

AIAA-2017-3572

AIAA 35th Applied Aerodynamics Conference, Denver, Colorado, June 5-9, 2017

Implementation of a Body Force Model in OVERFLOW for Propulsor Simulations

H. Dogus Akaydin*,

Science and Technology Corporation, NASA Ames Research Center, Moffett Field, CA

Shishir A. Pandya †,

NASA Ames Research Center, Moffett Field, CA

We present an implementation of a propulsor model based on body force method into the Overflow computational fluid dynamics code to model turbofan engines and propulsors of similar type. The model estimates the forces imparted on the fluid by the blade camber

Applied Modeling and Simulation Seminar – 19 October 2017

Computational Aerosciences Branch

NASA Ames Research Center

Outline/FAQ

What?

We implemented in Overflow a new body force method to model aircraft propulsors.

Why?

Because the future aircraft engines may have to deal with more of inlet distortions. This method makes modeling engines under such conditions more accurate.

How?

We developed code and practices to automate generation of the grids and computation of necessary quantities within CGT and Overflow framework.

Validation?

We validated the implementation on the SDT (Source Diagnostics Test) fan with R4 rotors. This is a benchmark case with no inlet distortion.

Application?

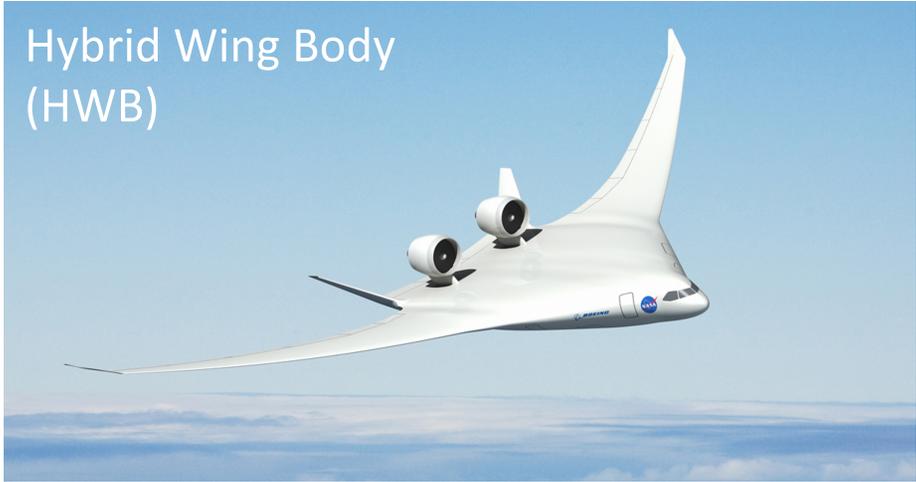
We utilized the implementation on a D8 aircraft which features inlet distortions due to boundary layer ingestion. Also compared against wind-tunnel data.

Lessons learned?

The model seem to work reasonably well but we need more data from tests with variety of inlet distortions. There is much room for improvement on the implementation, on the model and much need for a BLI test data.

Why?

NASA is sponsoring R&D on fuel-efficient transport aircraft.



Hybrid Wing Body (HWB)

NASA, Boeing



STARC-ABL / TCT

NASA



D8

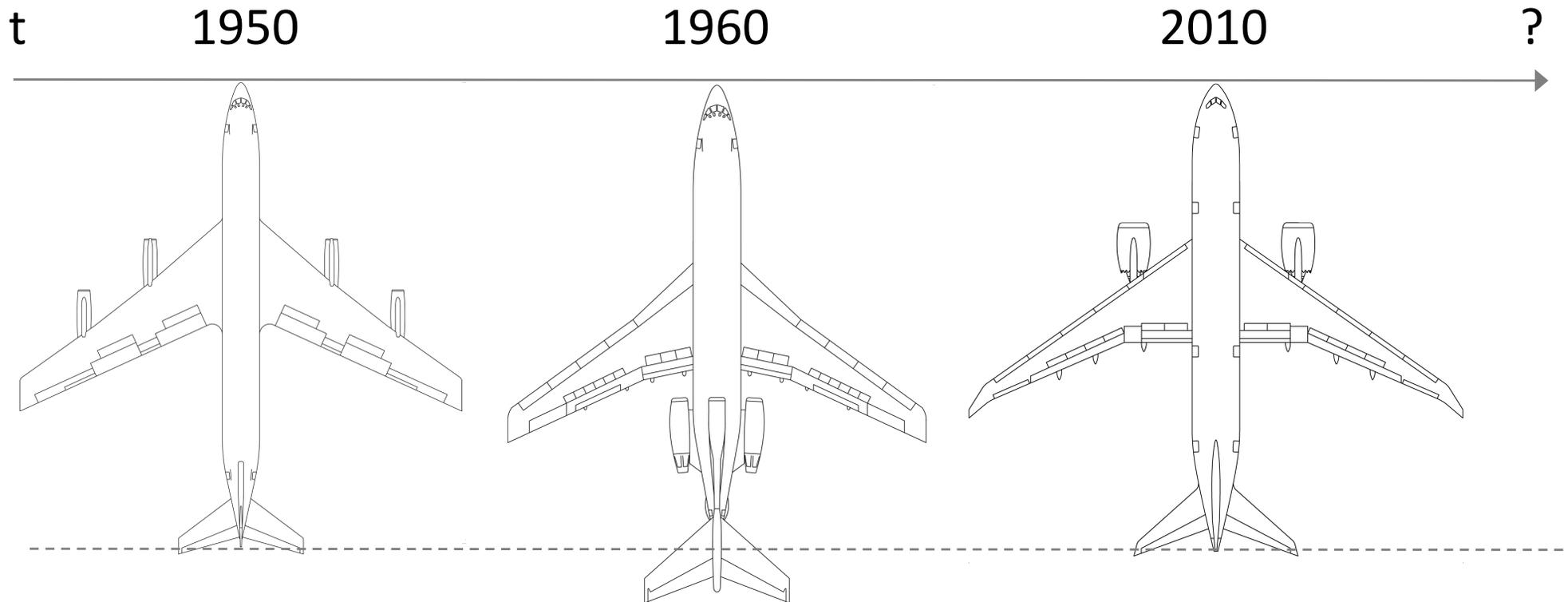
NASA, MIT, Aurora Flight Sciences, Pratt & Whitney



Truss-Braced Wing (TBW)

NASA, Boeing

An overview of the state of the art: plan



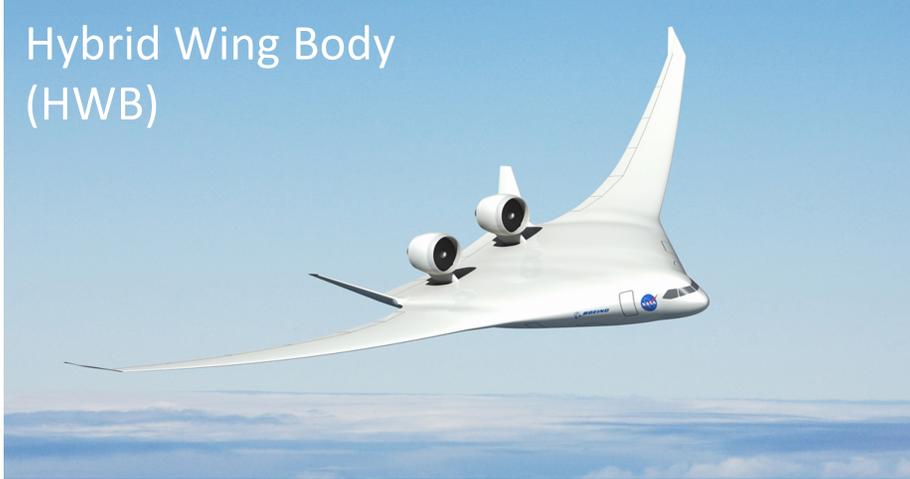
Progress in commercial transport aircraft has been driven mainly by

- Engines (became more efficient, more powerful and more reliable and even electric.)
- Composite structures and rapid manufacturing methods (became more reliable and viable)
- Control surfaces and avionics (became more sophisticated and more robust)

Engine-airframe integration, however, lagged: Thin-tube fuselages and wing-mounted engines are still the state of the art.

And most of them feature propulsors with distorted inflows

Hybrid Wing Body
(HWB)



NASA, Boeing

STARC-ABL / TCT



NASA

D8



NASA, MIT, Aurora Flight Sciences, Pratt & Whitney

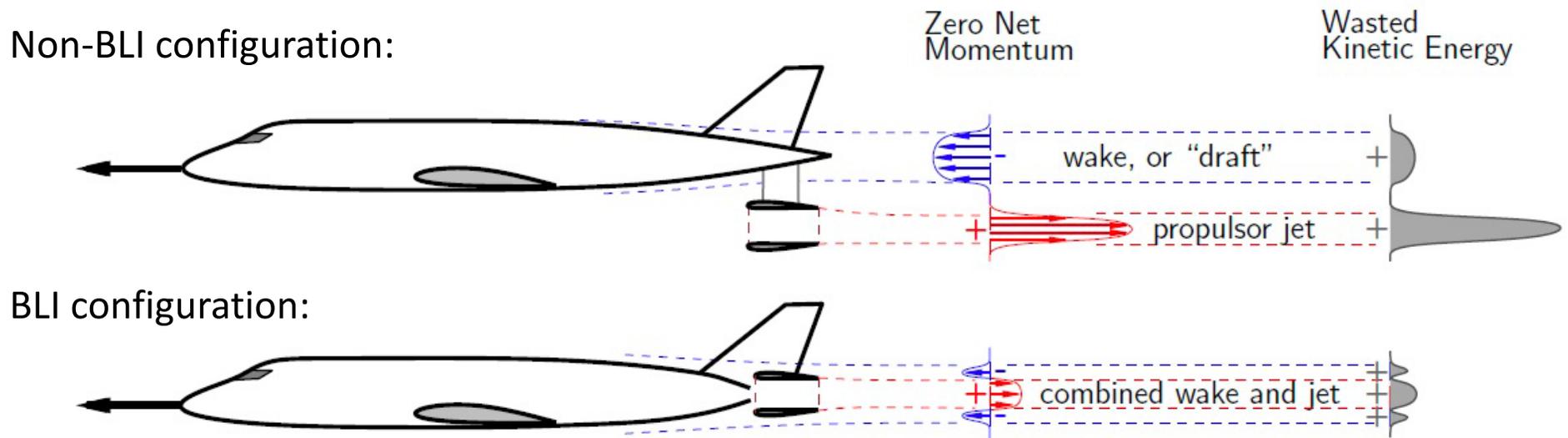
Truss-Braced Wing
(TBW)



NASA, Boeing

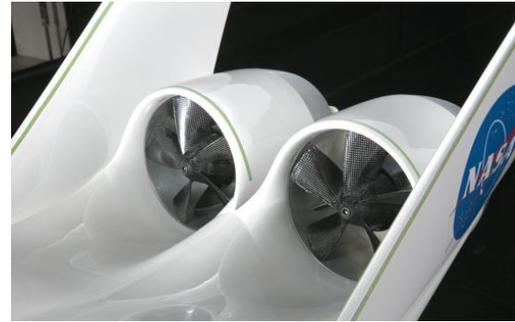
How does Boundary Layer Ingestion work?

Instead of accelerating the freestream, the propulsors accelerate the fluid that was decelerated earlier by the body. This reduces the total form drag of the aircraft.



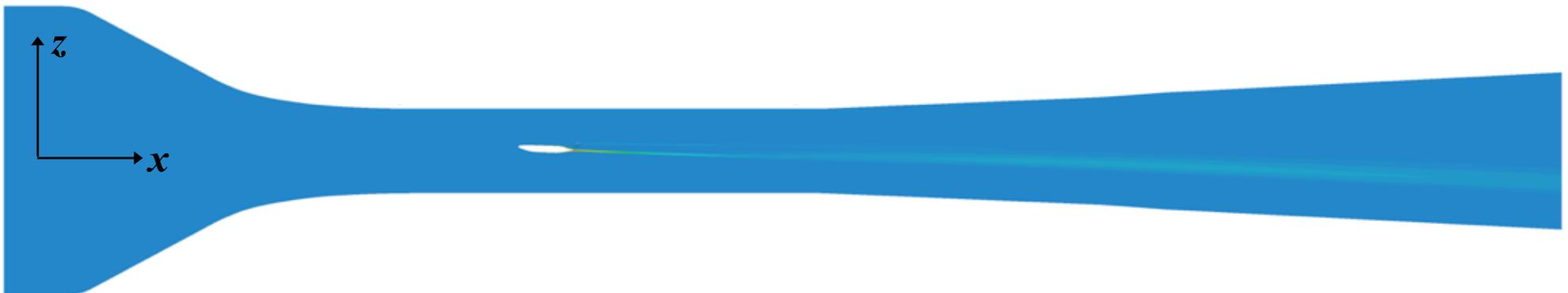
The D8 aircraft in wind tunnel

Tests 14x22ft Wind Tunnel at NASA Langley Research Center



Uranga et al., *Preliminary Experimental Assessment of the Boundary Layer Ingestion Benefit for the D8 Aircraft*, AIAA-2014-0906

CFD (Computational Fluid Dynamics) simulations of the model in the wind tunnel



Pandya, *External Aerodynamics Simulations for the MIT D8 "Double-Bubble" Aircraft Design*, 2012, ICCFD7-4304

Evidence of BLI benefit through wind tunnel tests and CFD

“Integrated”



“Podded”

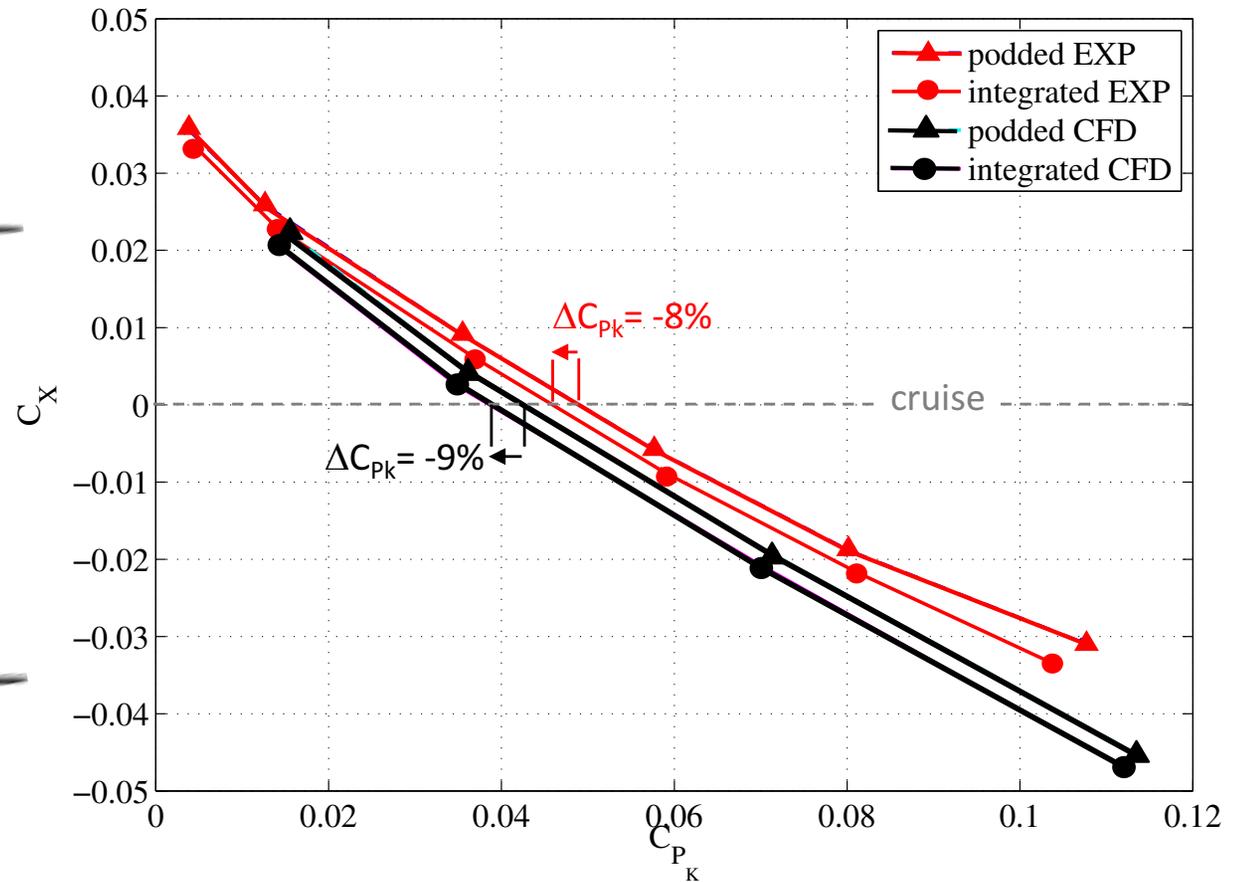


Evidence of BLI benefit through wind tunnel tests and CFD

“Integrated”



“Podded”

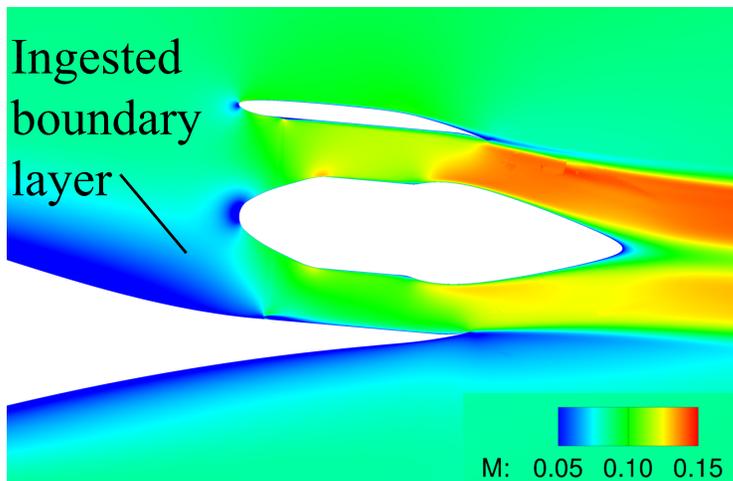
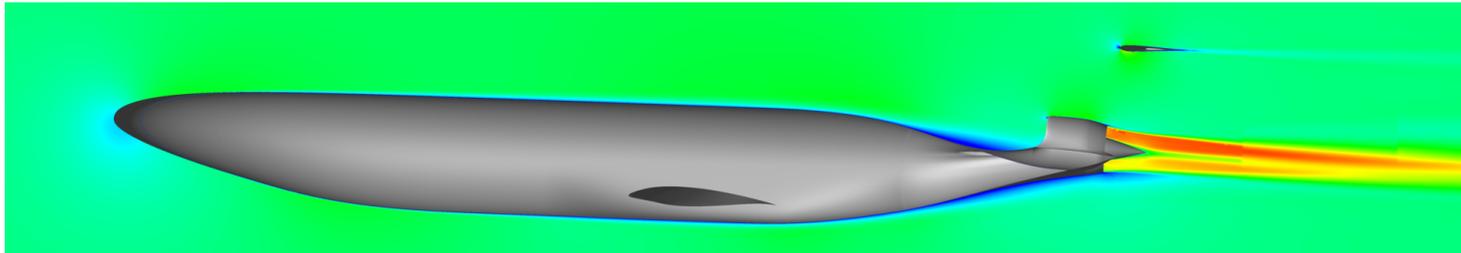


Wind tunnel tests (EXP) showed an 8% reduction in propulsive power to sustain cruise.

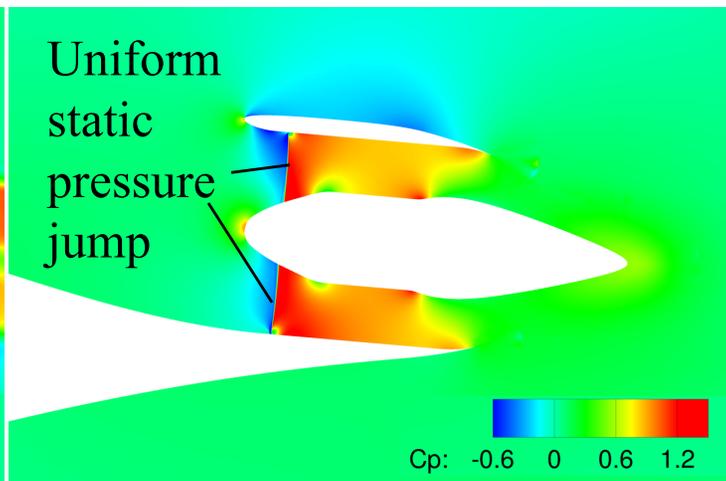
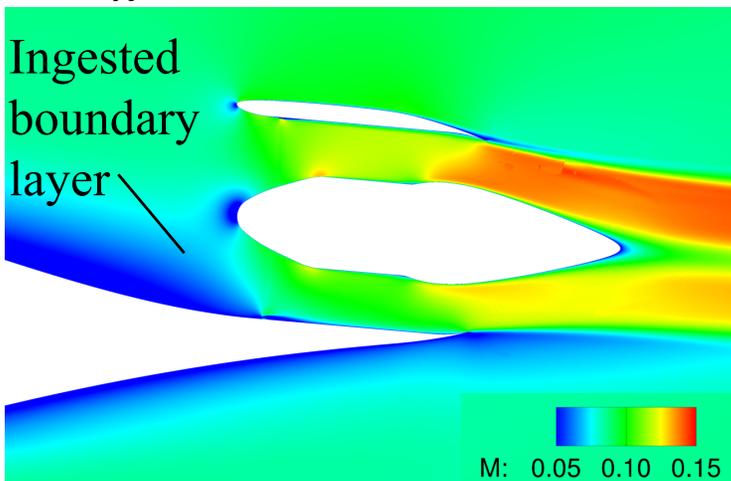
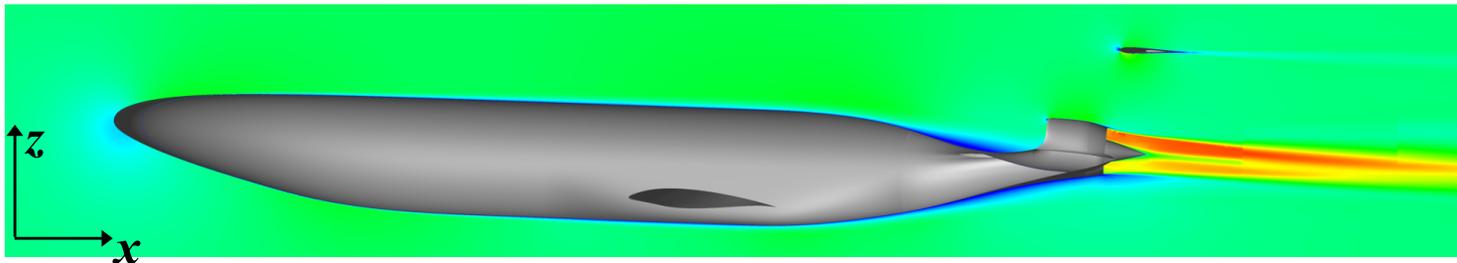
Flow simulations (CFD) predicted a 9% reduction



Issues in propulsor modeling with BLI



Issues in propulsor modeling with BLI



Previous method:
Uniform static
pressure jump

$$p_{\text{avg}} = 0.5(p_J + p_{J+1})$$

$$p_J = p_{\text{avg}} - 0.5\Delta p$$

$$p_{J+1} = p_{\text{avg}} + 0.5\Delta p$$

(No jump applied on ρ , T)

What?

Literature on Propulsor Modeling

Variants of actuator disk or blade element models

Helicopter rotors & wind turbine applications

Fejtek and Roberts [1992]

Zori and Rajagopalan [1995]

Chaffin and Berry [1997] --> Two versions are already in Overflow

O'Brien and Smith [2005]

... many others.

Literature on Propulsor Modeling

Variants of actuator disk or blade element models

Helicopter rotors & wind turbine applications

Fejtek and Roberts [1992]

Zori and Rajagopalan [1995]

Chaffin and Berry [1997] --> Two versions are already in Overflow

O'Brien and Smith [2005]

... many others.

Turbomachine applications

Joo and Hynes [1997]

Kim et al. [1999]

...

Literature on Propulsor Modeling

Variants of actuator disk or blade element models

Helicopter rotors & wind turbine applications

Fejtek and Roberts [1992]

Zori and Rajagopalan [1995]

Chaffin and Berry [1997] --> Two versions are already in Overflow

O'Brien and Smith [2005]

... many others.

Turbomachine applications

Joo and Hynes [1997]

Kim et al. [1999]

...

A particular series of “body-force” approaches for turbomachines

Marble [1964]

...

Gong et al. [1998]

Defoe and Spakovszky [2013]

Peters et al. [2014]

Hall et al. [2017]

The implemented body force model by Hall et al.

$$\nabla \cdot (\rho \mathbf{V}) = 0$$

$$\mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} = 0$$

$$\mathbf{V} \cdot \nabla h_t = 0$$

(Steady-state Euler equations)

The implemented body force model by Hall et al.

$$\nabla \cdot (\rho \mathbf{V}) = 0$$

$$\mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} = \mathbf{f}$$

$$\mathbf{V} \cdot \nabla h_t = \mathbf{V} \cdot \mathbf{f} + \dot{e}$$

The implemented body force model by Hall et al.

$$\begin{aligned}\nabla \cdot (\rho \mathbf{V}) &= 0 \\ \mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} &= \mathbf{f} \\ \mathbf{V} \cdot \nabla h_t &= \mathbf{V} \cdot \mathbf{f} + \dot{e}\end{aligned}\quad f = \frac{2\pi\delta(\frac{1}{2}|\mathbf{W}|^2)}{\frac{2\pi r}{B}|n_\theta|}$$

The implemented body force model by Hall et al.

$$\begin{aligned}\nabla \cdot (\rho \mathbf{V}) &= 0 & f &= \frac{2\pi\delta(\frac{1}{2}|\mathbf{W}|^2)}{\frac{2\pi r}{B}|n_\theta|} \\ \mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} &= \mathbf{f} \\ \mathbf{V} \cdot \nabla h_t &= \mathbf{V} \cdot \mathbf{f} + \dot{e} & \dot{e} &= T \cdot \mathbf{V} \nabla s = -\mathbf{W} \cdot \mathbf{f}\end{aligned}$$

The implemented body force model by Hall et al.

$$\nabla \cdot (\rho \mathbf{V}) = 0$$
$$\mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} = \mathbf{f}$$

$$\mathbf{f} = \frac{2\pi\delta(\frac{1}{2}|\mathbf{W}|^2)}{\frac{2\pi r}{B}|n_\theta|}$$

$$\mathbf{V} \cdot \nabla h_t = \mathbf{V} \cdot \mathbf{f} + \dot{e}$$

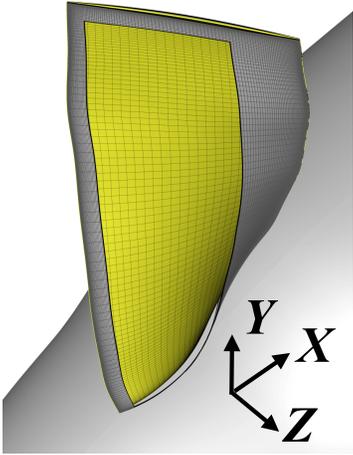
$$\dot{e} = T \cdot \mathbf{V} \nabla s = -\mathbf{W} \cdot \mathbf{f}$$

$$\mathbf{W} \cdot \mathbf{f} = 0 \text{ (with isentropic flow turning assumption)}$$

How?

Implementation of the model to CGT¹ and Overflow²

1. Import:



Import surface
definition of one
of the blades

[1] CGT: Chimera Grid Tools.

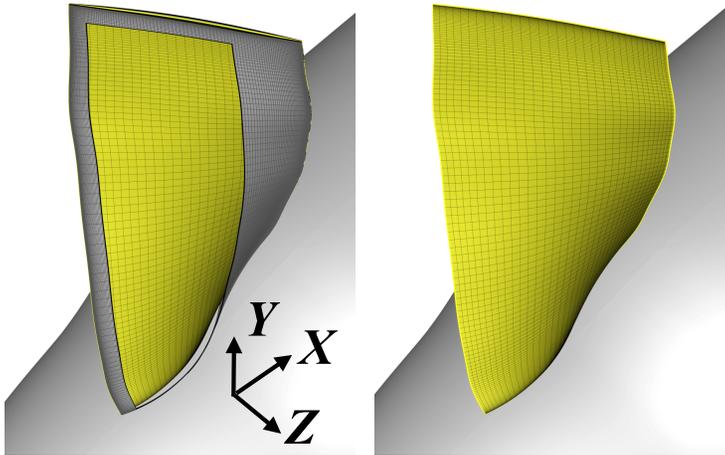
Chan, Gomez, Rogers and Buning, *Best Practices in Overset Grid Generation*, 2002, AIAA-2002-3191

[2] Nichols and Buning, *Users Manual for OVERFLOW 2.2*, <https://overflow.larc.nasa.gov>

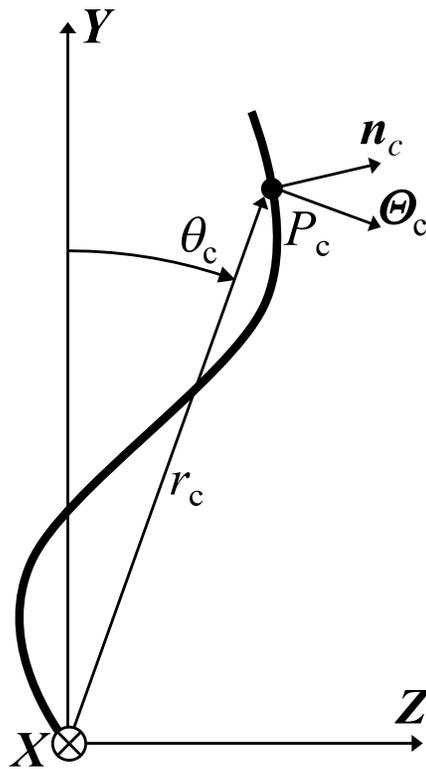
Implementation of the model to CGT and Overflow

1. Import:

2. Extract:



Extract the camber surface of the blade

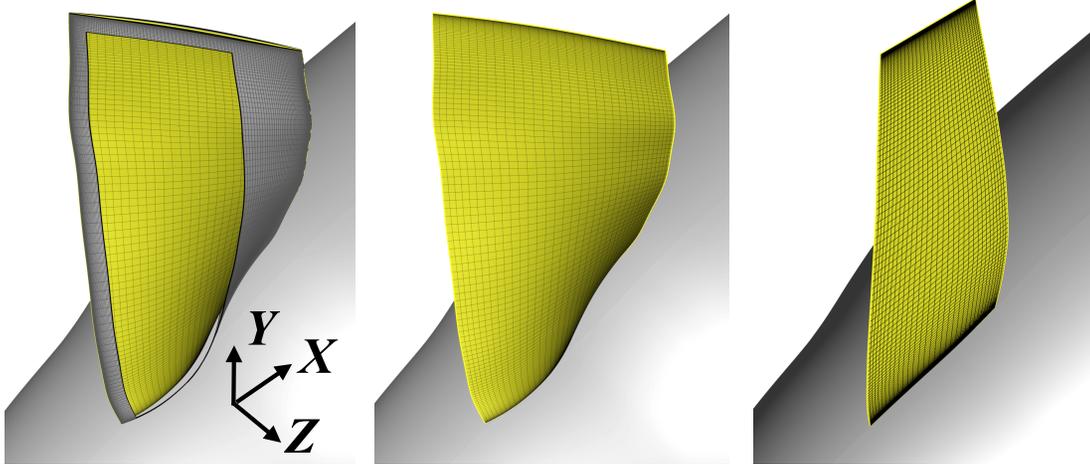


Implementation of the model to CGT and Overflow

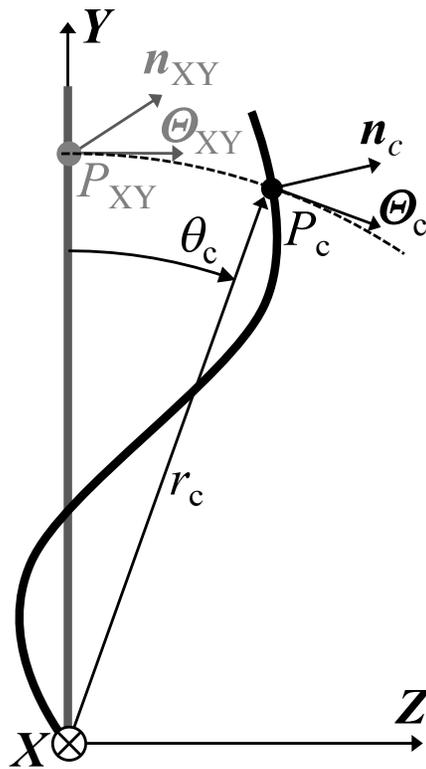
1. Import

2. Extract:

3. Flatten:

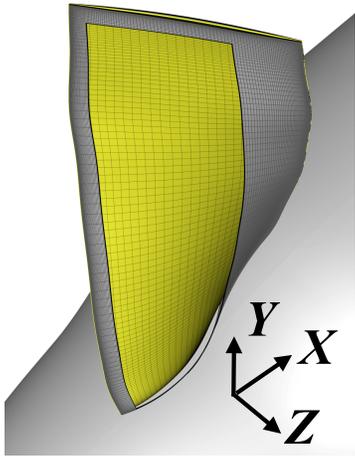


Flatten the camber surface on $Z=0$ plane

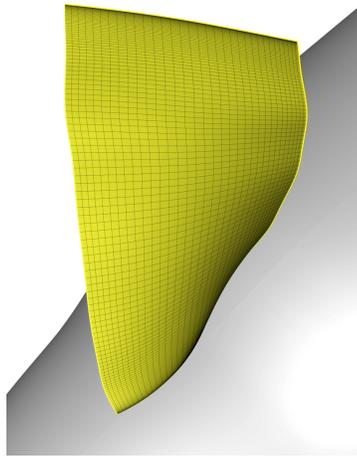


Implementation of the model to CGT and Overflow

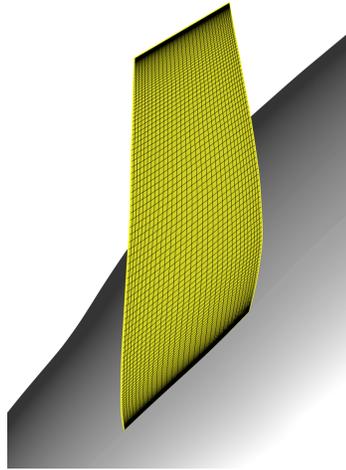
1. Import



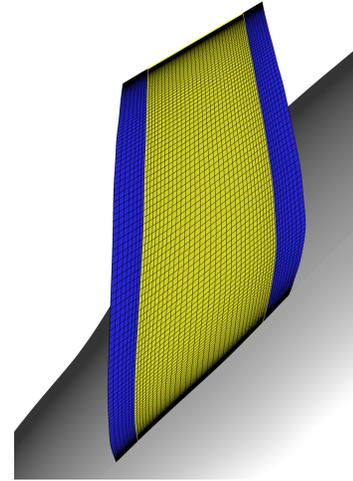
2. Extract:



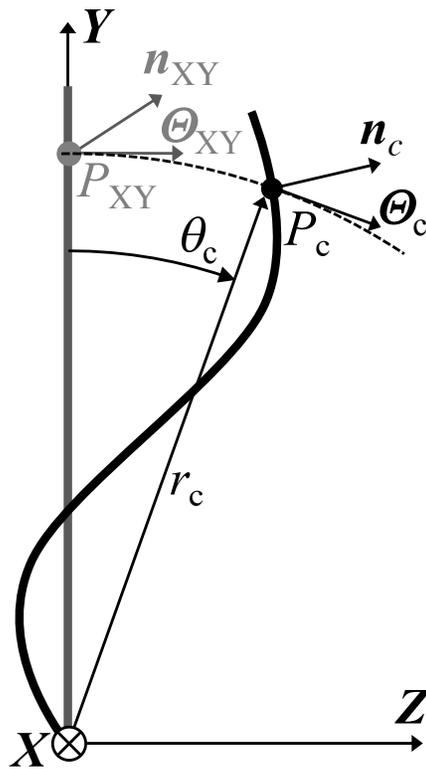
3. Flatten:



4. Extend:

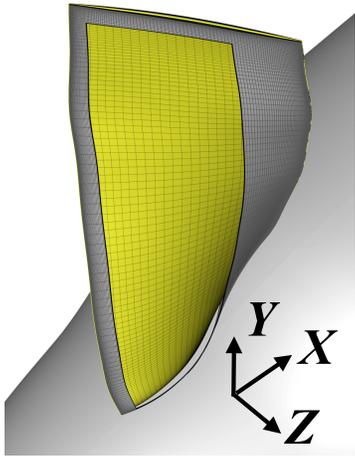


Extend for a proper overlap with neighboring grids

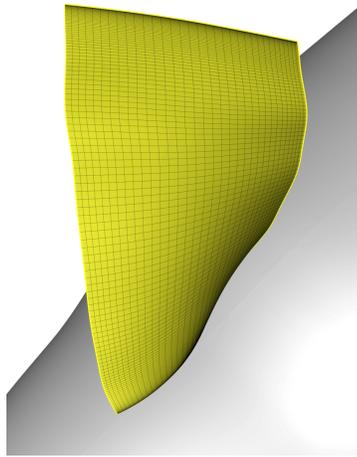


Implementation of the model to CGT and Overflow

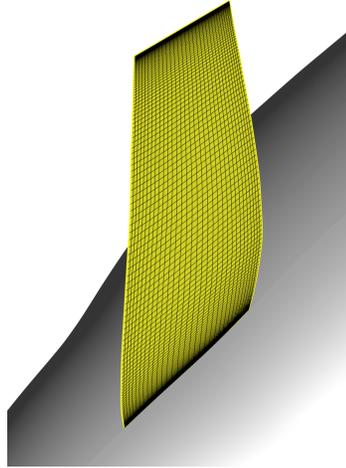
1. Import



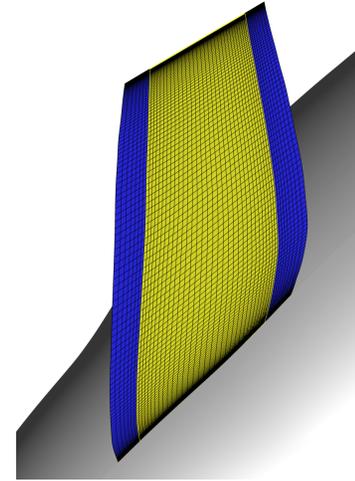
2. Extract:



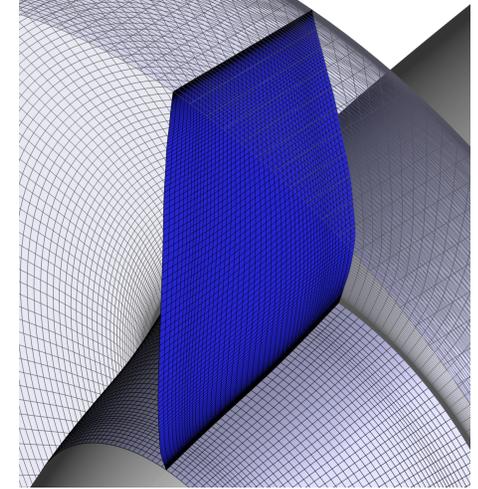
3. Flatten:



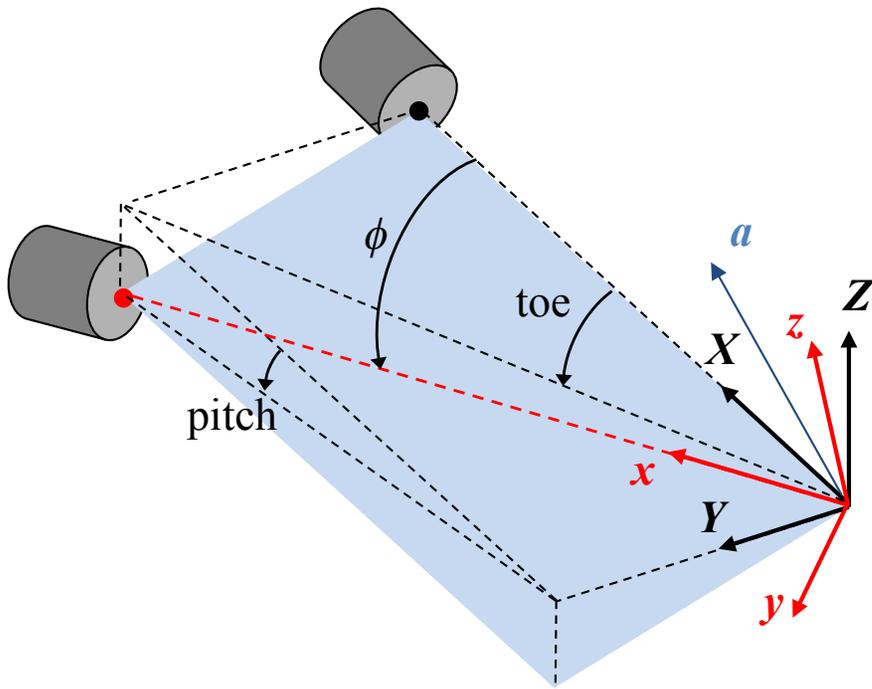
4. Extend:



5. Revolve:



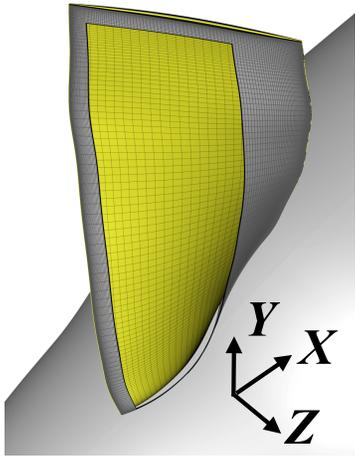
6. Rotate:



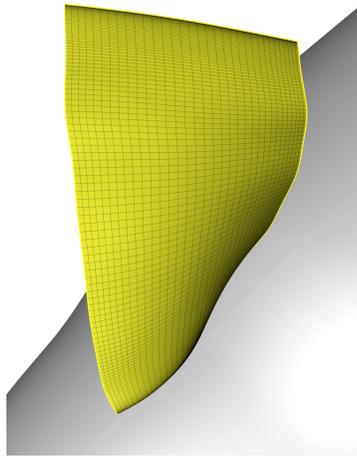
Rotate the whole grid for
pitch and toe angles

Implementation of the model to CGT and Overflow

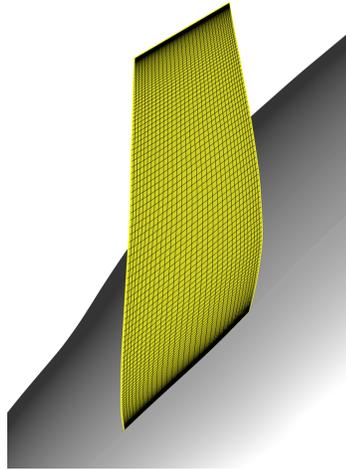
1. Import



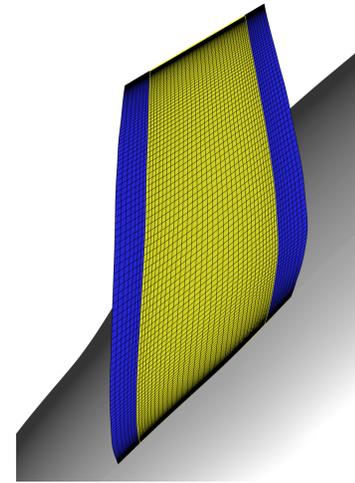
2. Extract:



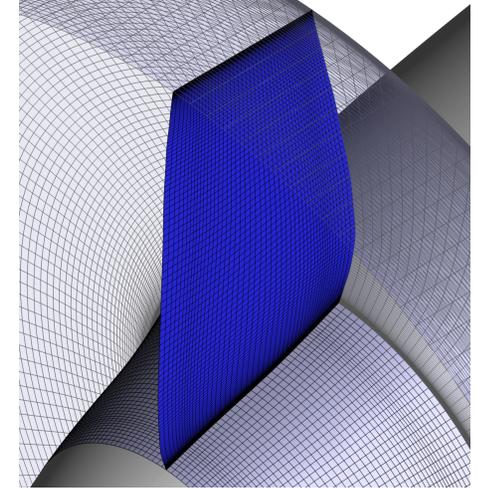
3. Flatten:



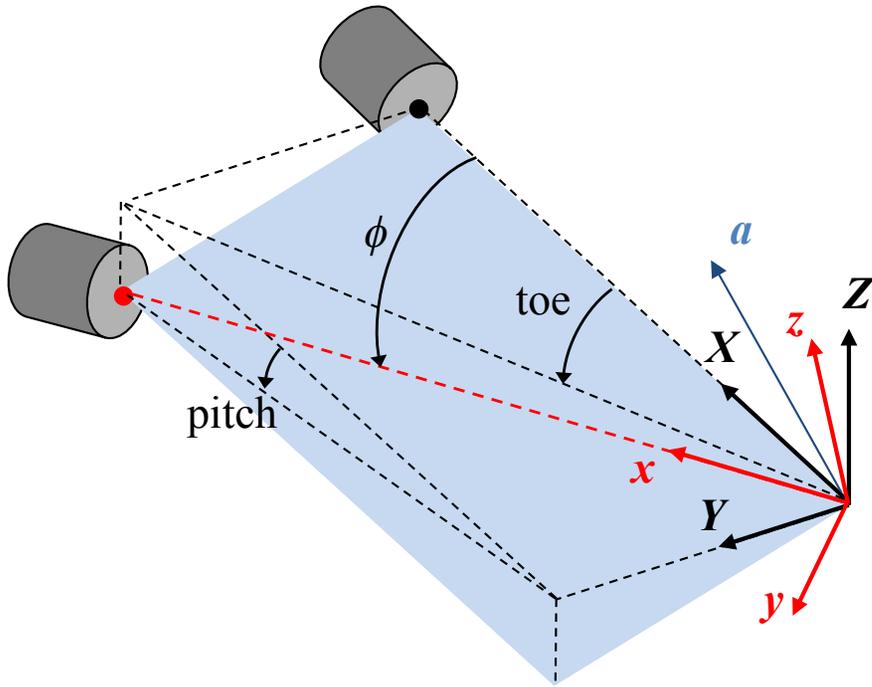
4. Extend:



5. Revolve:



6. Rotate:



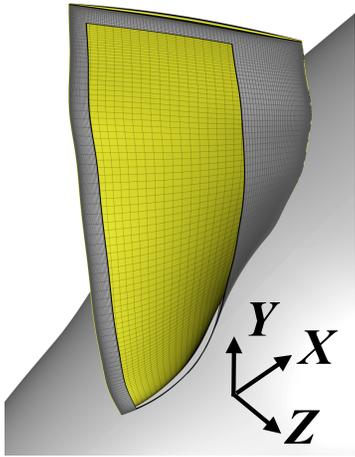
7. Save:

- r
- n_x
- n_y
- n_z
- n_θ
- Θ_x
- Θ_y
- Θ_z

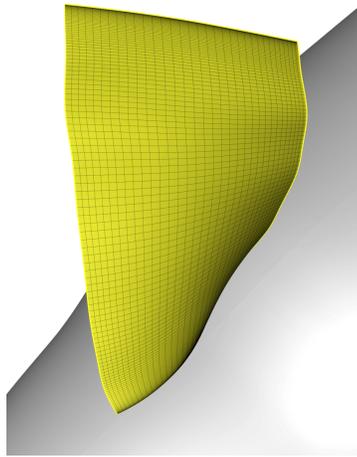
Save the camber surface orientation metrics in a file

Implementation of the model to CGT and Overflow

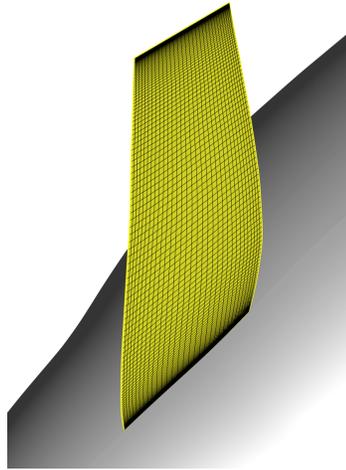
1. Import



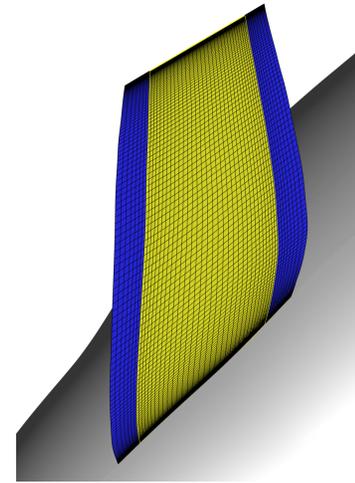
2. Extract:



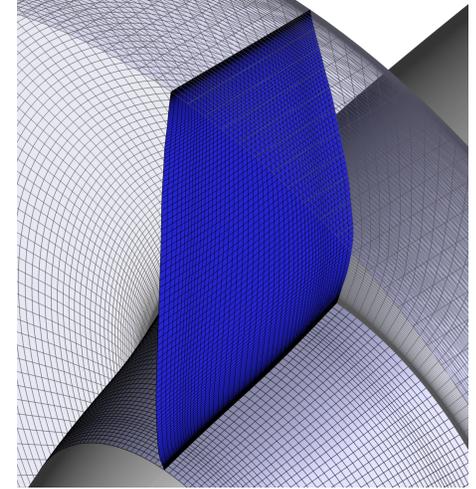
3. Flatten:



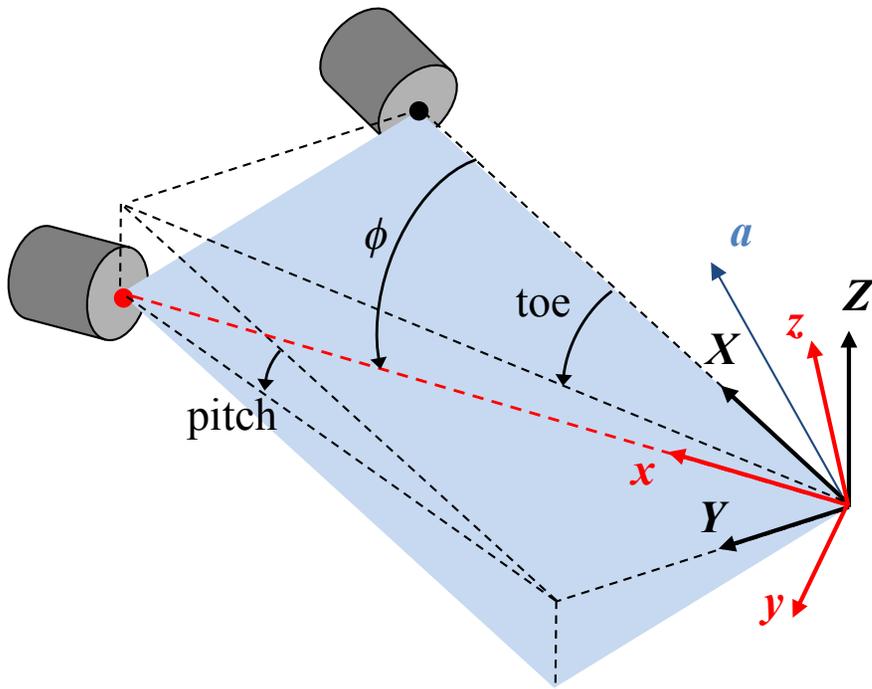
4. Extend:



5. Revolve:



6. Rotate:



7. Save:

- r
- n_x
- n_y
- n_z
- n_θ
- Θ_x
- Θ_y
- Θ_z

8. Read in the solver, compute the source terms at each iteration

$$f = \frac{2\pi\delta(\frac{1}{2}|\mathbf{W}|^2)}{\frac{2\pi r}{B}|n_\theta|}$$

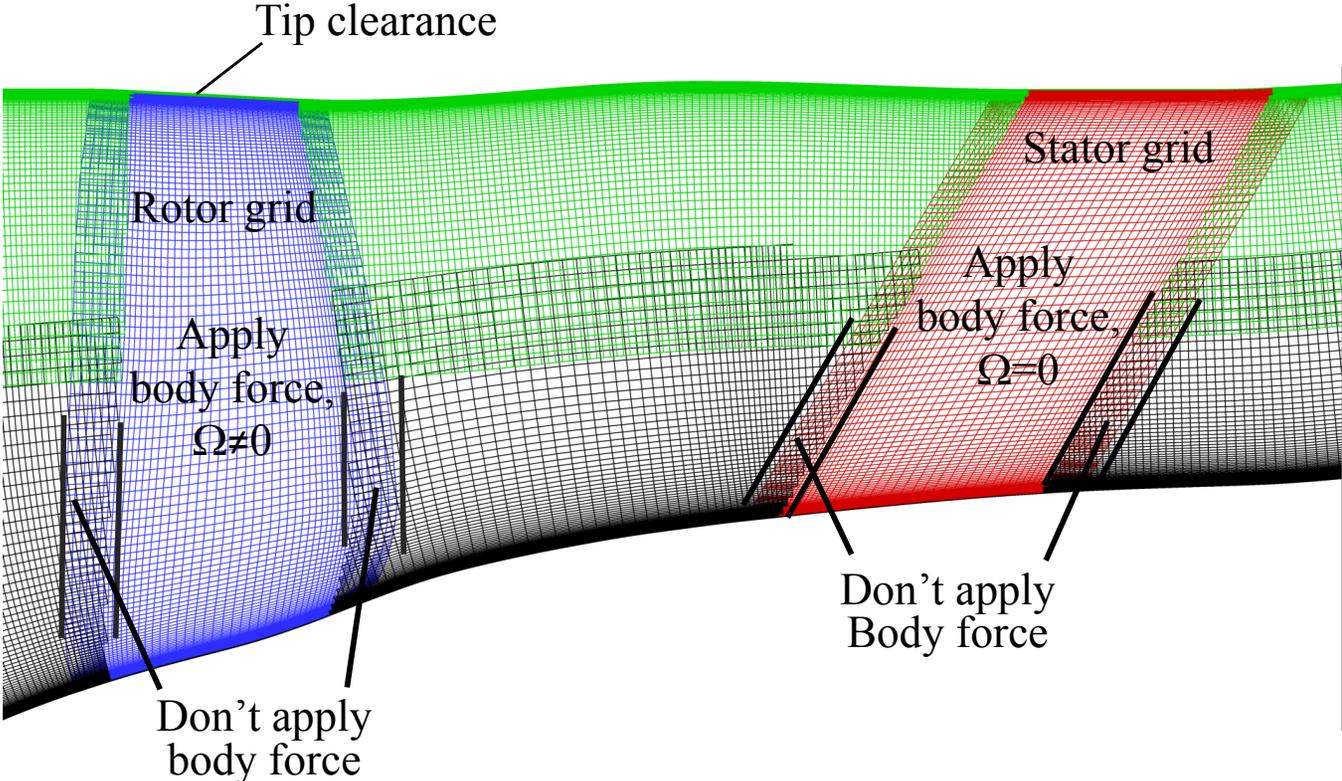
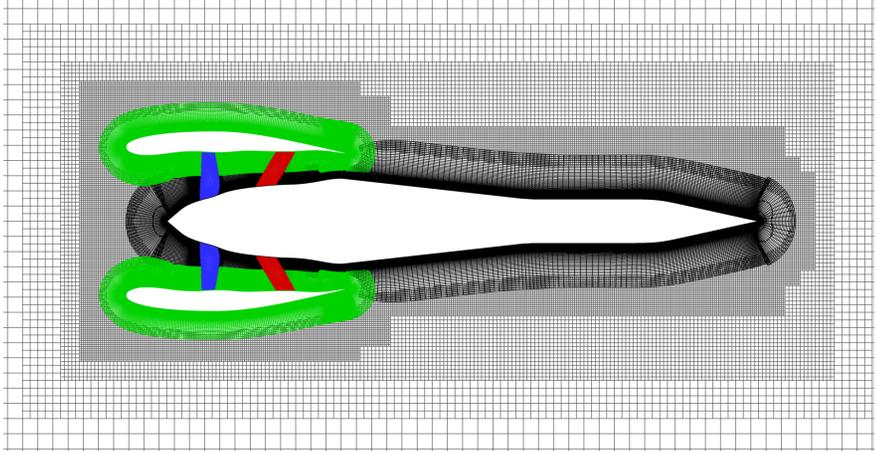
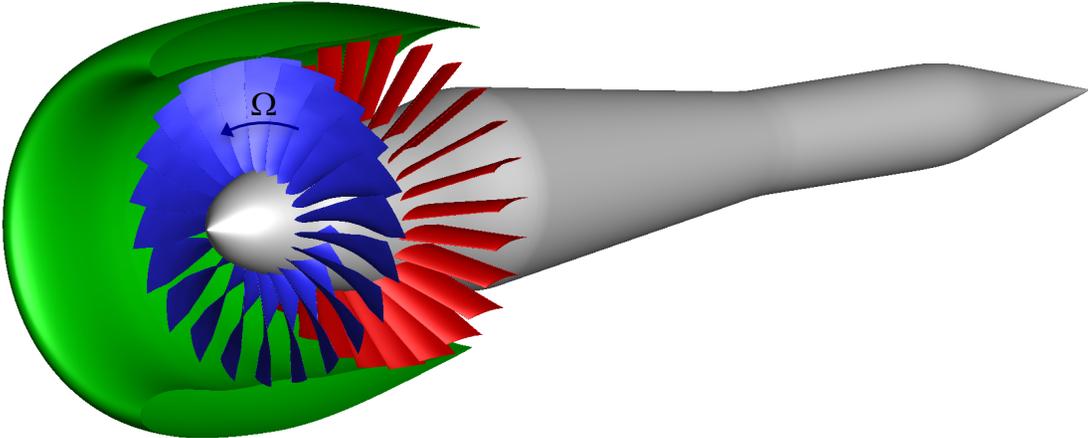
$$\nabla \cdot (\rho \mathbf{V}) = 0$$

$$\mathbf{V} \cdot \nabla \mathbf{V} + \frac{\nabla p}{\rho} = \mathbf{f}$$

$$\mathbf{V} \cdot \nabla h_t = \mathbf{V} \cdot \mathbf{f} + \dot{e}$$

Now NS equations
(not Euler equations)

Implementation of the model to CGT and Overflow



Step-by-step implementation instructions in the paper

2. At each vertex $P_c(X, Y, Z)$ on the camber surface grid, compute the axisymmetric coordinates $P_c(X, r, \theta)$:

$$r_c = \sqrt{Y_c^2 + Z_c^2} \quad (11a)$$

$$\theta_c = \arctan(Z_c/Y_c) \quad (11b)$$

3. At each vertex P_c on the camber surface grid find the surface normal $\mathbf{n}_c = \langle n_{c,x}, n_{c,y}, n_{c,z} \rangle$ and compute (Figure 3f):

$$n_{c,r} = (X_c n_{c,y} + Z_c n_{c,z})/r_c \quad (12a)$$

$$n_{c,\theta} = (-Z_c/n_{c,y} + Y_c n_{c,z})/r_c \quad (12b)$$

The surface normals of the camber surface should point in the positive rotation direction around the X -axis of the propulsor; this can be accomplished by multiplying \mathbf{n}_c by -1 if necessary. In this implementation, we computed the surface normals using a routine readily available in CGT.

4. Flatten the camber surface representation on XY plane such that each vertex $P_c(X, Y, Z)$ on the camber surface corresponds to a point $P_{XY}(x, y = r, 0)$ on the flattened camber surface representation (Figure 3c). Accordingly, rotate the actual camber surface normals (n_c) to find the corresponding camber surface normals on the flattened camber surface grid (n_{XY}):

$$n_{XY,X} = n_{c,X} \quad (13a)$$

$$n_{XY,Y} = n_{c,r} \quad (13b)$$

$$n_{XY,Z} = n_{c,\theta} \quad (13c)$$

5. Extend the flattened grid upstream and downstream by a few (e.g. 4 to 10) grid lines to ensure proper overlap with neighboring grids (Figure 3d).
6. Fully revolve the flattened camber grid around the X -axis to generate a structured volume grid of revolution by computing

$$X = X_c \quad (14a)$$

$$Y = r_c \cos(K\Delta\theta) \quad (14b)$$

$$Z = r_c \sin(K\Delta\theta) \quad (14c)$$

for each point $P(X, Y, Z)$ in the rotor grid zone (Figure 3e). Here $K = 1, 2, 3, \dots, K_{\max}$ is the index number of the point in the periodic direction of the grid and $\Delta\theta = 2\pi/(K_{\max} - 1)$. In the grid system of this work, we used $K_{\max} = 181$ ($K = 1$ plane overlapping with $K = 181$ plane) which resulted in $\Delta\theta = 2^\circ$.

Additionally, revolve the camber surface orientation metrics for the subzone where the body force model will be applied (Figure 3f).

$$n_X = n_{c,X} \quad (15a)$$

$$n_Y = n_{c,r} \cos(K\Delta\theta) - n_{c,\theta} \sin(K\Delta\theta) \quad (15b)$$

$$n_Z = n_{c,r} \sin(K\Delta\theta) + n_{c,\theta} \cos(K\Delta\theta) \quad (15c)$$

$$n_\theta = n_{c,\theta} \quad (15d)$$

$$\Theta_X = 0 \quad (15e)$$

$$\Theta_Y = 0 \quad (15f)$$

$$\Theta_Z = 1 \quad (15g)$$

where $\Theta_X, \Theta_Y, \Theta_Z$ are the components of Θ the unit vector in the direction of the tangential velocity of the rotor blade.

The body force model is applied to the subzone ($J_b : -J_e, 1 : K_{\max}, L_b : -L_e$) where J_b and J_e are the number of added grid lines to extend the grid in upstream and downstream directions in Step 5. L_b and L_e are the number of grid lines retracted from the hub and casing surfaces, respectively, to

constrain the application of the body force model in radial direction. In case of the stator grid zone, a radial retraction of at least one grid line is necessary to avoid division-by-zero error while computing the source terms at $L = 1$ and $L = L_{\max}$ since W is zero due to static wall boundary conditions applied at the hub and the casing surfaces. In case of the rotor grid zone, no body force zone retraction is necessary at tip ($L_e = 1$) if there is already a physical gap between the rotor tip and the casing. In addition, no body force retraction is necessary at the hub ($L_b = 1$) since W is never zero due to rotation.

7. Finally, if the axis of revolution (X -axis) of the installed propulsor grid is not aligned with the x -axis of the solution domain (i.e. non-zero toe and/or pitch angles), then rotate the propulsor grids as well as the source term orientation metrics accordingly (Figure 3g). In this implementation we expressed the orientation of the propulsor as a rotation by an angle ϕ around a unit orientation vector $\mathbf{a} = \langle a_x, a_y, a_z \rangle$ and used Rodrigues' Rotation Formula²² to rotate the orientation metrics:

$$\mathbf{n}_{xyz} = \mathbf{R} \mathbf{n}_{XYZ}^T \quad (16a)$$

$$\Theta_{xyz} = \mathbf{R} \Theta_{XYZ}^T \quad (16b)$$

where

$$\mathbf{R} = \mathbf{I} + \sin(\phi)\mathbf{A} + (1 - \cos(\phi))\mathbf{A}^2 \quad (17)$$

and where

$$\mathbf{A} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \quad (18)$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (19)$$

8. Write $r, n_x, n_y, n_z, n_\theta, \Theta_x, \Theta_y, \Theta_z$ in a file to be read by Overflow for computation of the source terms during the flow solution.

The extraction, flattening, revolution and rotation of the camber surface and computation of its orientation metrics are programmed in executable codes that are called from CGT grid scripts with appropriate user inputs.

A set of subroutines added to the Overflow code reads the camber surface orientation metrics files for the rotor and the stator grids and computes the source terms at each body force grid point at each flow solver iteration as follows:

1. Compute the components of relative velocity $\mathbf{W} = \mathbf{U} - \Omega r \Theta$ in the Cartesian frame of the flow simulation domain:

$$W_x = U_x - \Omega r \Theta_x \quad (20a)$$

$$W_y = U_y - \Omega r \Theta_y \quad (20b)$$

$$W_z = U_z - \Omega r \Theta_z \quad (20c)$$

where U_x, U_y, U_z and W_x, W_y, W_z are the components of $\Theta, \mathbf{U}, \mathbf{W}$ in the cartesian frame of reference of the flow solution domain.

2. Compute the normal component of \mathbf{W}

$$W_n = \mathbf{W} \cdot \mathbf{n} = W_x n_x + W_y n_y + W_z n_z \quad (21)$$

Note that W_n can have a positive or negative value.

3. Compute the local deviation angle:

$$\delta = \arcsin(W_n/|W|) \quad (22)$$

where $|W| = \sqrt{W_x^2 + W_y^2 + W_z^2}$.

The Tools and Methods

Grid Generation: *Chimera Grid Tools (CGT)*

The steps 1 to 7 are automated by routines that are added to CGT codebase

Solver: *Overflow 2.2/*

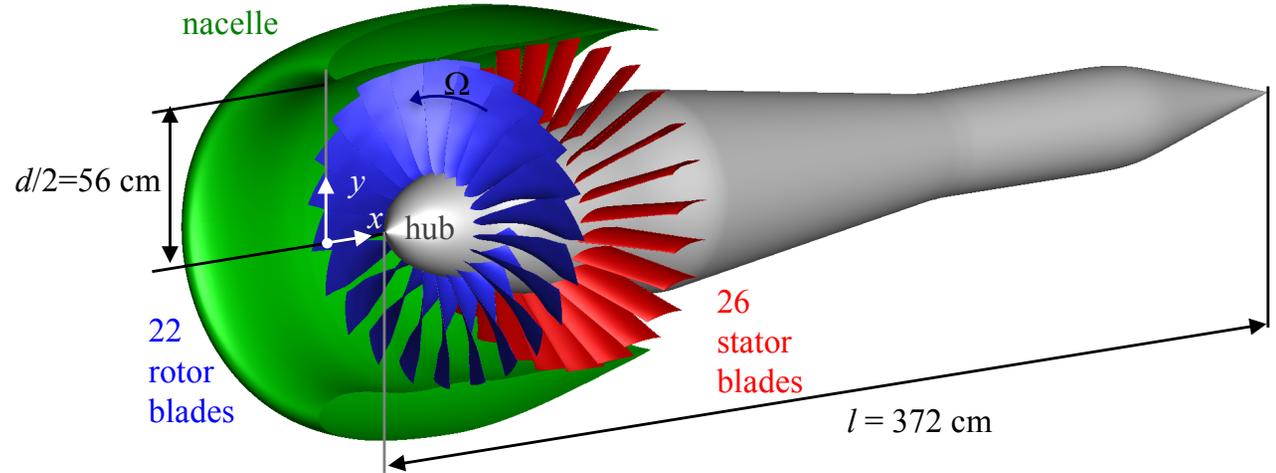
An implicit RANS solver for body-fitted structured overset grid systems

Simulations here used

- Diagonalized approximate factorization scheme [Pulliam and Chaussee 1981]
- Central difference in Euler terms
- Steady-state simulations with a constant CFL number
- Matrix dissipation
- Spalart Allmaras (SA) turbulence model (SA-noft2 implementation in Overflow)
- Body force method grids and metric files are automatically split
- Currently no multigrid on the grids that use the body force model
- Jacobians of source terms are not currently added to left hand side
(Hence no low Mach preconditioning when the body force model is used)

Validation?

Source Diagnostics Test (SDT) fan with R4 Rotors

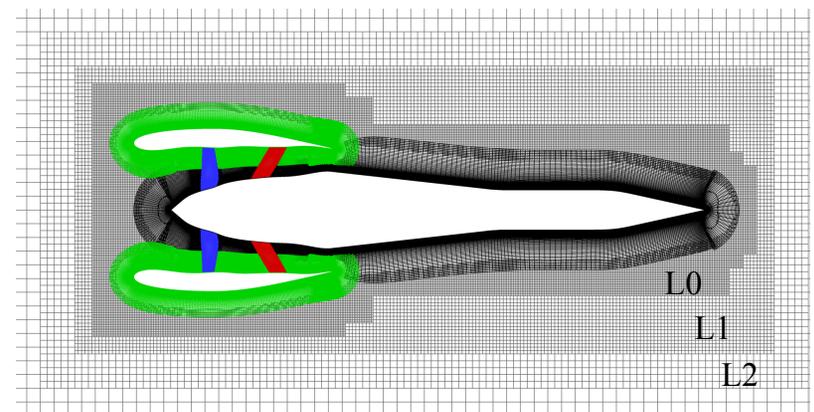
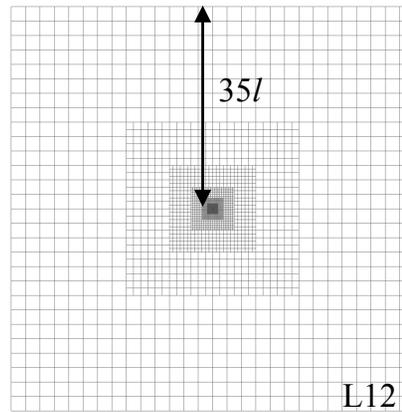
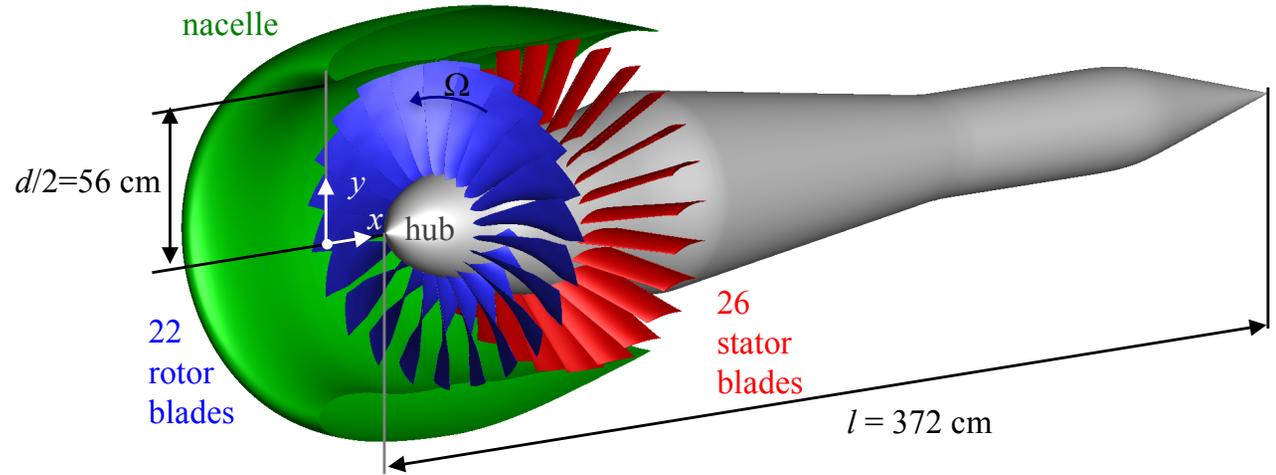


Envia, E., "Fan Noise Source Diagnostic Test Completed and Documented,"
NASA Tech. Memo. TM-2003-211990

Source Diagnostics Test (SDT) fan with R4 Rotors

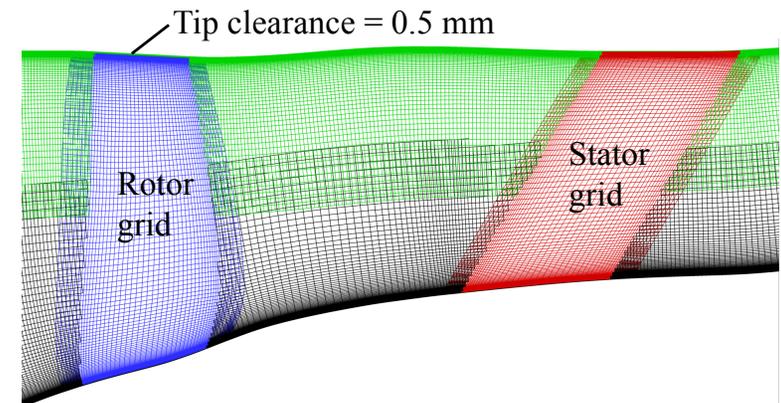
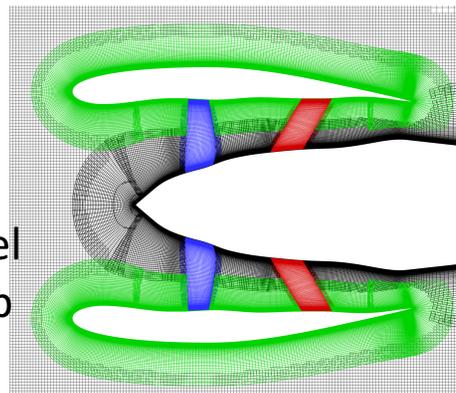


Envia, E., "Fan Noise Source Diagnostic Test Completed and Documented,"
NASA Tech. Memo. TM-2003-211990

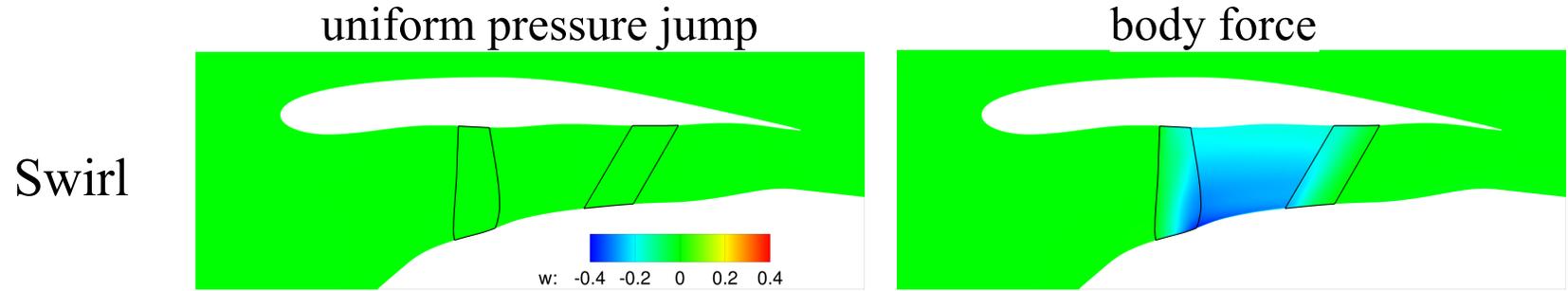


35 million vertices, $y^+ \approx 1$
4 to 8 hours on 128 Haswell cores

Full convergence with body force model
Partial convergence with pressure jump



SDT fan results

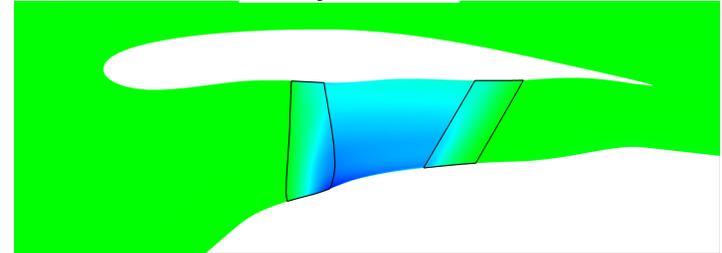
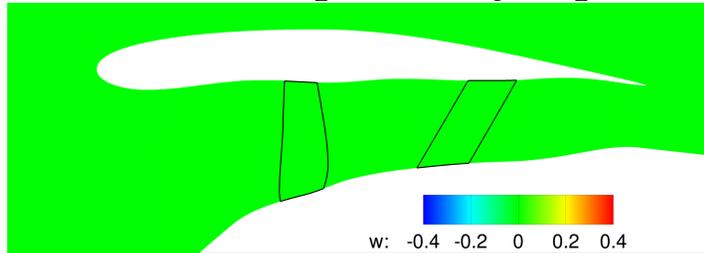


SDT fan results

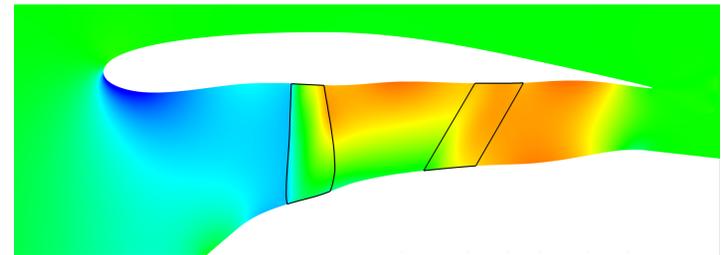
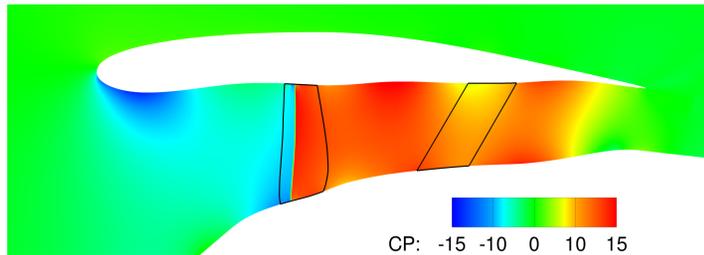
uniform pressure jump

body force

Swirl



Static Pressure

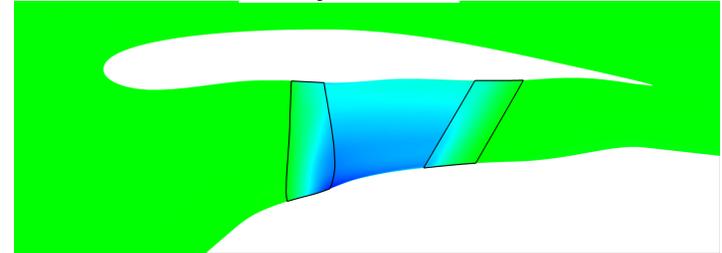
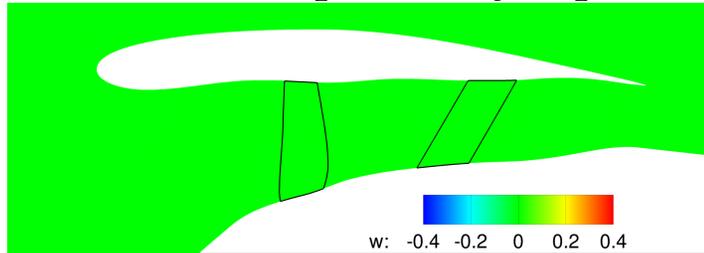


SDT fan results

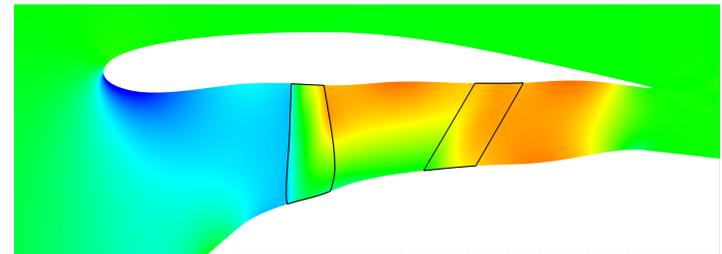
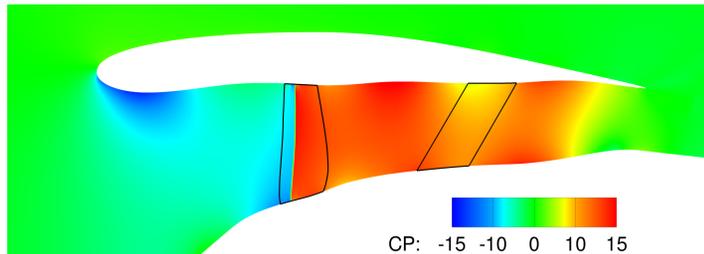
uniform pressure jump

body force

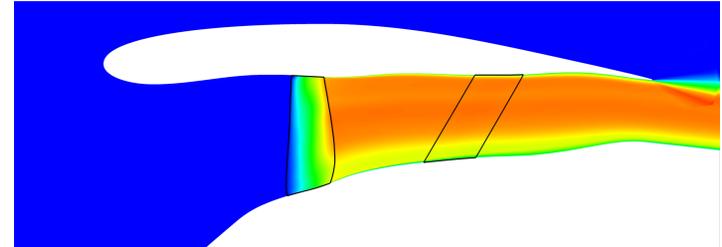
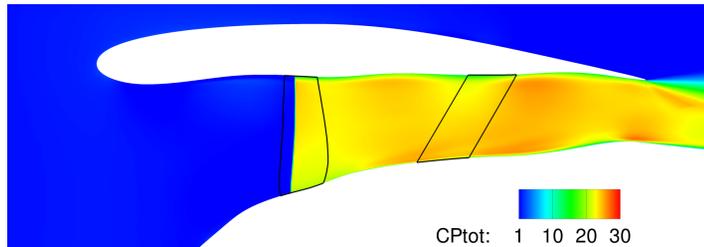
Swirl



Static Pressure



Total Pressure

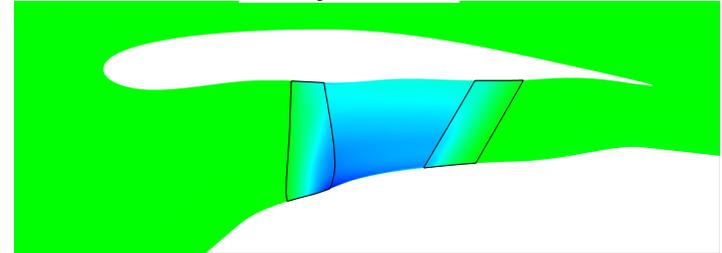
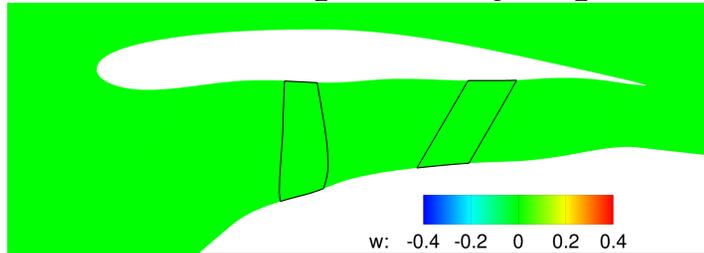


SDT fan results

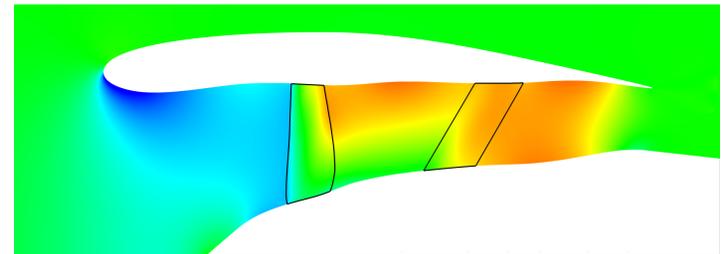
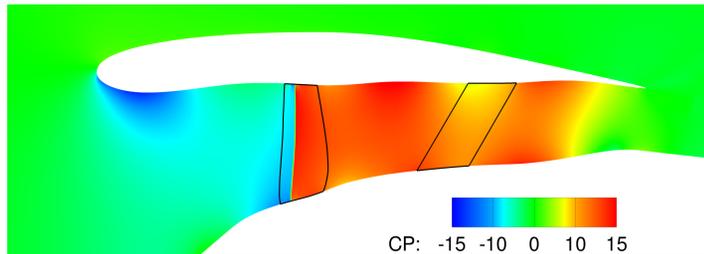
uniform pressure jump

body force

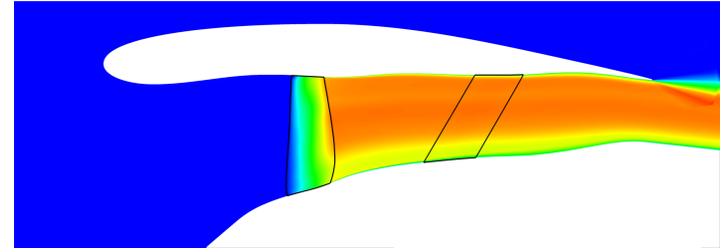
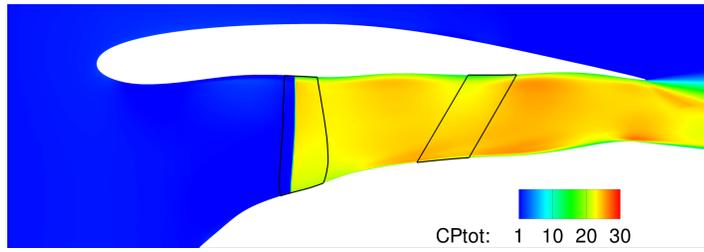
Swirl



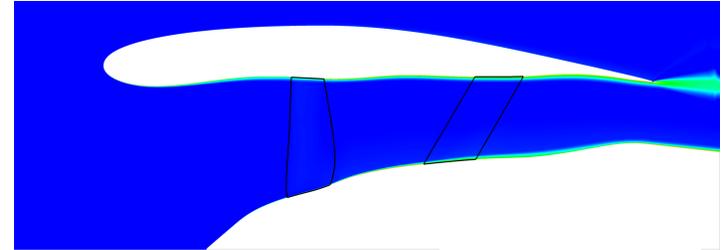
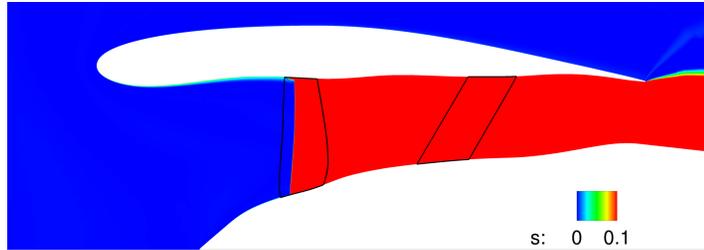
Static Pressure



Total Pressure



Entropy

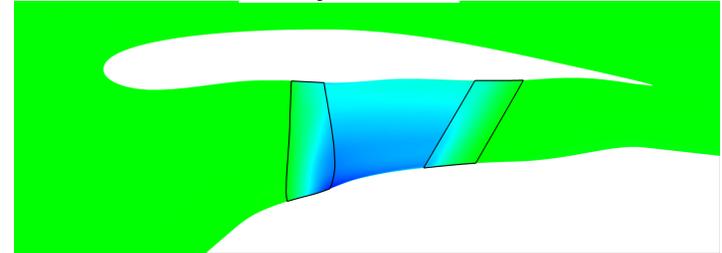
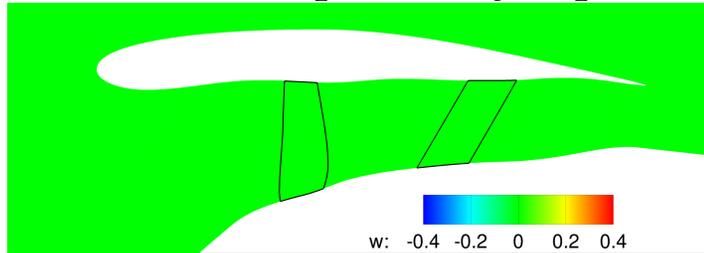


SDT fan results

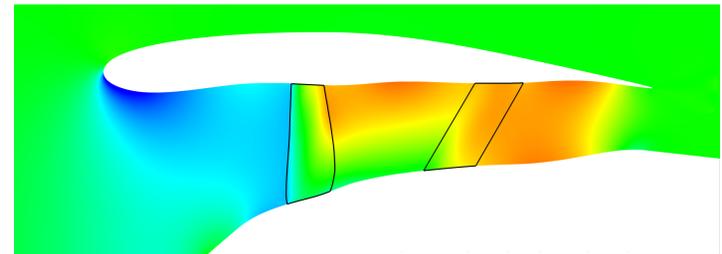
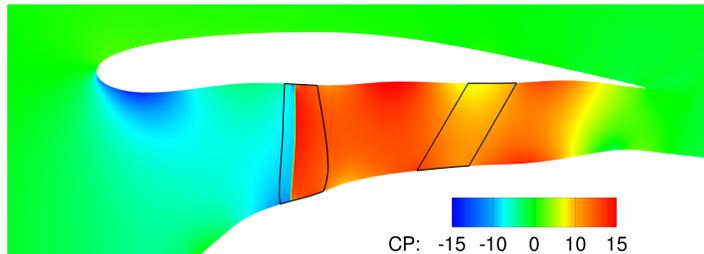
uniform pressure jump

body force

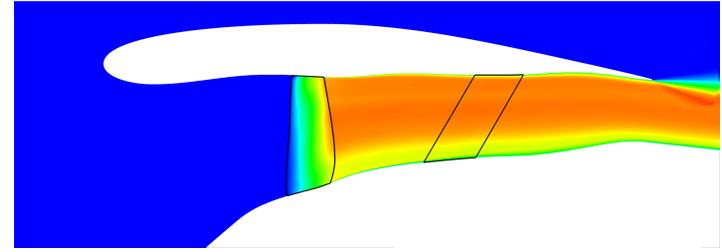
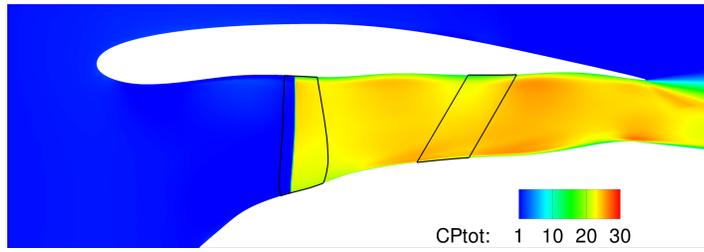
Swirl



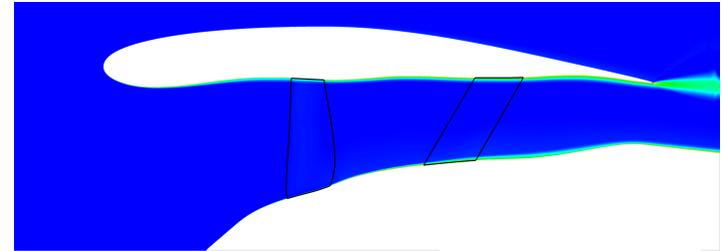
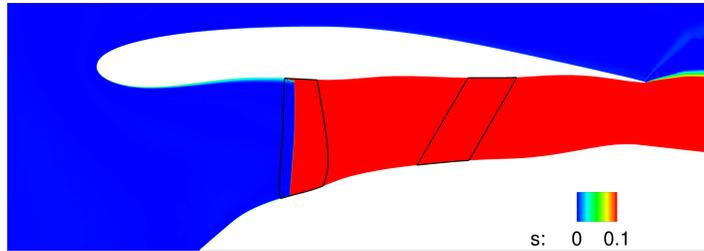
Static Pressure



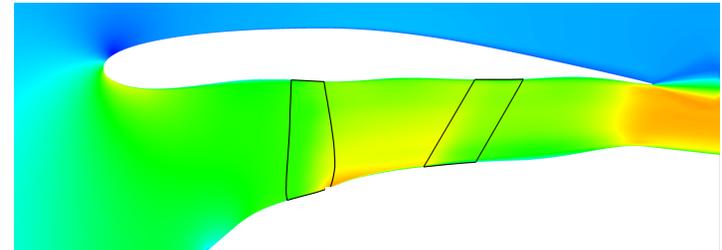
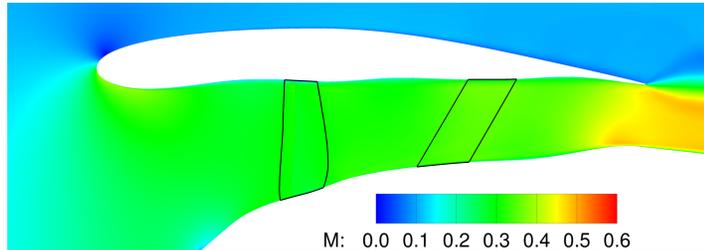
Total Pressure



Entropy



Mach number

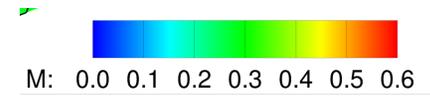
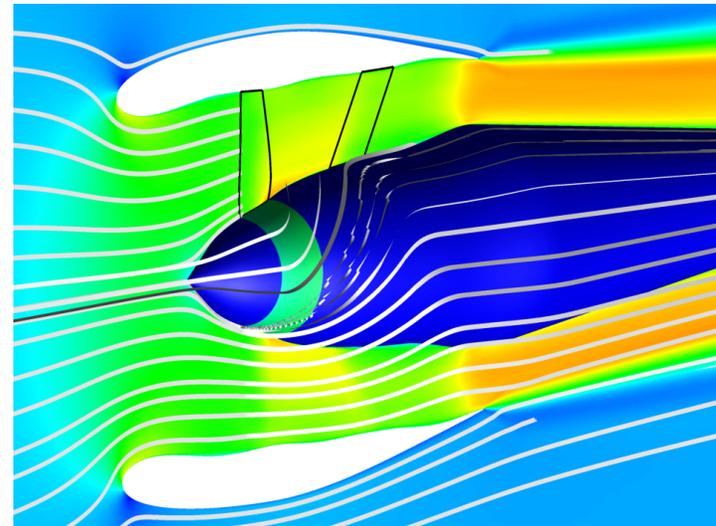
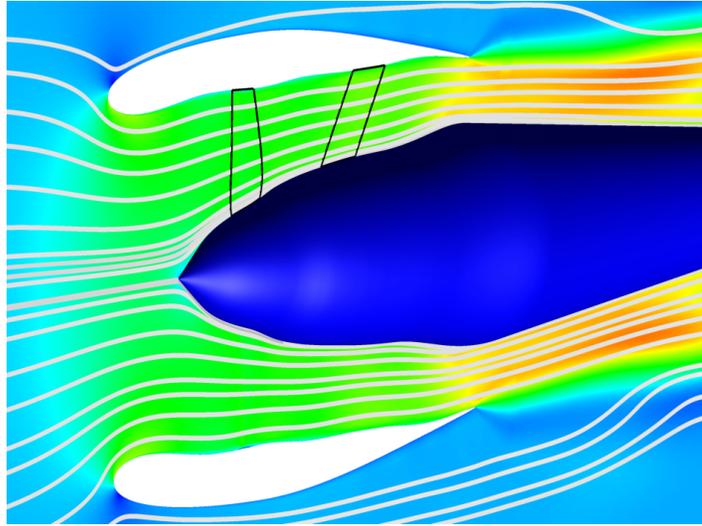


SDT fan results

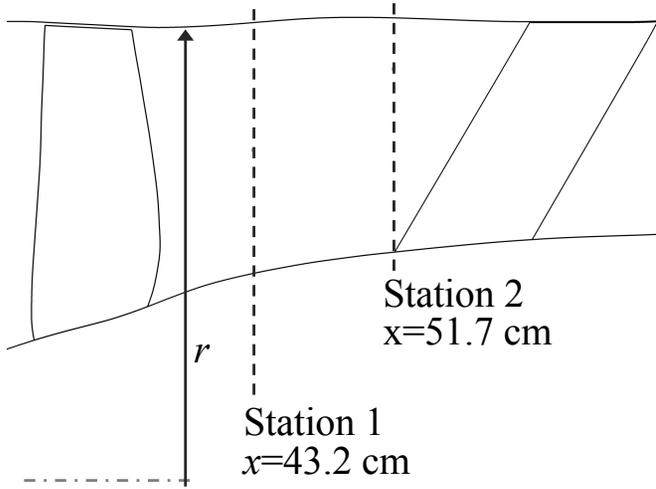
uniform pressure jump

body force

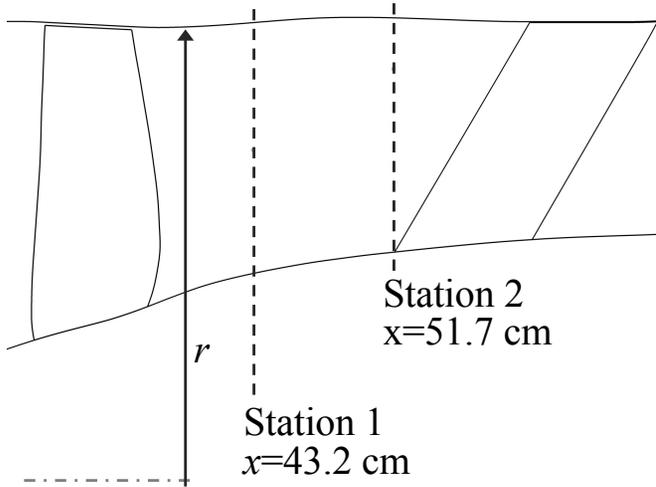
Streamlines



SDT fan results



SDT fan results

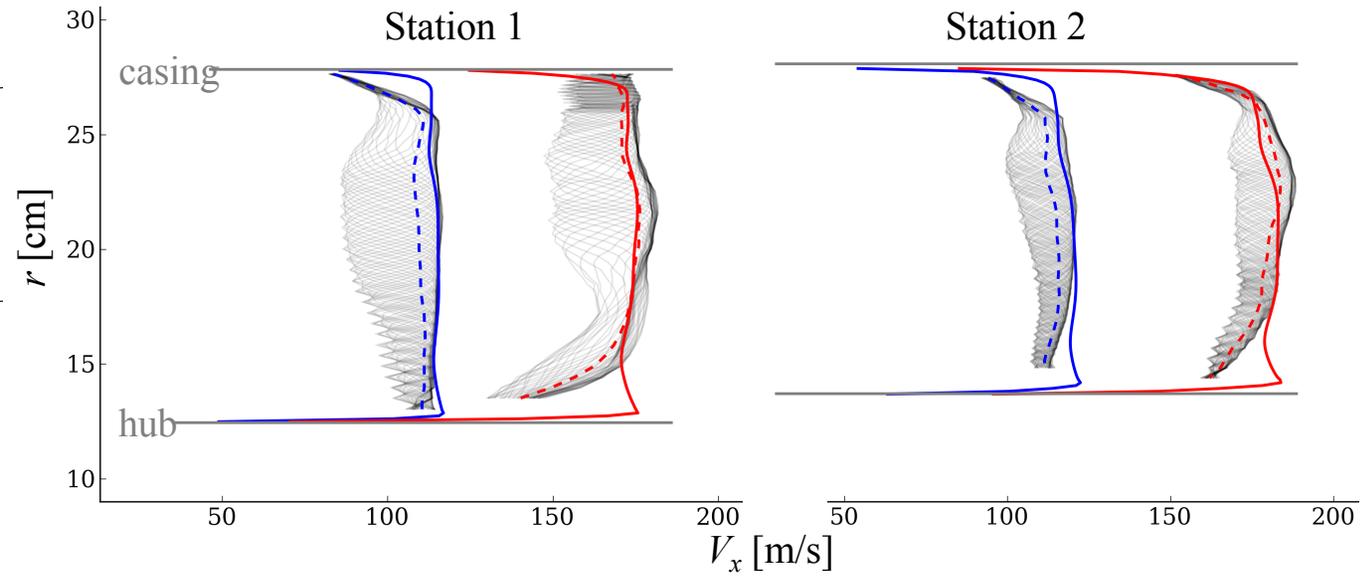
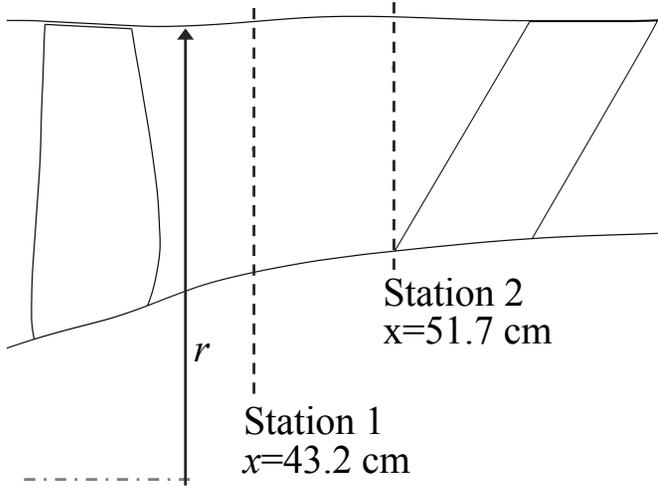


7,808 rpm 12,657 rpm

- | | | |
|-------|-------|------------------------------------|
| — | — | Experiment
(phase-avg.) |
| - - - | - - - | Experiment
(mean of phase-avg.) |
| — | — | Simulation
(body force model) |

} SDT campaign at NASA Glenn Research Center
POC: Dr. Ed Envia

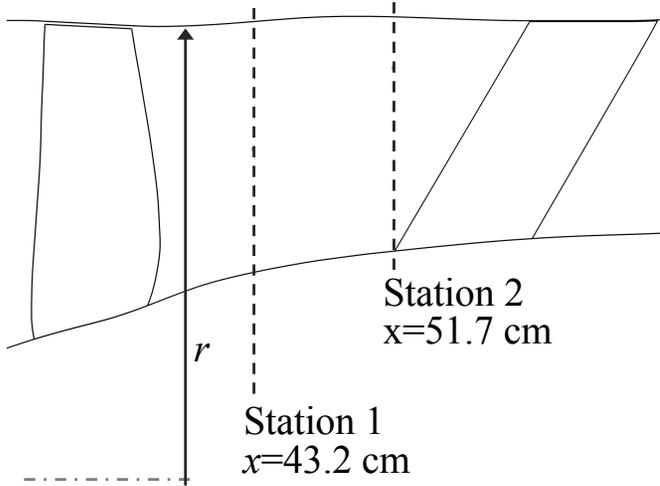
SDT fan results



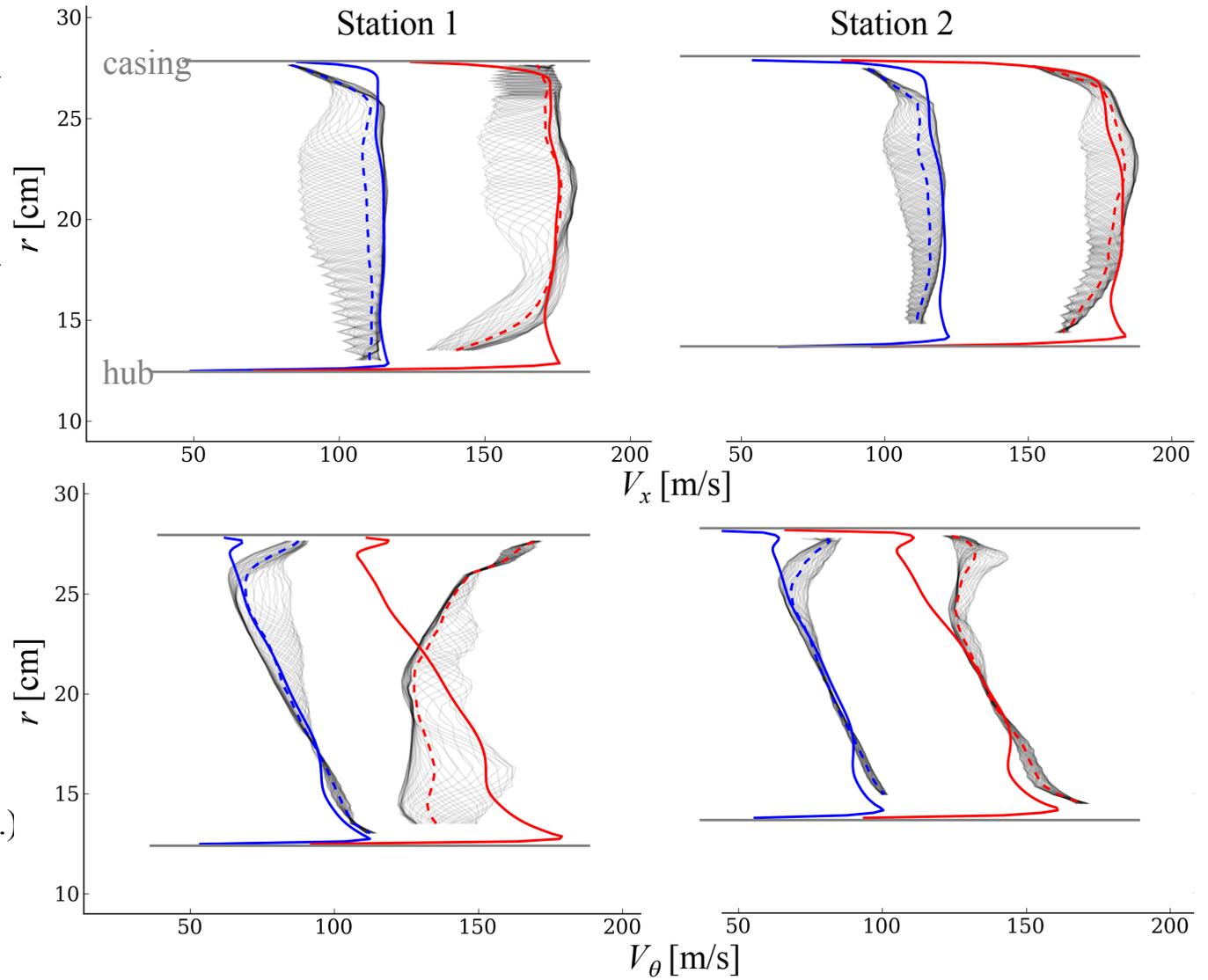
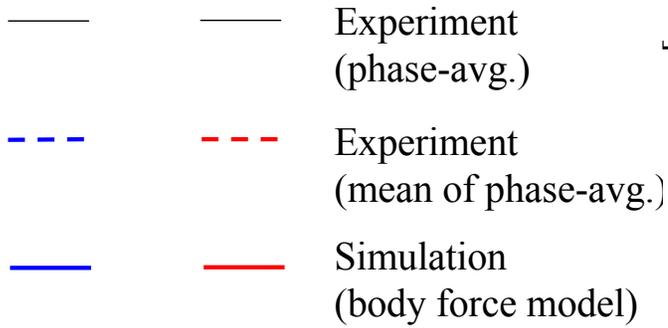
7,808 rpm 12,657 rpm

- Experiment (phase-avg.)
- Experiment (mean of phase-avg.)
- Simulation (body force model)

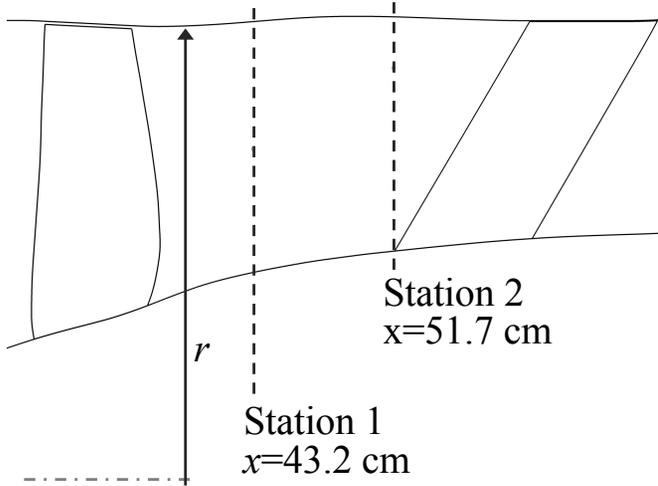
SDT fan results



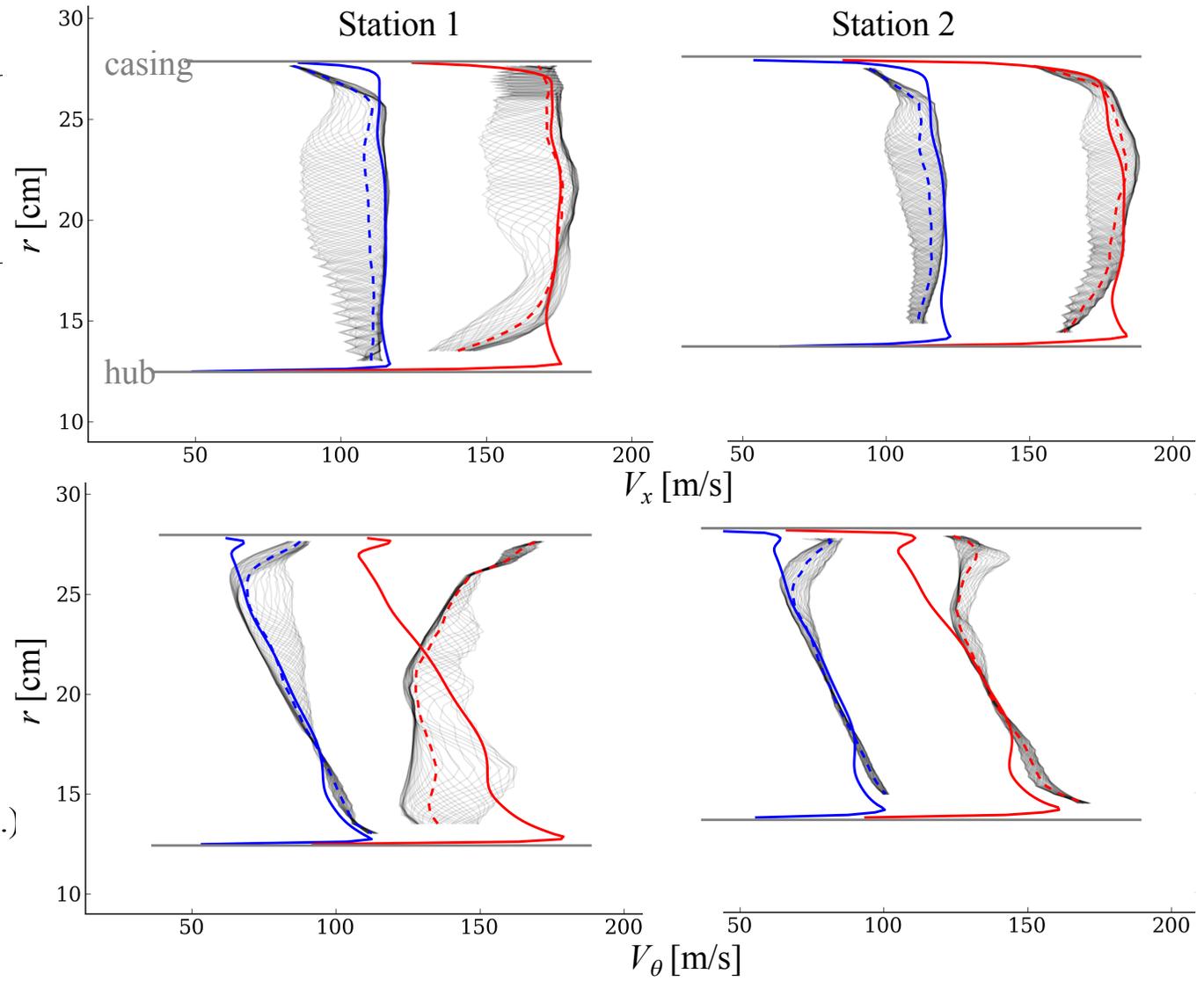
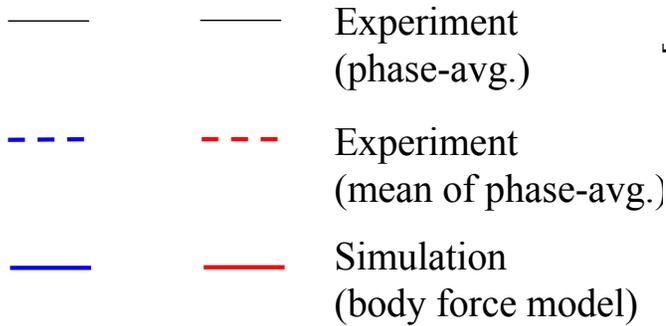
7,808 rpm 12,657 rpm



SDT fan results



7,808 rpm 12,657 rpm



at Station 1, $\Omega=12,657$ rpm

\bar{V}_x [m/s] \bar{V}_θ [m/s] $P_{o,1} / P_{o,\infty}$

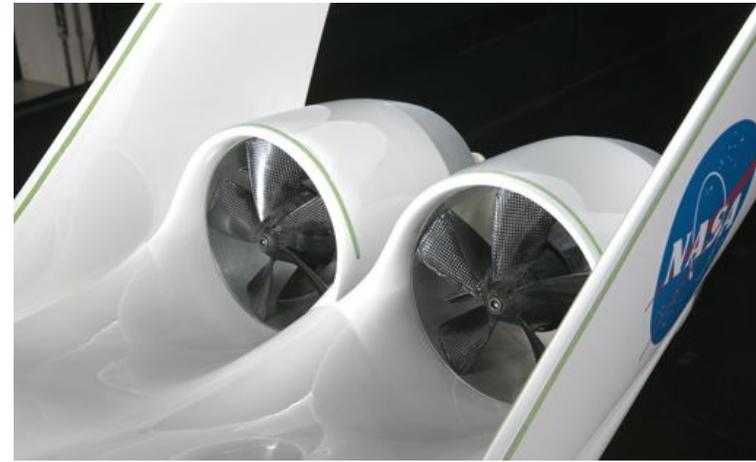
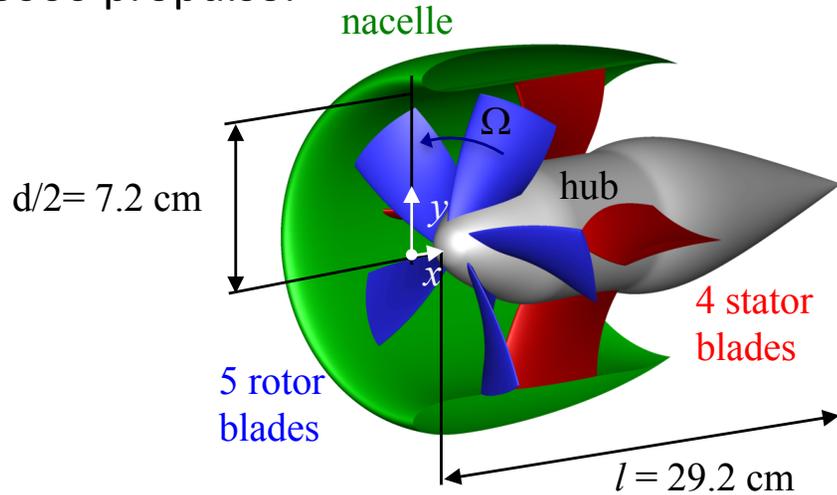
Experiment	171	138	1.509
Body Force Model	172	133	1.491

Hughes et al., 2005

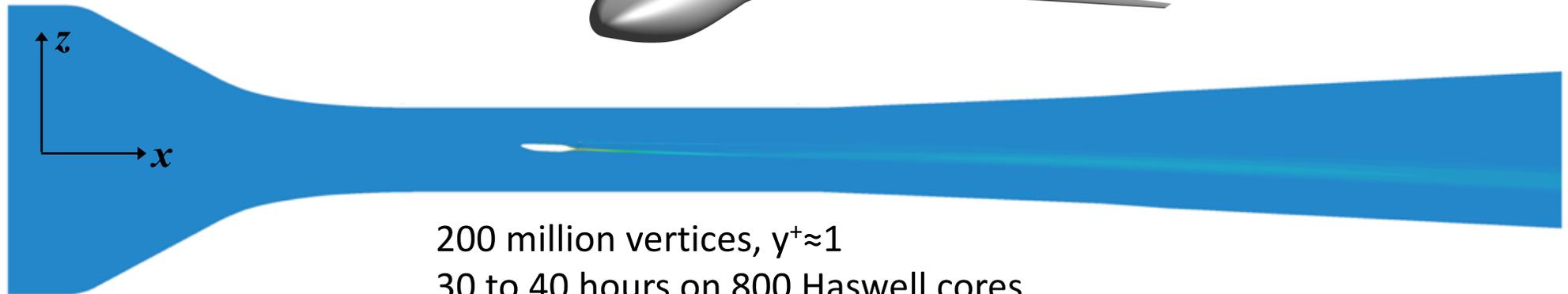
Application?

The D8 wind tunnel model with TF8000 propulsor

TF8000 propulsor



D8 aircraft in wind tunnel



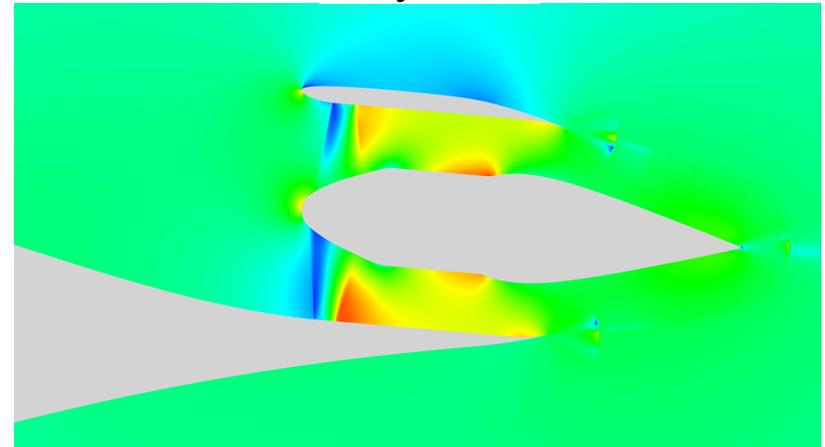
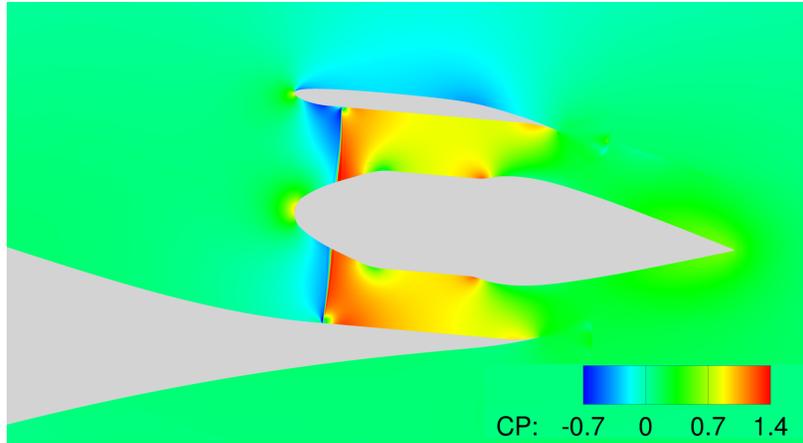
200 million vertices, $y^+ \approx 1$
30 to 40 hours on 800 Haswell cores

The D8 wind tunnel model with TF8000 propulsor

uniform pressure jump

body force

Static Pressure

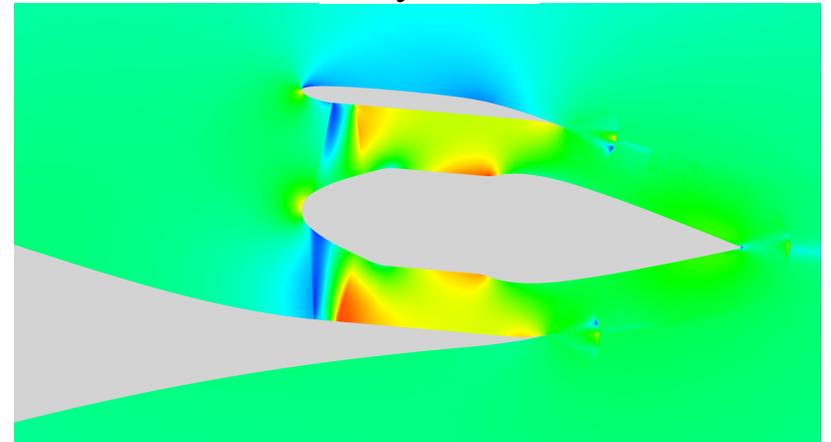
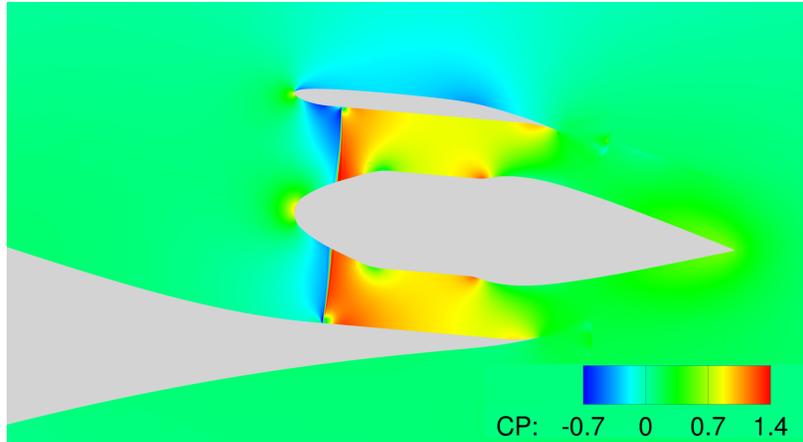


The D8 wind tunnel model with TF8000 propulsor

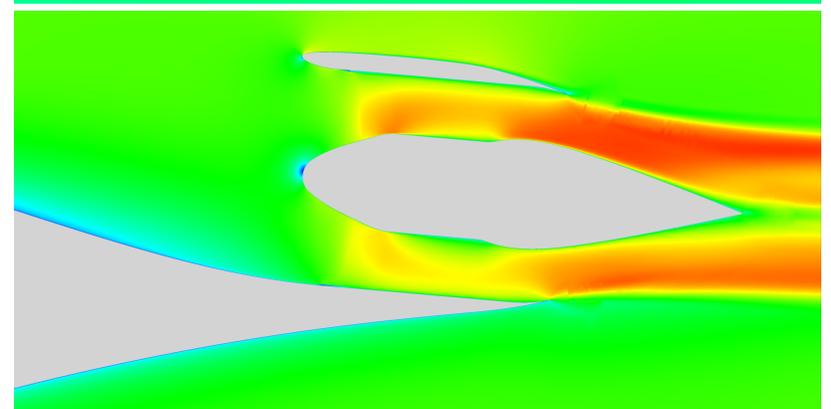
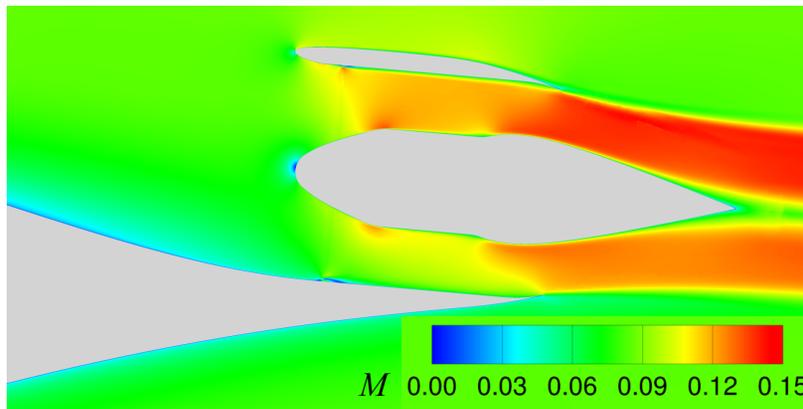
uniform pressure jump

body force

Static Pressure



Mach number

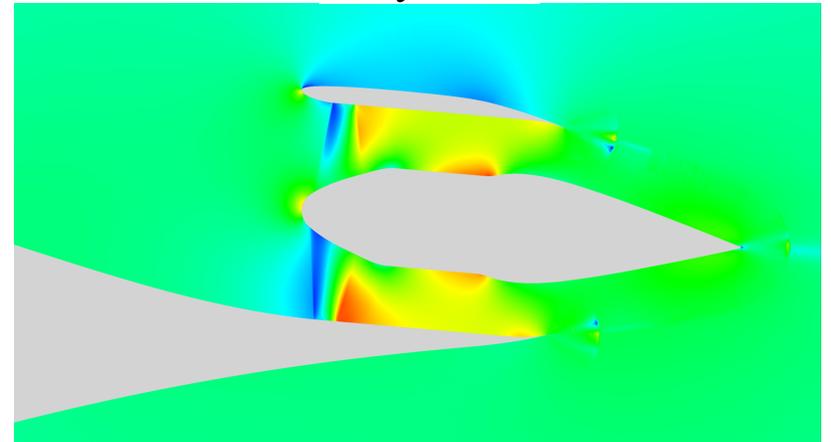
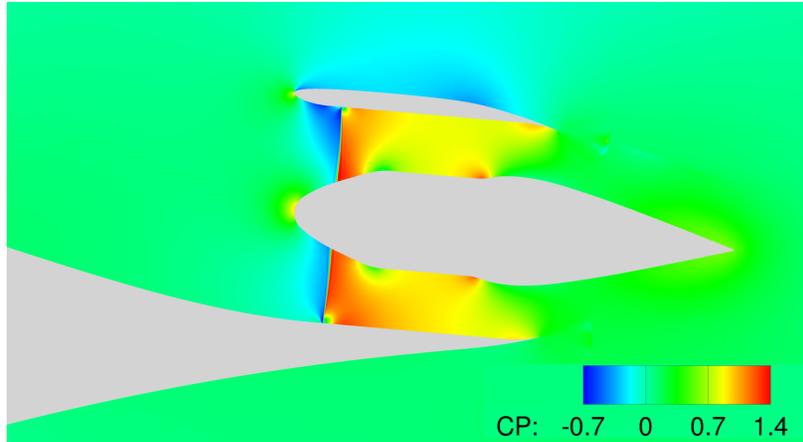


The D8 wind tunnel model with TF8000 propulsor

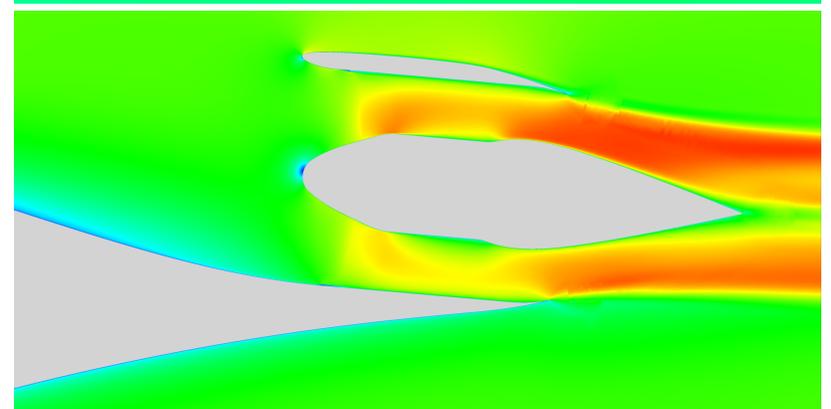
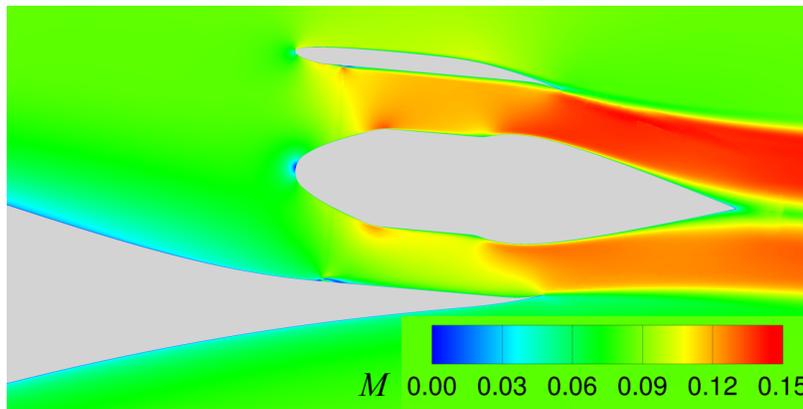
uniform pressure jump

body force

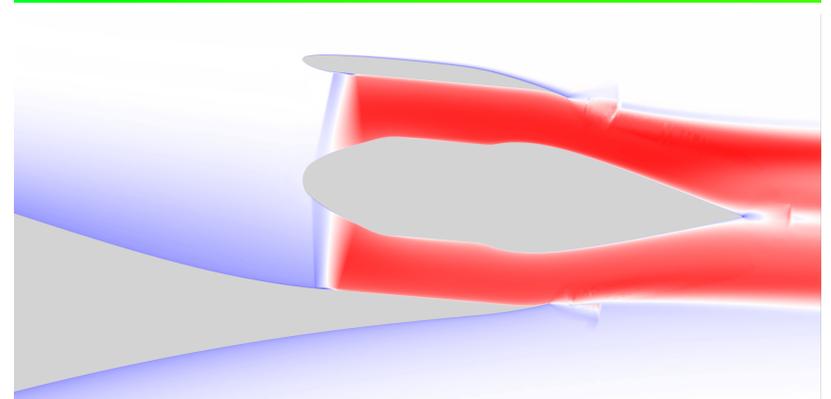
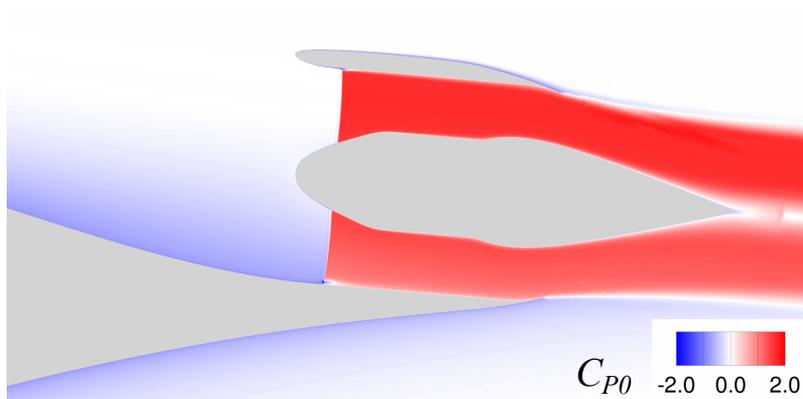
Static Pressure



Mach number

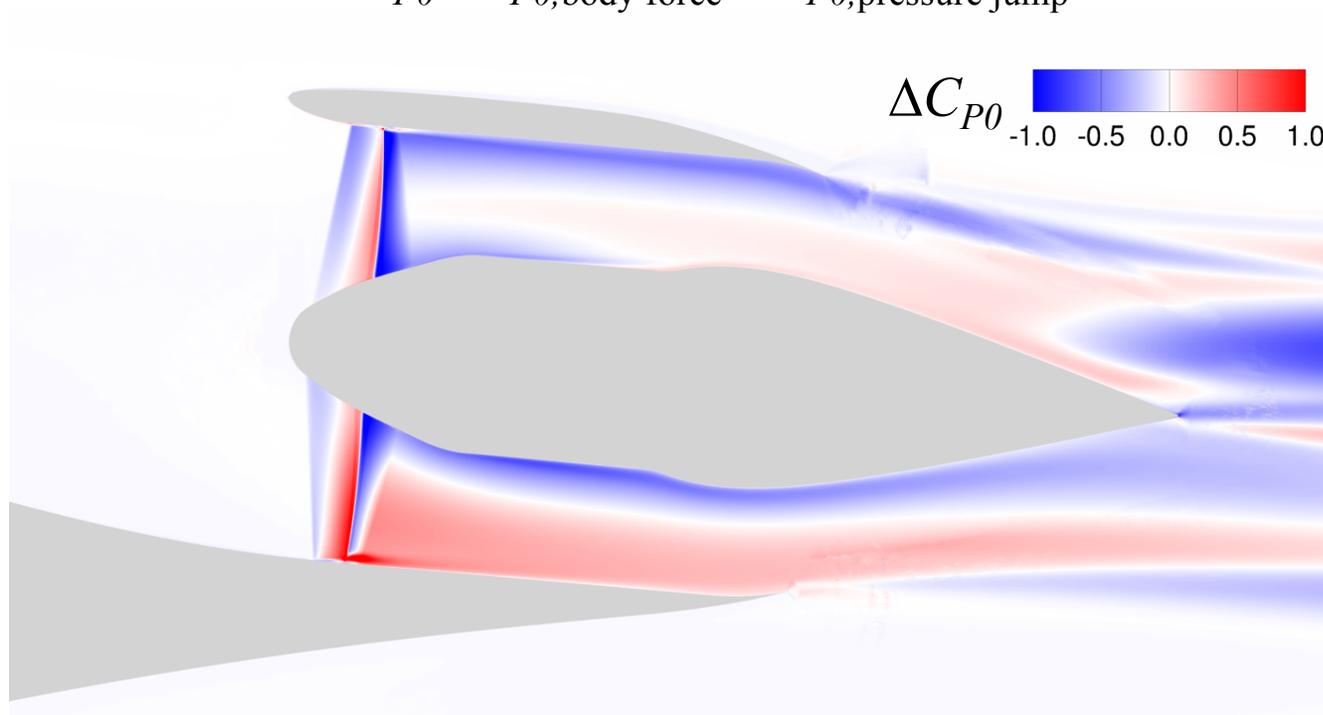


Total Pressure

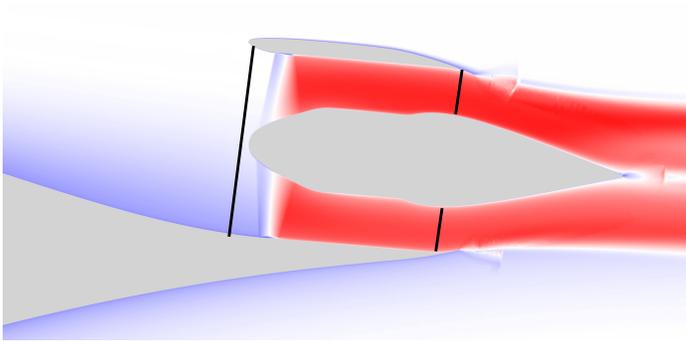
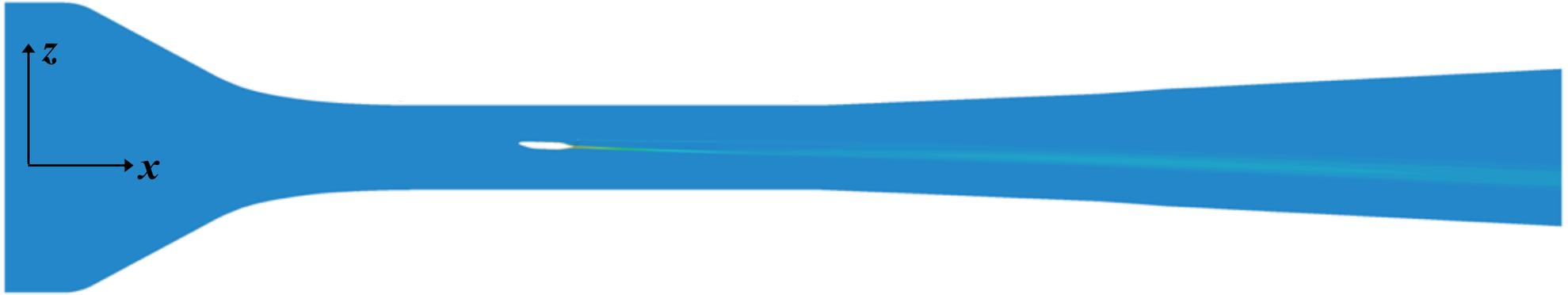


The D8 wind tunnel model with TF8000 propulsor

$$\Delta C_{P0} = C_{P0, \text{body force}} - C_{P0, \text{pressure jump}}$$

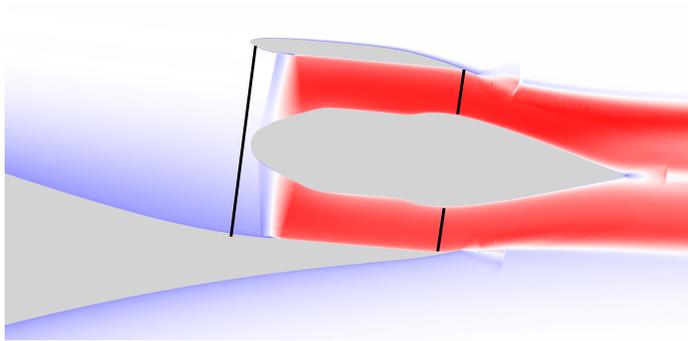
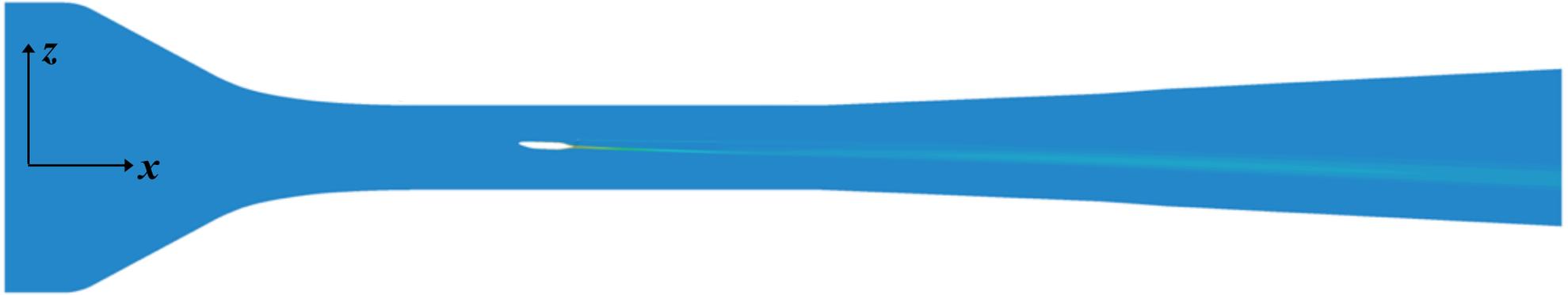


The D8 wind tunnel model with TF8000 propulsor



$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$

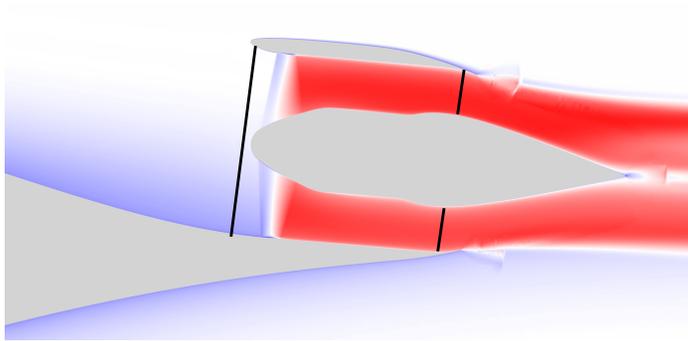
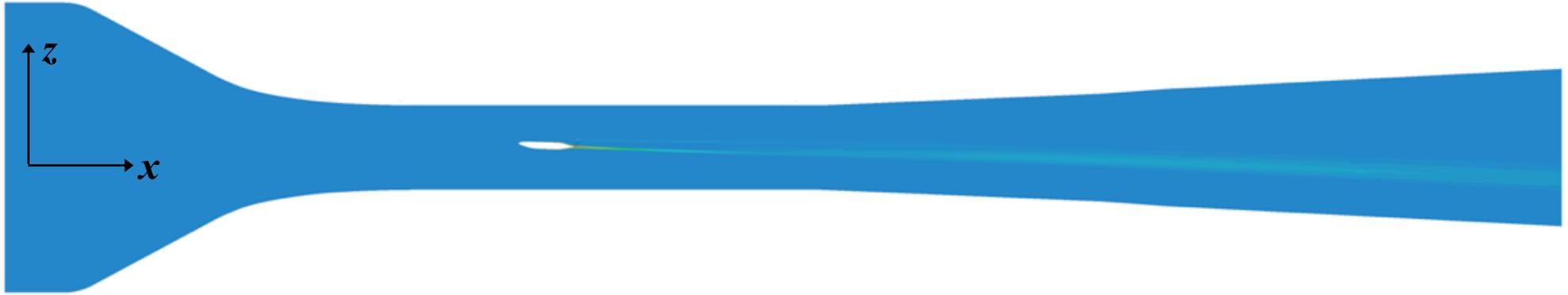
The D8 wind tunnel model with TF8000 propulsor



$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$

Method	C_x	C_z	C_{PK}	$C_{\dot{m}}$
Experiment (11,100 rpm)	0.0000 ± 0.0006	0.644 ± 0.001	0.045 ± 0.001	0.0267 ± 0.0006
Uniform Pressure Jump	0.0002	0.651	0.045	0.0282

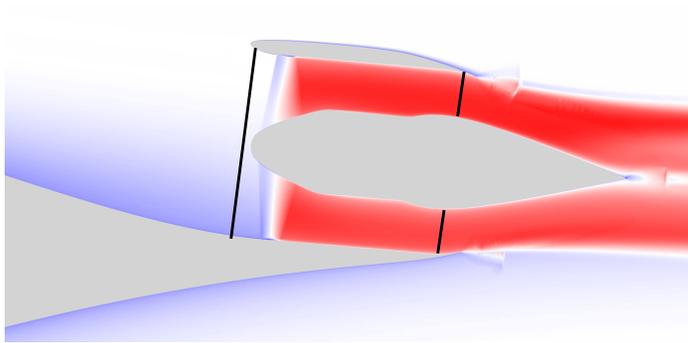
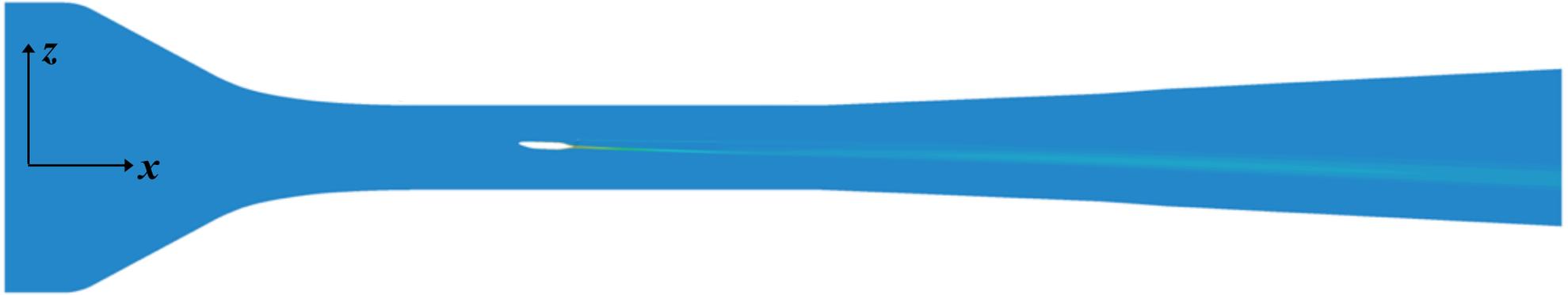
The D8 wind tunnel model with TF8000 propulsor



$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$

Method	C_x	C_z	C_{PK}	$C_{\dot{m}}$
Experiment (11,100 rpm)	0.0000 ± 0.0006	0.644 ± 0.001	0.045 ± 0.001	0.0267 ± 0.0006
Uniform Pressure Jump	0.0002	0.651	0.045	0.0282
Body Force, 11,100 rpm	0.0028	0.672	0.039	0.0275

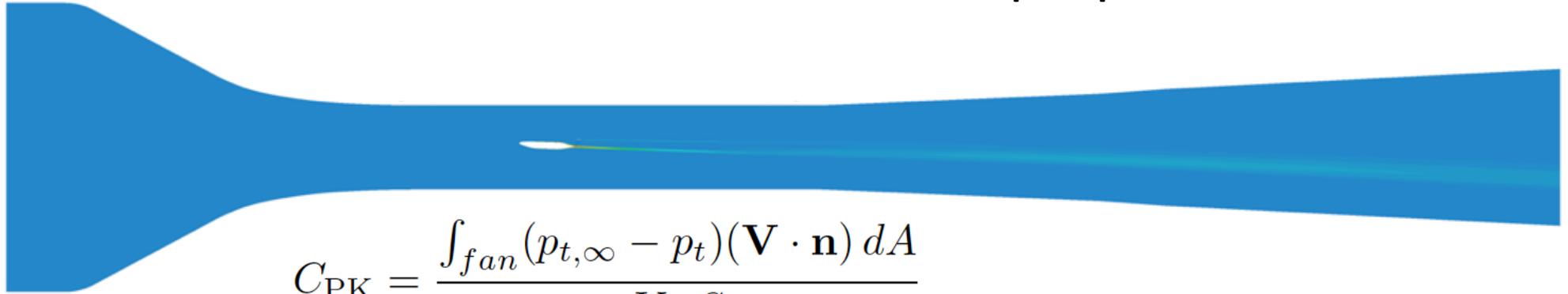
The D8 wind tunnel model with TF8000 propulsor



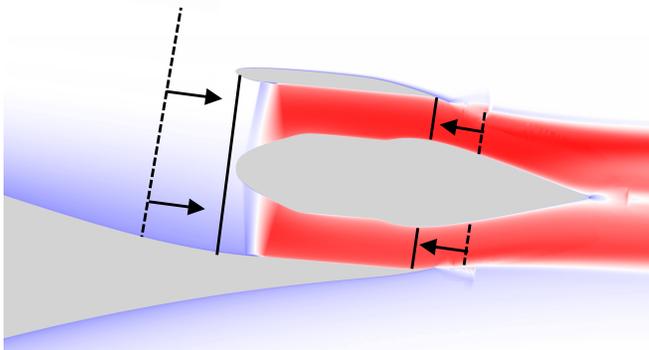
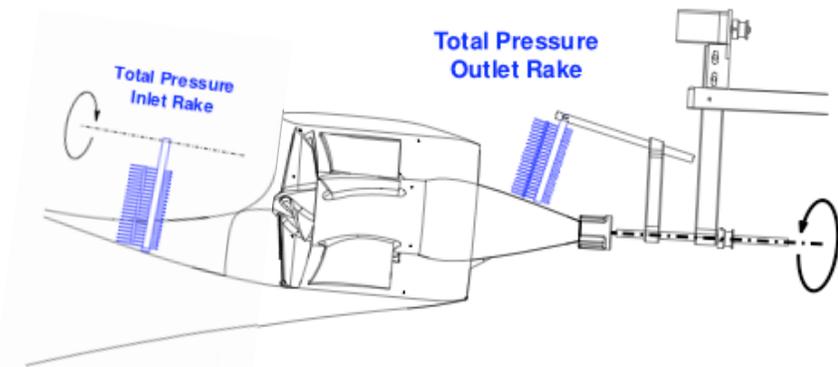
$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$

Method	C_x	C_z	C_{PK}	$C_{\dot{m}}$
Experiment (11,100 rpm)	0.0000 ± 0.0006	0.644 ± 0.001	0.045 ± 0.001	0.0267 ± 0.0006
Uniform Pressure Jump	0.0002	0.651	0.045	0.0282
Body Force, 11,100 rpm	0.0028	0.672	0.039	0.0275
Body Force, 11,450 rpm	0.0005	0.678	0.043	0.0281

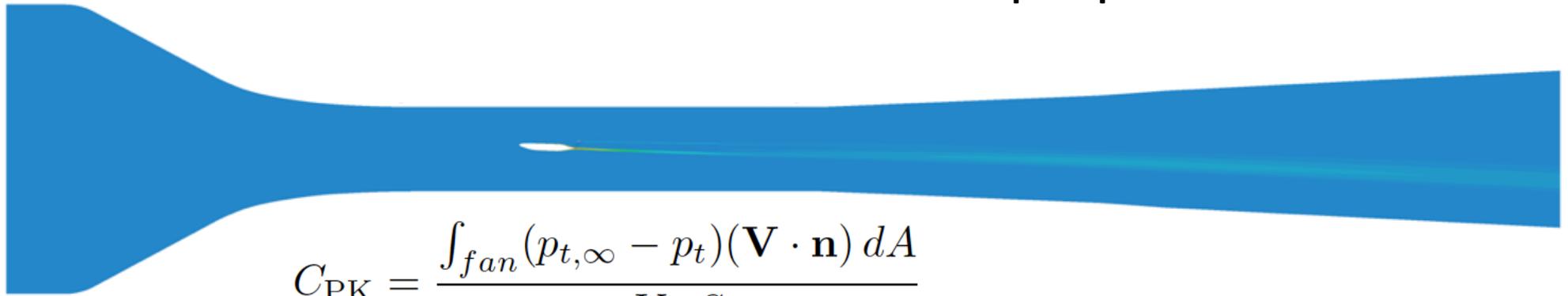
The D8 wind tunnel model with TF8000 propulsor



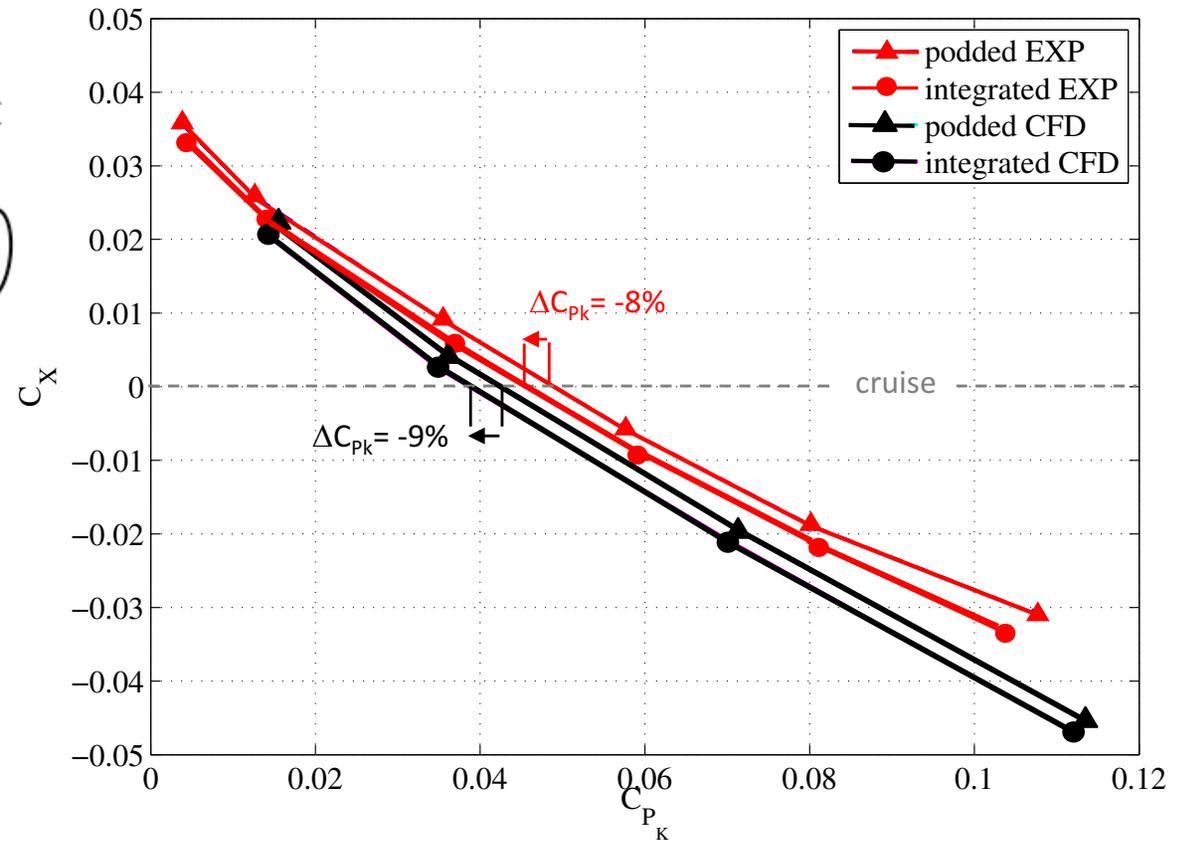
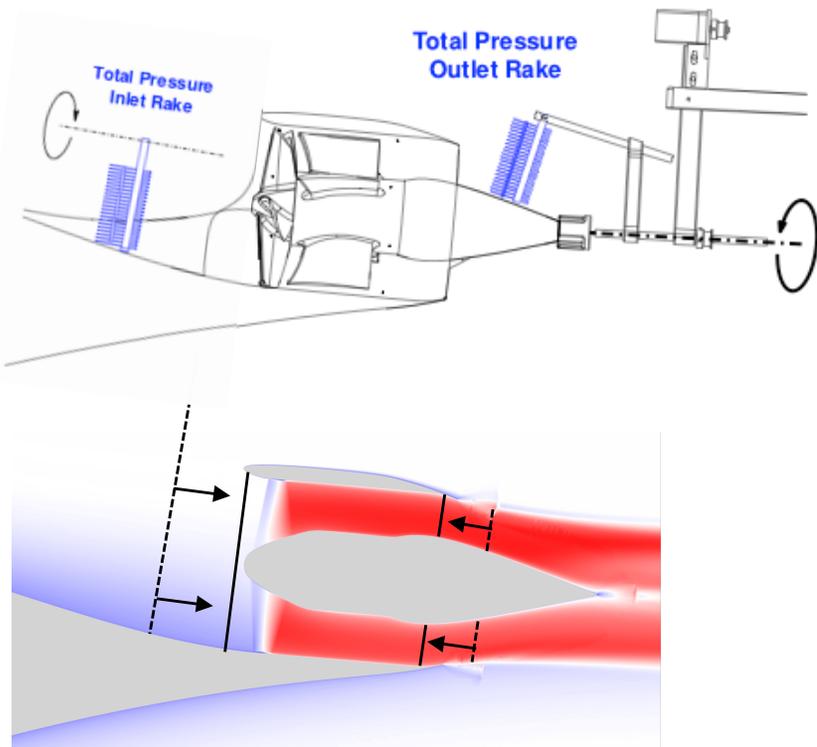
$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$



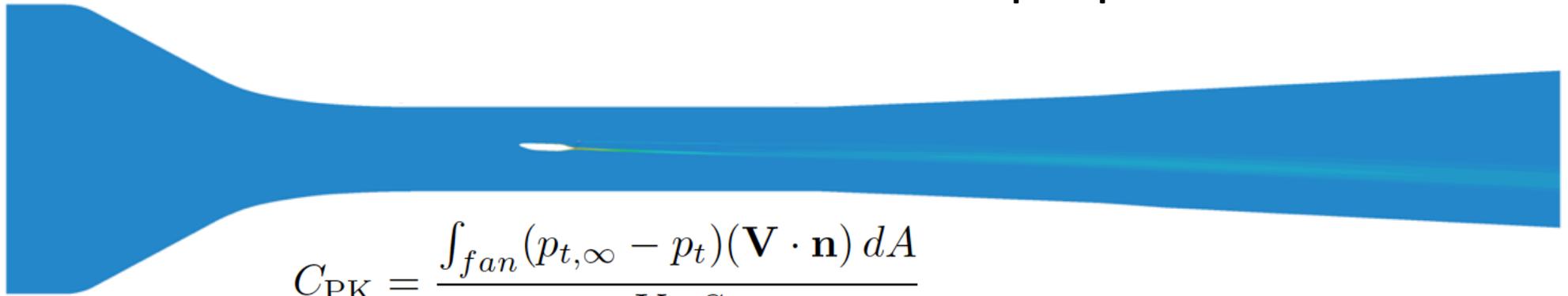
The D8 wind tunnel model with TF8000 propulsor



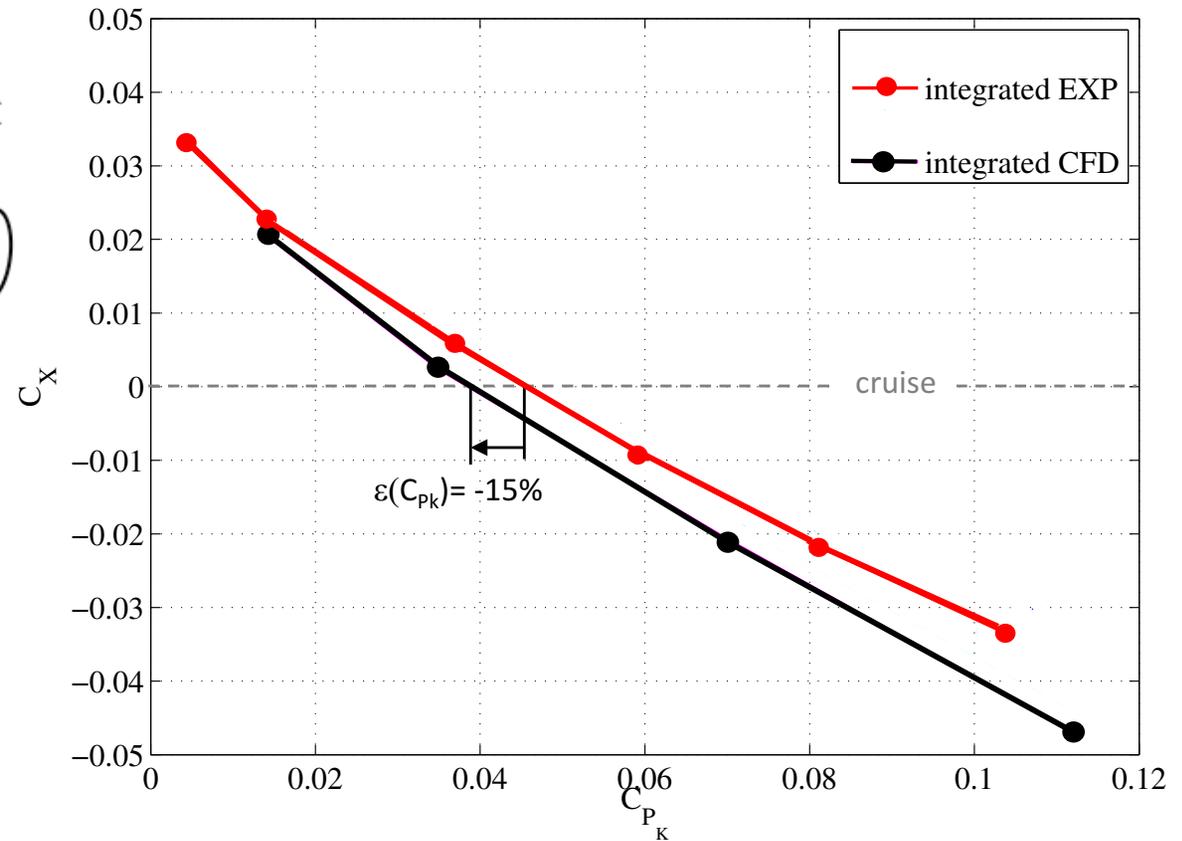
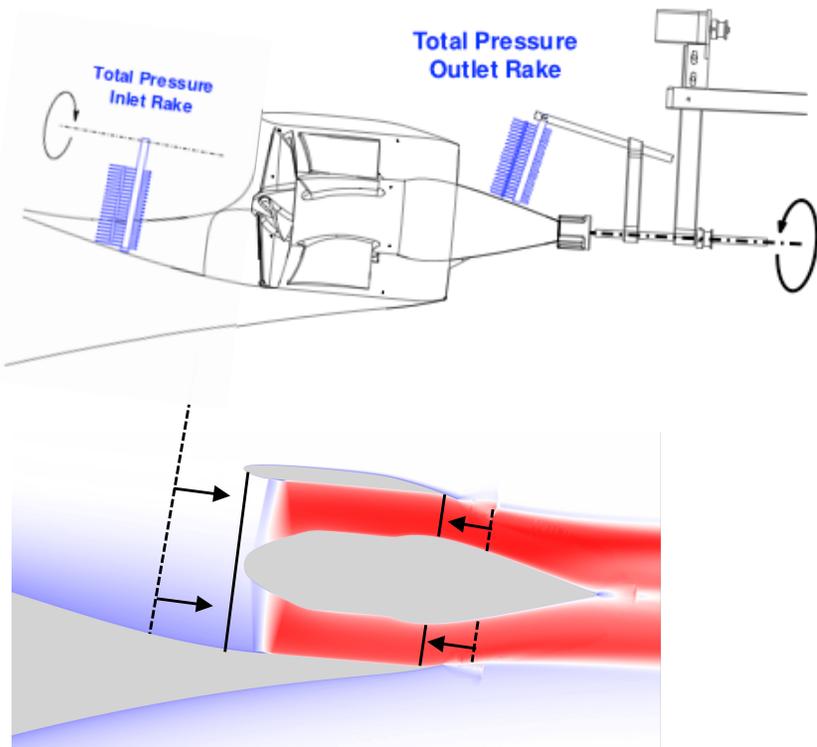
$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$



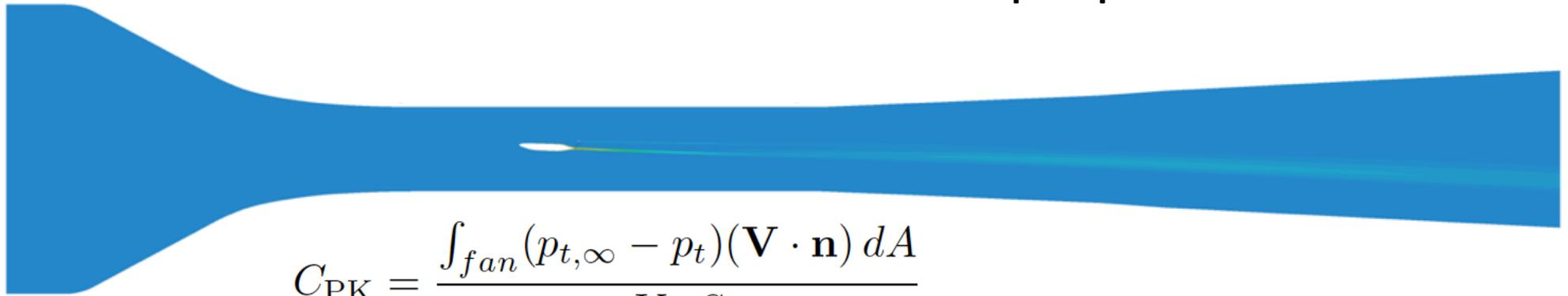
The D8 wind tunnel model with TF8000 propulsor



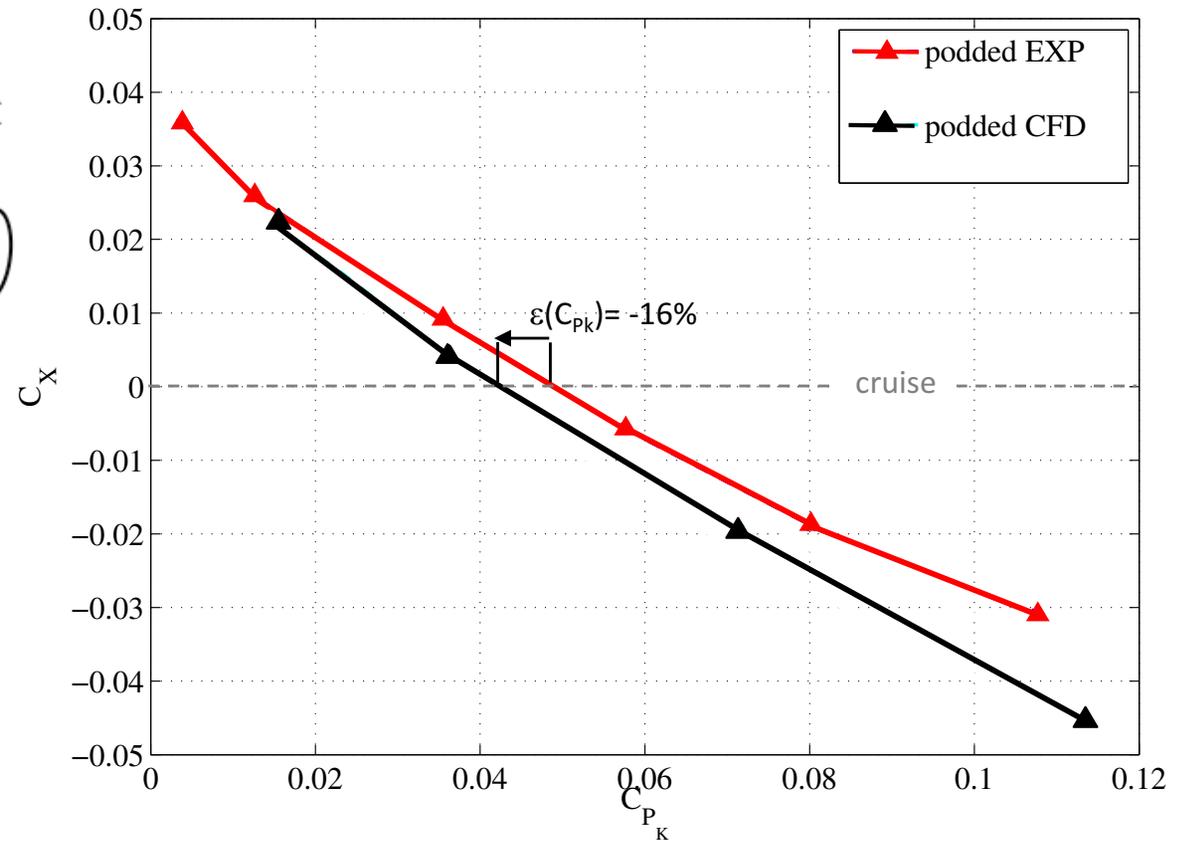
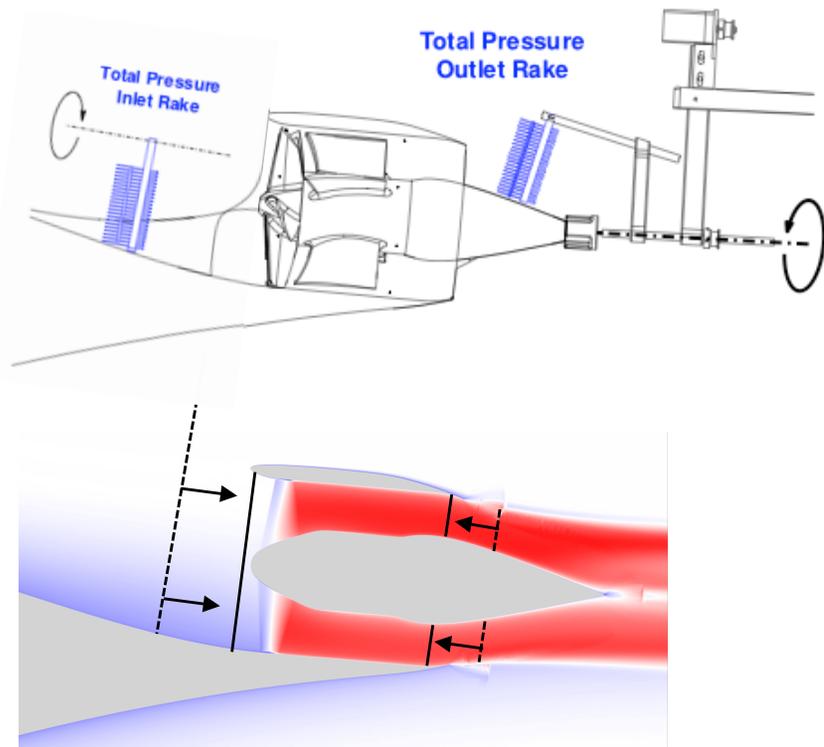
$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$



The D8 wind tunnel model with TF8000 propulsor

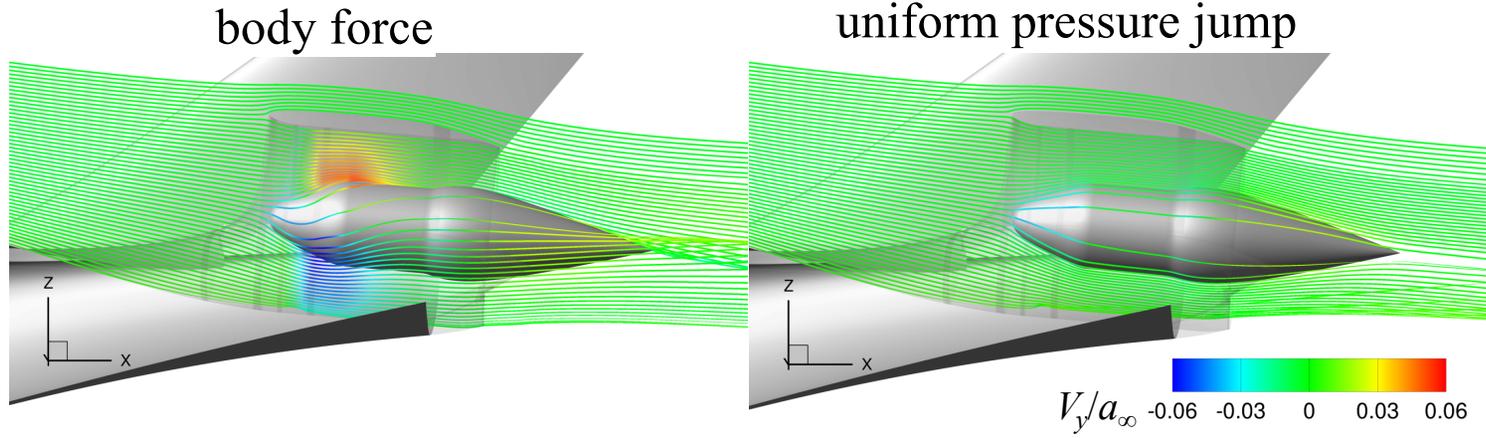


$$C_{PK} = \frac{\int_{fan} (p_{t,\infty} - p_t)(\mathbf{V} \cdot \mathbf{n}) dA}{q_\infty V_\infty S_{ref}}$$



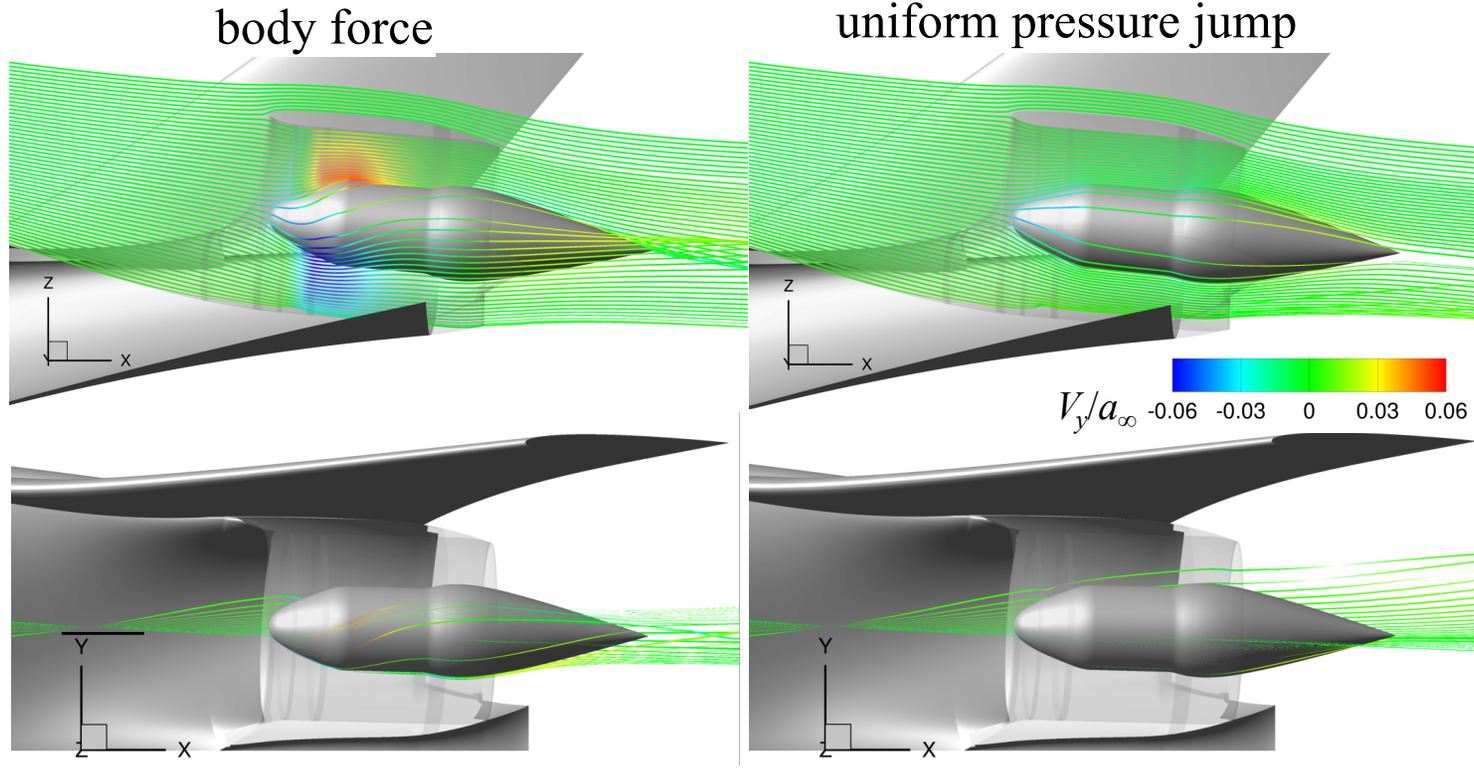
The D8 wind tunnel model with TF8000 propulsor

Swirl



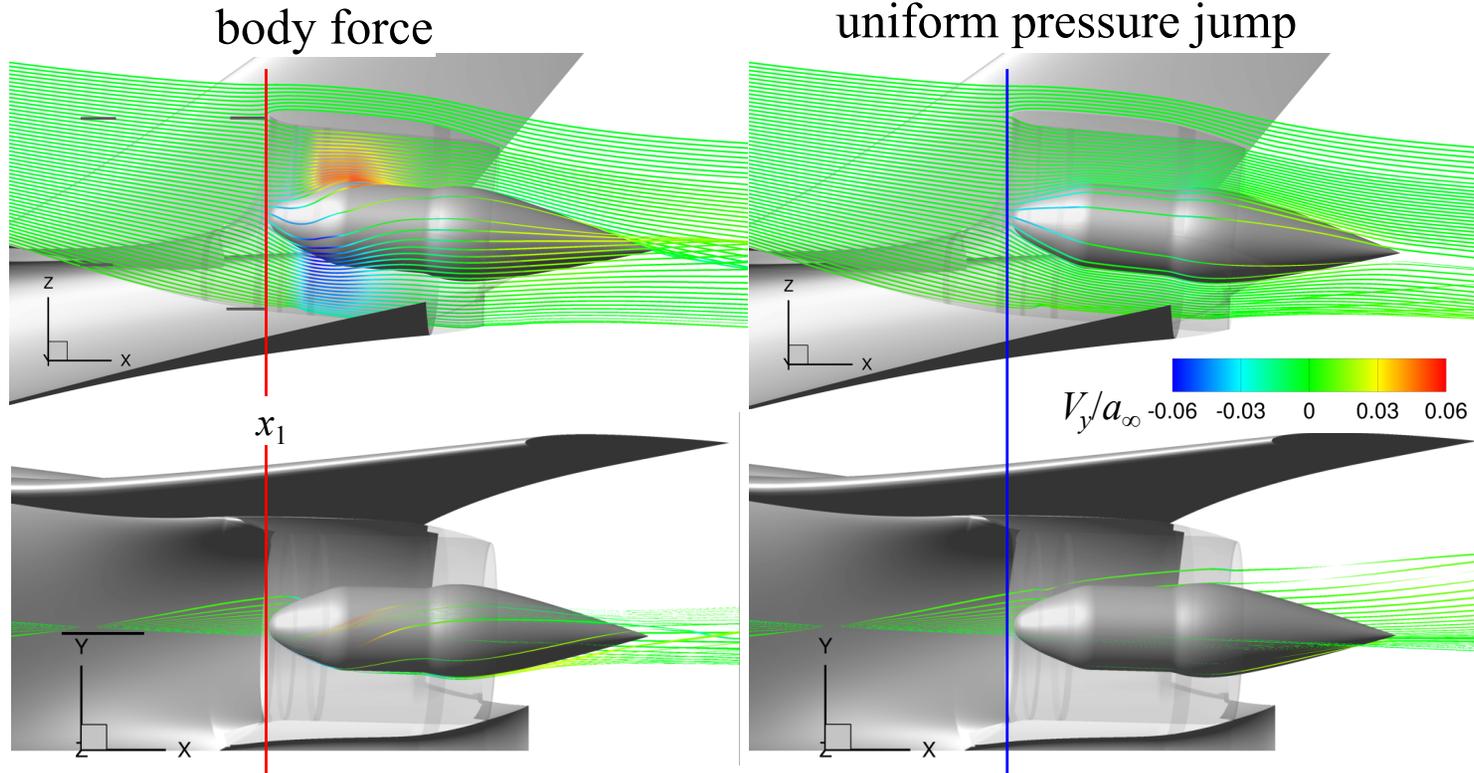
The D8 wind tunnel model with TF8000 propulsor

Swirl



The D8 wind tunnel model with TF8000 propulsor

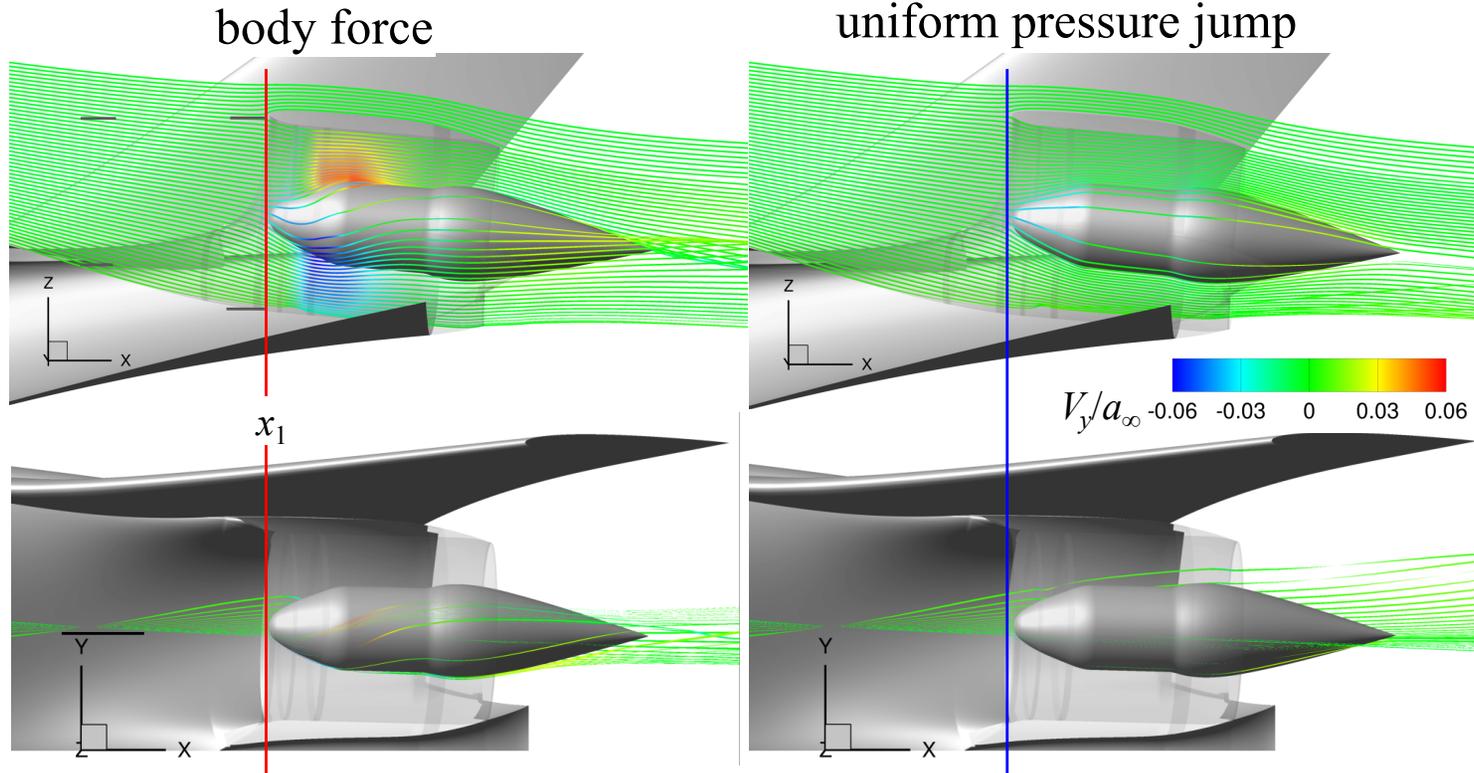
Swirl



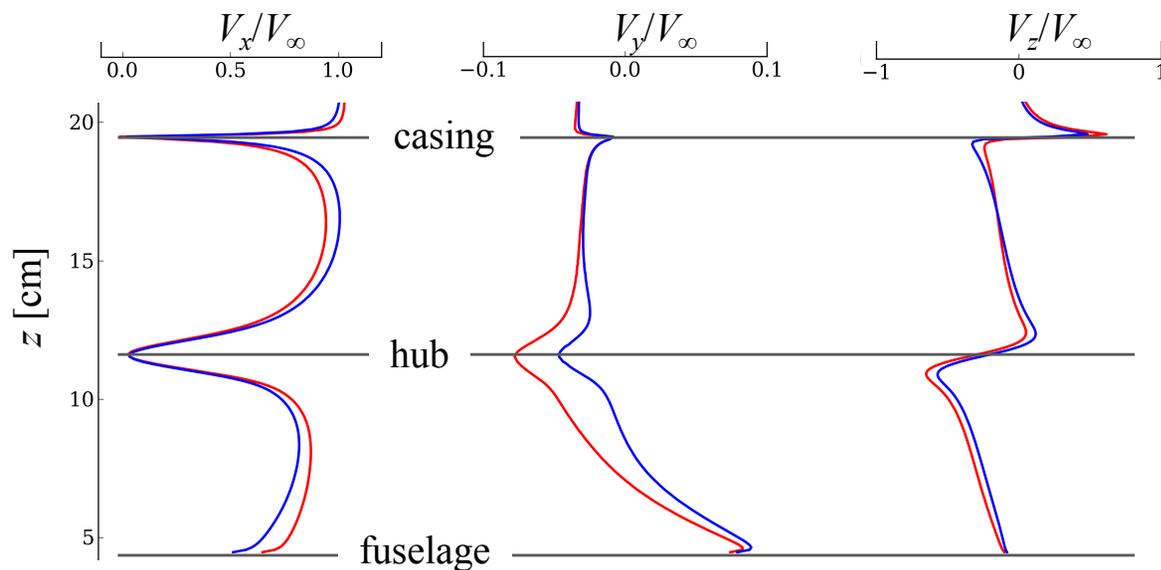
- body force model
- pressure jump model

The D8 wind tunnel model with TF8000 propulsor

Swirl



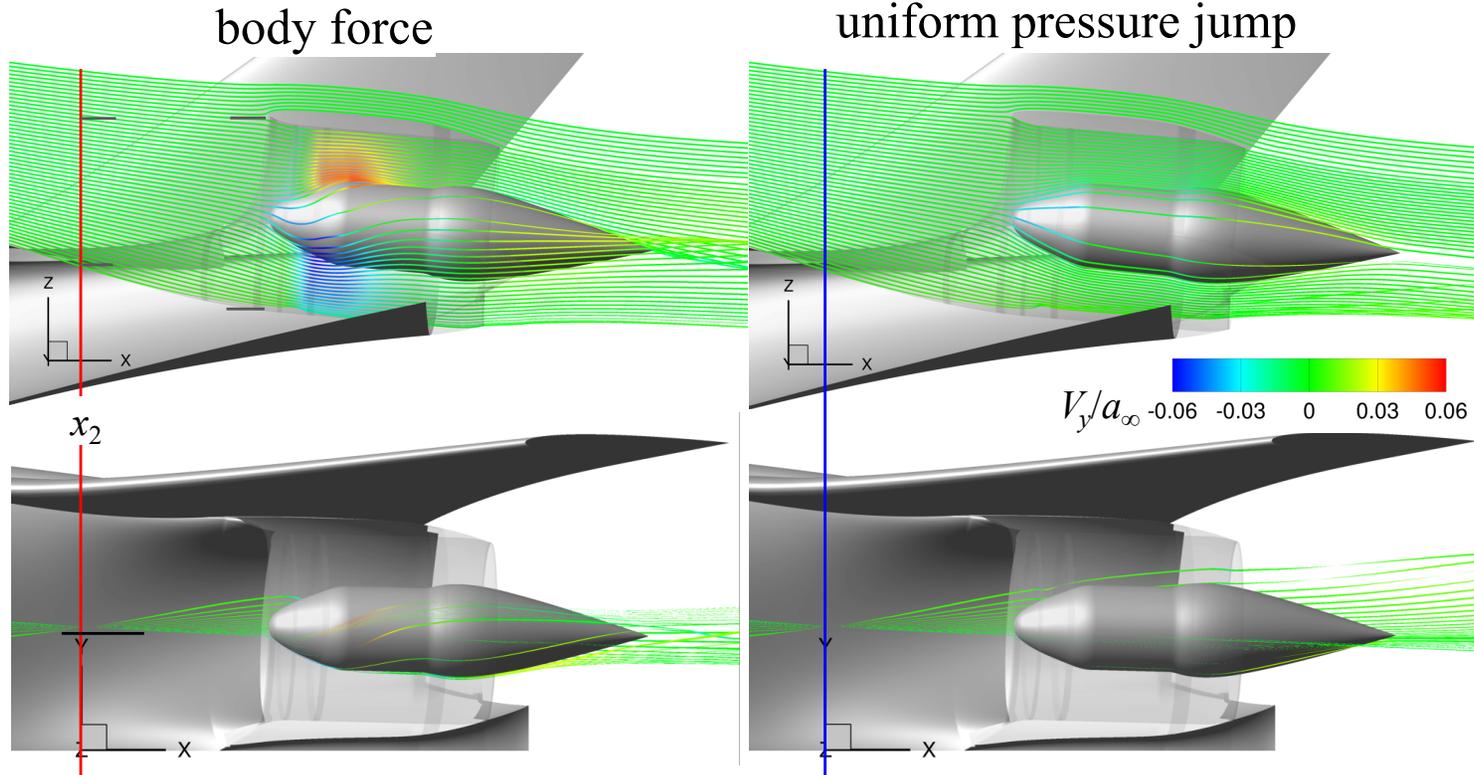
$x_1=2.79$ m:
(fan face)



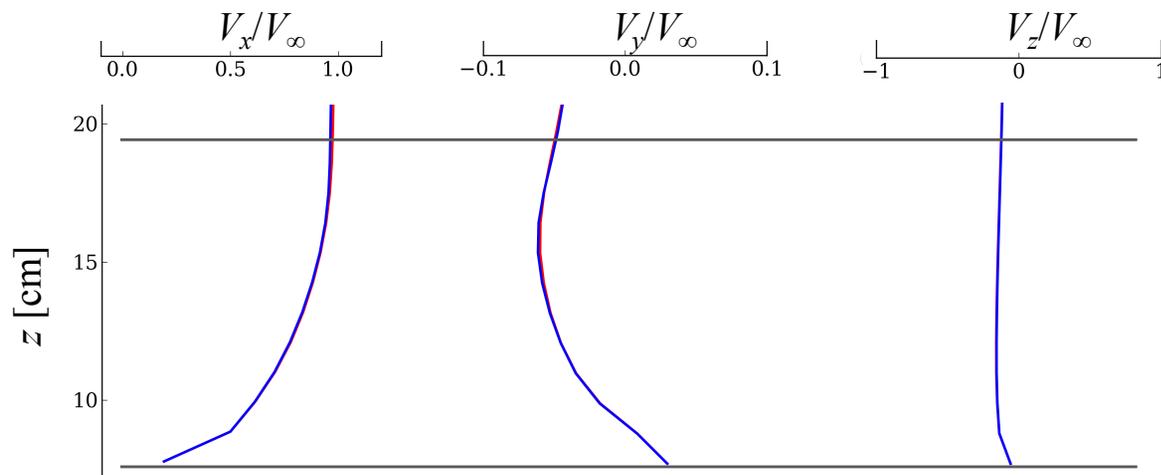
— body force model
— pressure jump model

The D8 wind tunnel model with TF8000 propulsor

Swirl



$x_2=2.67$ m:
($0.88d$ upstream)



Lessons learned

- The body force model predicted some of the integrated quantities within a few percent on SDT with R4 rotor blades. However, some trends were missed especially at the near end-wall regions.
- Further work could include adding compressibility, blade blockage and endwall corrections into the blade loading.
- But before those enhancements, a grid and solution scheme study must be made.
- When applied on a BLI (boundary layer ingestion) setting, the model provided insights that could not have been obtained by the pressure jump model (effect of swirl, buildup of pressure in the fan, attenuation of the distortion)
- We need a detailed test data on an isolated BLI propulsor to validate the model with a higher confidence. SDT did not have BLI, D8 model did not have measurements within the propulsor.

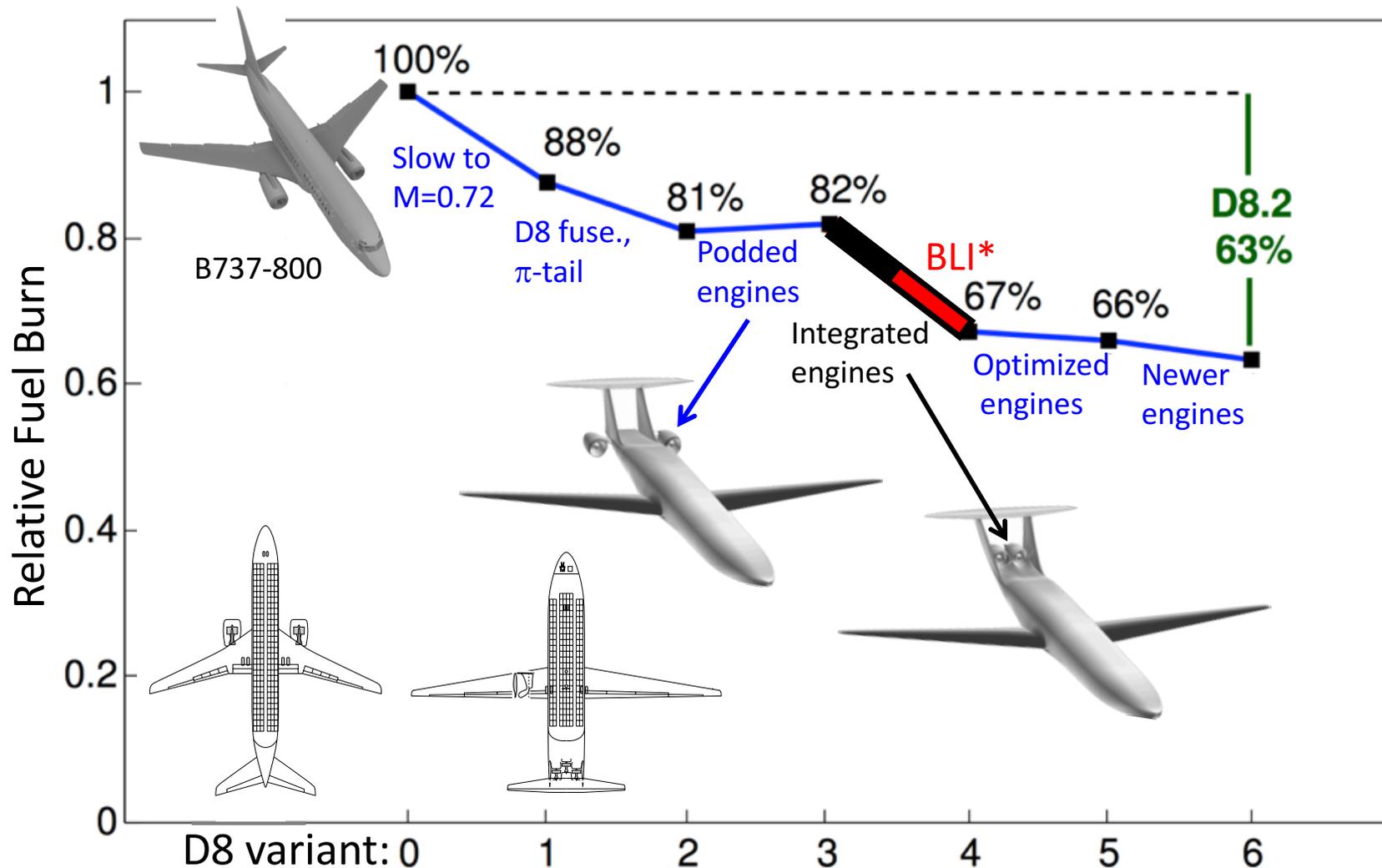
Acknowledgements

- Dr. David K. Hall of the MIT Gas Turbine Laboratory provided a description of the source term computation algorithm.
- Dr. Edmane Envia of NASA Glenn Research Center provided the SDT aerodynamic data and geometry definition files.
- NASA Advanced Air Transport Technology (AATT) project provided the funding for this work.
- NASA Advanced Supercomputing (NAS) Division at NASA Ames Research Center provided computing resources.

Questions & Comments

Back-up slides

>30% reduction in fuel burn due to the synergistic *integration* of airframe components



*BLI: Boundary Layer Ingestion

Greitzer *et al.*, *N+3 Aircraft Concept Designs and Trade Studies. Volume 1*, 2010, NASA CR-2010-216794/VOL1

Uranga *et al.*, *Preliminary Experimental Assessment of the Boundary Layer Ingestion Benefit for the D8 Aircraft*, AIAA-2014-0906 71/69

Test Cases



A stand-alone Source Diagnostics Test (SDT) fan with R4 rotor blades

Test Cases



A stand-alone Source Diagnostics Test (SDT) fan with R4 rotor blades



A stand-alone TF8000 propulsor

Test Cases



A stand-alone Source Diagnostics Test (SDT) fan with R4 rotor blades



A stand-alone TF8000 propulsor



The D8 aircraft model in a wind tunnel