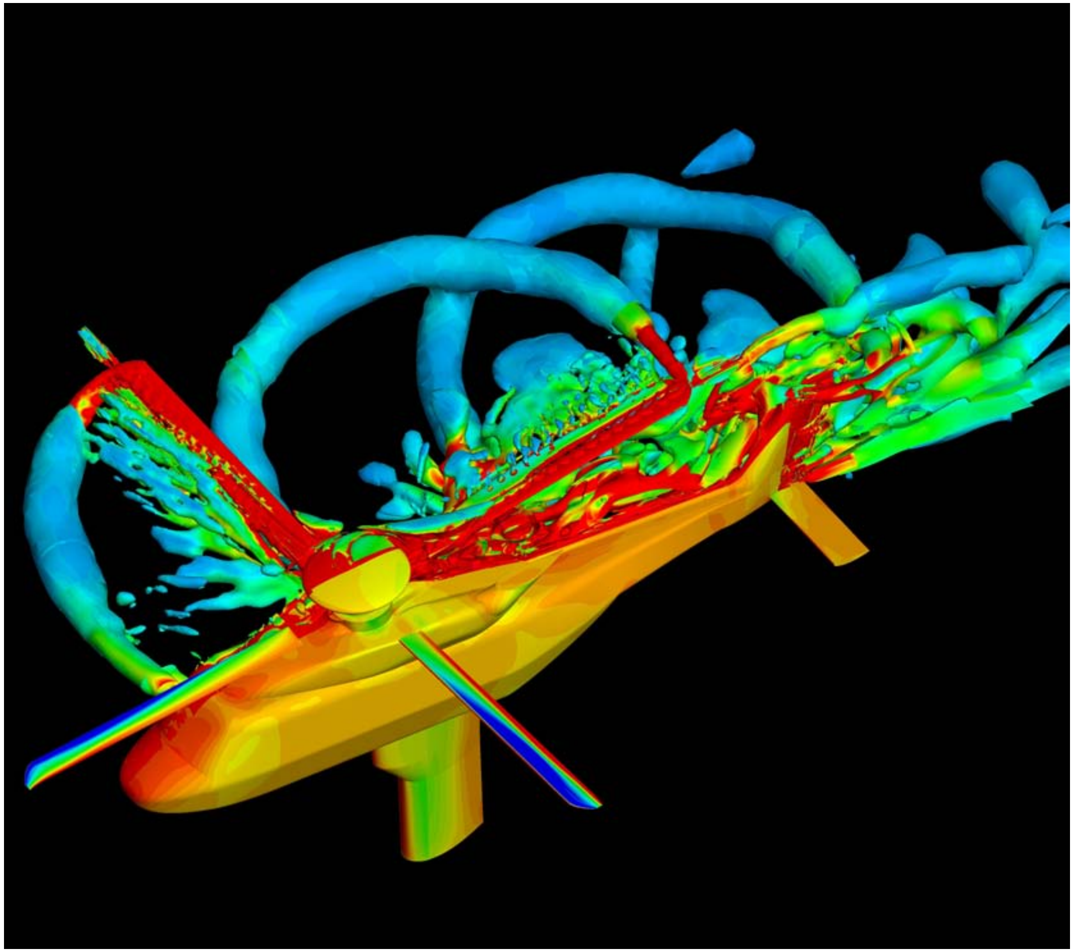


Global view: Cartesian grids out of the domain bounded by the blue grids



View after inverted blanking

Application to the GoAhead configuration (4)



Isocontours of the Q-criterion (T. Renaud)

elsA RANS computation (court. of T. Renaud)

T. Renaud, M. Costes, S. Péron, 36th ERF, 2010

Conclusions and perspectives

- Connector:
  - Pre-processing Chimera python module
  - Input/output data: CGNS python trees
  - Enables to add technological details in a relatively simple way

- Future work:
  - Turbomachine CFD simulations currently performed, with several technological details assembled by Connector
  - Storage of interpolation coefficients in the python CGNS tree (already computed)
  - Use of Connector in unsteady CFD simulations