ROYAL INSTITUTE
OF TECHNOLOGY

Adaptive FEM for Aerospace and Aeroacoustics Applications

Rodrigo Vilela de Abreu   Niclas Jansson   Johan Jansson
Johan Hoffman

Computational Technology Laboratory, CSC/HPCViz, KTH

5th **ANSA** and μ**ETA** International Conference, Thessaloniki,
5-7 June, 2013.

About us

What do we do?

- Develop the **open-source** turbulent flow solver **UNICORN**.
  - *FEniCS open-source project.*
  - *MPI-IO, parallel mesh-refinement, dynamic load-balancing.*
  - *Adjoint based mesh adaptivity.*
  - *Linear scalability up to 12,000 cores.*
- Study **turbulent flow** phenomena with several **applications**.
- 1 Professor, 1 senior researcher, 2 post-docs, 5 PhD students.

Who am I?

- 4^{th} year (of 5) **PhD candidate**.
- Use **UNICORN** to study **aerodynamics** and **aeroacoustics**.
  - *Separation, airframe noise (landing gear, slat-noise, etc) and duct-acoustics.*

ROYAL INSTITUTE OF TECHNOLOGY

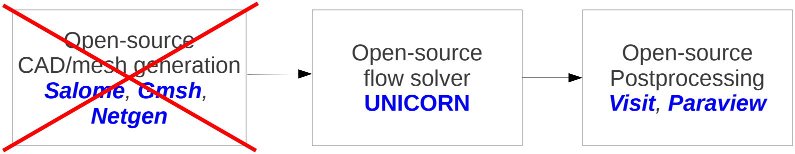
Collaboration with **ANSA**

Workflow

ROYAL INSTITUTE OF TECHNOLOGY

Collaboration with ANSA

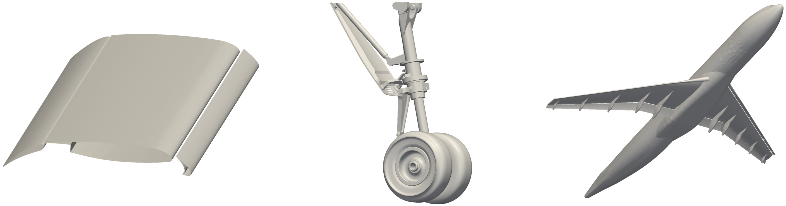
Workflow



Difficult to use for complex geometries...

Collaboration with ANSA

ANSA mesh generation



From left to right...

- 30P30N from NASA, benchmark workshops BANC-I and BANC-II.
- Gulfstream G550 nose landing gear, also BANC-I and BANC-II.
- DLR model airplane, High-Lift Prediction Workshop 2.

Collaboration with ANSA

ANSA mesh generation

From left to right...

- 30P30N from NASA, benchmark workshops BANC-I and BANC-II.
- Gulfstream G550 nose landing gear, also BANC-I and BANC-II.
- DLR model airplane, High-Lift Prediction Workshop 2.

ROYAL INSTITUTE OF TECHNOLOGY

Adaptive FEM for turbulent flows

General Galerkin (G2)

- **FEM** with piecewise linear approximation in space and time.
- Fully **unstructured meshes**.
- Time-resolved method where **numerical stabilization** accounts for unresolved scales.
- **Simple wall shear stress model** based on skin friction, slip velocity boundary condition, in the spirit of simpler models.[1]
- **Adaptive mesh refinement** with respect to output of interest using associated adjoint problem to estimate errors in output.

---

[1] U. Schumann, *Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli.*

ROYAL INSTITUTE OF TECHNOLOGY

Adaptive FEM for turbulent flows

Adjoint-based mesh refinement

For $\hat{U} = (U, P)$ a weak solution, $\hat{\varphi} = (\varphi, \theta)$ a solution to a linearized adjoint problem, and $M(\hat{U}) = ((\hat{U}, \hat{\psi}))$ a mean value output, with $\hat{\psi}$ a weight function, we define the error estimate:

$$\boxed{|M(\hat{u}) - M(\hat{U})| = |((\hat{u} - \hat{U}, \hat{\psi}))| \leq \sum_{K \in \mathcal{T}_n} \mathcal{E}_K,}$$

with the error indicator:

$$\mathcal{E}_K \equiv \sum_{n=1}^{N} \left[\int_{I_n} |R_1(\hat{U})|_K \cdot \omega_1 \, dt + \int_{I_n} |R_2(U)|_K \, \omega_2 \, dt + \int_{I_n} |SD_\delta^n(\hat{U}; \hat{\varphi})_K| \, dt \right],$$

for each element K in the mesh $\mathcal{T}_n$, with stability weights ω_i, $i = 1, 2$.

Adaptive FEM for turbulent flows

Adjoint-based mesh refinement

For $\hat{U} = (U, P)$ a weak solution, $\hat{\varphi} = (\varphi, \theta)$ a solution to a linearized adjoint problem, and $M(\hat{U}) = ((\hat{U}, \hat{\psi}))$ a mean value output, with $\hat{\psi}$ a weight function, we define the error estimate:

$$|M(\hat{u}) - M(\hat{U})| = |((\hat{u} - \hat{U}, \hat{\psi}))| \leq \sum_{K \in \mathcal{T}_n} \mathcal{E}_K,$$

with the error indicator:

$$\mathcal{E}_K \equiv \sum_{n=1}^{N} \left[\int_{I_n} |R_1(\hat{U})|_K \cdot \omega_1 \, dt + \int_{I_n} |R_2(U)|_K \, \omega_2 \, dt + \int_{I_n} |SD_\delta^n(\hat{U}; \hat{\varphi})_K| \, dt \right],$$

for each element K in the mesh $\mathcal{T}_n$, with stability weights ω_i, $i = 1, 2$.

Adaptive FEM for turbulent flows

Adjoint-based mesh refinement

For $\hat{U} = (U, P)$ a weak solution, $\hat{\varphi} = (\varphi, \theta)$ a solution to a linearized adjoint problem, and $M(\hat{U}) = ((\hat{U}, \hat{\psi}))$ a mean value output, with $\hat{\psi}$ a weight function, we define the error estimate:

$$|M(\hat{u}) - M(\hat{U})| = |((\hat{u} - \hat{U}, \hat{\psi}))| \leq \sum_{K \in \mathcal{T}_n} \mathcal{E}_K,$$

with the error indicator:

$$\mathcal{E}_K \equiv \sum_{n=1}^{N} \left[\int_{I_n} |R_1(\hat{U})|_K \cdot \omega_1 \, dt + \int_{I_n} |R_2(U)|_K \, \omega_2 \, dt + \int_{I_n} |SD_\delta^n(\hat{U}; \hat{\varphi})_K| \, dt \right],$$

for each element K in the mesh $\mathcal{T}_n$, with stability weights ω_i, $i = 1, 2$.

ROYAL INSTITUTE OF TECHNOLOGY

Adaptive FEM for turbulent flows

Adjoint-based mesh refinement

For $\hat{U} = (U, P)$ a weak solution, $\hat{\varphi} = (\varphi, \theta)$ a solution to a linearized adjoint problem, and $M(\hat{U}) = ((\hat{U}, \hat{\psi}))$ a mean value output, with $\hat{\psi}$ a weight function, we define the error estimate:

$$|M(\hat{u}) - M(\hat{U})| = |((\hat{u} - \hat{U}, \hat{\psi}))| \leq \sum_{K \in \mathcal{T}_n} \mathcal{E}_K,$$

with the error indicator:

error in M(û) ≡ f(turbulence,adjoint solution)

for each element K in the mesh $\mathcal{T}_n$, with stability weights ω_i, $i = 1, 2$.

Adaptive Mesh Refinement

How do we generate the mesh?

Adaptive algorithm

1. For the mesh $\mathcal{T}_n$: compute primal and adjoint problem.
2. Compute $\mathcal{E}_K$, $K \in \mathcal{T}_n$.
3. Mark 10% of the elements with highest "error indicator" for refinement.
4. Generate the refined mesh $\mathcal{T}_{n+1}$, and goto 1.

Example 30P30N high-lift wing:

Initial mesh: 1M cells.
Mesh after 7 adaptive refinements: 6.6M cells.

$\Rightarrow$ Compare, e.g., with Imamura et al, 16.3M points![2]

[2]Imamura, T., Murayama, M., Hirai, T., and Yamamoto, K., *Aeroacoustic Simulations around 30P30N, JAXA's Result,"* *Proceedings for BANC-II*, 2012.

Adaptive Mesh Refinement

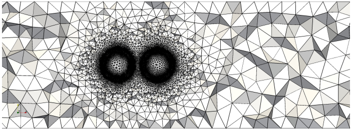
How to choose the refinement target $M(\hat{u})$?

It depends on the application...

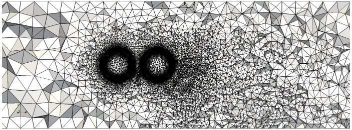
- For aerodynamics, drag, lift or drag+lift.
- For external aeroacoustics, Lighthill's analogy.
- Duct acoustics, pressure drop.
- ...

Adaptive Mesh Refinement

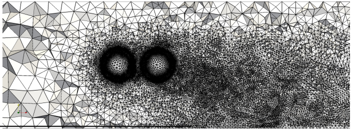
Initial mesh[3]:



3 refinements:



9 refinements:



[3] Vilela de Abreu et at, Adaptive computation of aeroacoustic sources for a rudimentary landing gear using Lighthill's analogy, Proceedings for the 17th AIAA/CEAS Aeroacoustics Conference, 2011.

ROYAL INSTITUTE
OF TECHNOLOGY

Adaptive Mesh Refinement

What are the advantages of an adaptively generated mesh?

- Mesh captures the relevant flow features.
- No need for *ad hoc* meshing.
- No need for a “mesh study” [4].
- Final mesh has “optimal” size.

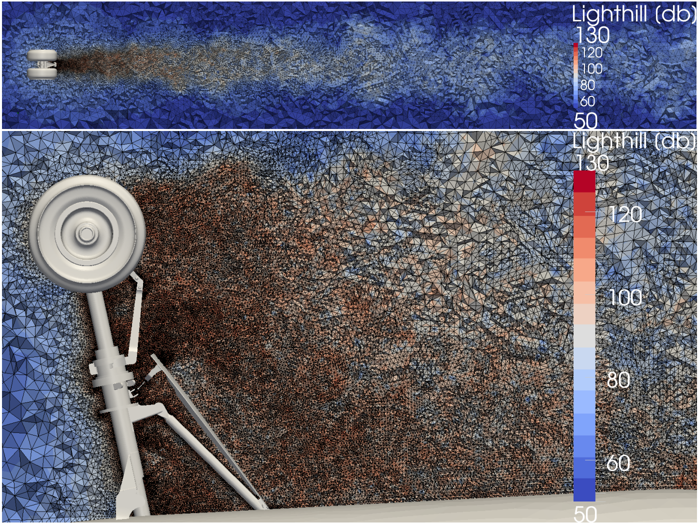
[4] A hierarchy of meshes is automatically generated by the adaptive algorithm and flow solutions are available for all meshes. Moreover, a stop criterion for the algorithm should be chosen to ensure “mesh convergence”.

Adaptive Mesh Refinement

Mesh captures the relevant flow features...

Adaptive Mesh Refinement

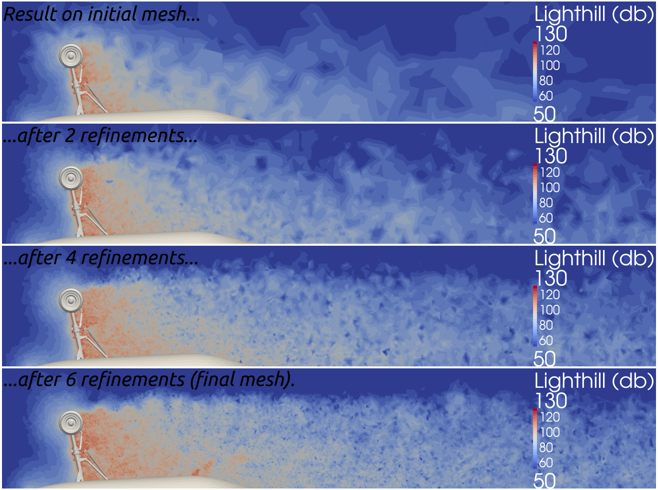
Mesh captures the relevant flow features...



ROYAL INSTITUTE OF TECHNOLOGY

Adaptive Mesh Refinement

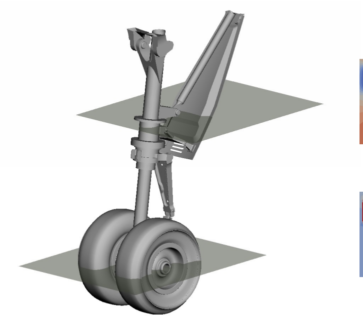
Solution on different meshes...



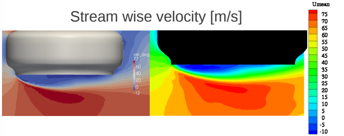
KTH VETENSKAP OCH KONST

ROYAL INSTITUTE OF TECHNOLOGY

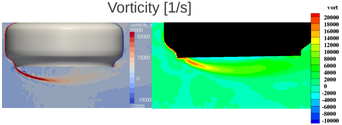
Benchmark results, BANC-II



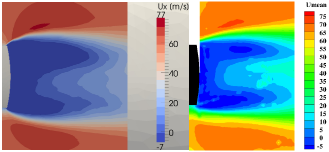
Stream wise velocity [m/s]



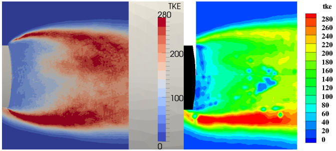
Vorticity [1/s]



Stream wise velocity [m/s]

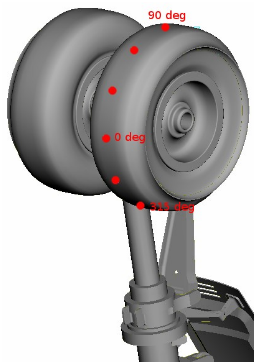


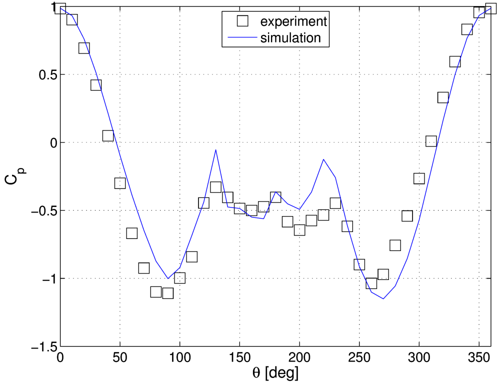
TKE [m2/s2]



In *all* figures: left, *sim*; right, *exp*.

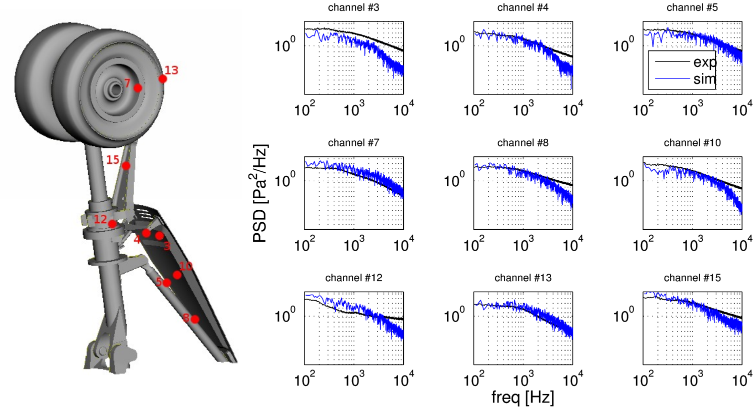
Benchmark results, BANC-II





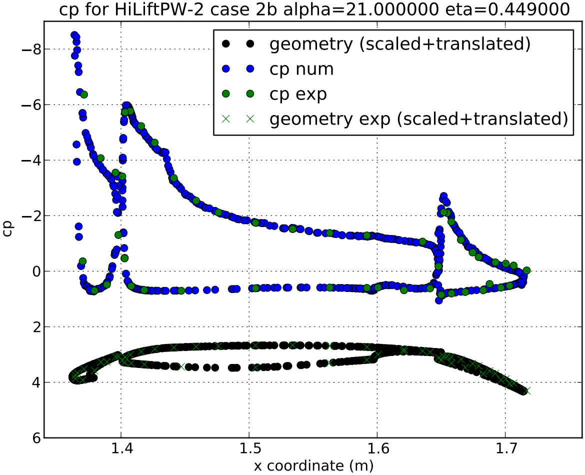
Mean static pressure coefficient distribution.

Benchmark results, BANC-II



Power Spectral Density unsteady pressure.

Benchmark results, HiLiPW-2



Mean static pressure coefficient distribution.

ROYAL INSTITUTE OF TECHNOLOGY

Why **ANSA**?

Enabling features

- Easy to clean-up geometries, even for new users.
- Batch mesh generation.
- Precise control of parameters (*e.g. leading edge curvature, growth rate, min-max cell sizes, quality*).
- High quality volume mesh (highly required in our framework for refinement).
- ...

⇒ *Very knowledgeable, efficient and helpful support! Thanks Vangelis!*

ROYAL INSTITUTE OF TECHNOLOGY

Unicorn and DOLFIN, open source

`http://launchpad.net/unicorn`

Acknowledgement

All initial meshes were generated with **ANSA** by Beta CAE Systems.

The code **Saaz** was used in "offline mode" for post-processing.[5]

Financial support from

- Swedish Foundation for Strategic Research
- European Research Council
- Swedish Research Council, Swedish Energy Agency

This work was performed on resources provided by the Swedish National Infrastructure for Computing (**SNIC**) at the Center for High-Performance Computing (**PDC**) at KTH.

[5]Alden King, Eric Arobone, Scott B. Baden and Sutanu Sarkar, *The Saaz Framework for Turbulent Flow Queries, 2011.*