



Use of gaming and affordable VR technology for the visualization of complex flow fields

Document Version

Other version

[Link to publication record in Manchester Research Explorer](#)

Citation for published version (APA):

Seu, E., Hewitt, S., & Margetts, L. (2018). *Use of gaming and affordable VR technology for the visualization of complex flow fields*. Abstract from 6th European Conference on Computational Mechanics (ECCM 6) 7th European Conference on Computational Fluid Dynamics (ECFD 7), Glasgow, United Kingdom.

Citing this paper

Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights

Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy

If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.





UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Eleonora Seu, Università degli Studi di Padova, Department of Industrial Engineering
Sam Hewitt, University of Manchester, School of Mechanical, Aerospace and Civil Engineering
Lee Margetts, University of Manchester, School of Mechanical, Aerospace and Civil Engineering

MANCHESTER
1824

The University of Manchester

USE OF GAMING AND AFFORDABLE VR TECHNOLOGY FOR THE VISUALIZATION OF COMPLEX FLOW FIELDS

Authors:

Eleonora Seu, Università degli Studi di Padova, Department of Industrial Engineering

Sam Hewitt, University of Manchester, School of Mechanical, Aerospace and Civil Engineering

Lee Margetts, University of Manchester, School of Mechanical, Aerospace and Civil Engineering

EQUIPMENT

A personal computer is used to:

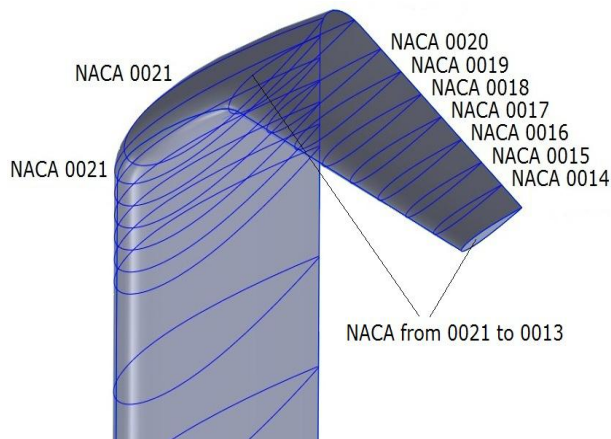
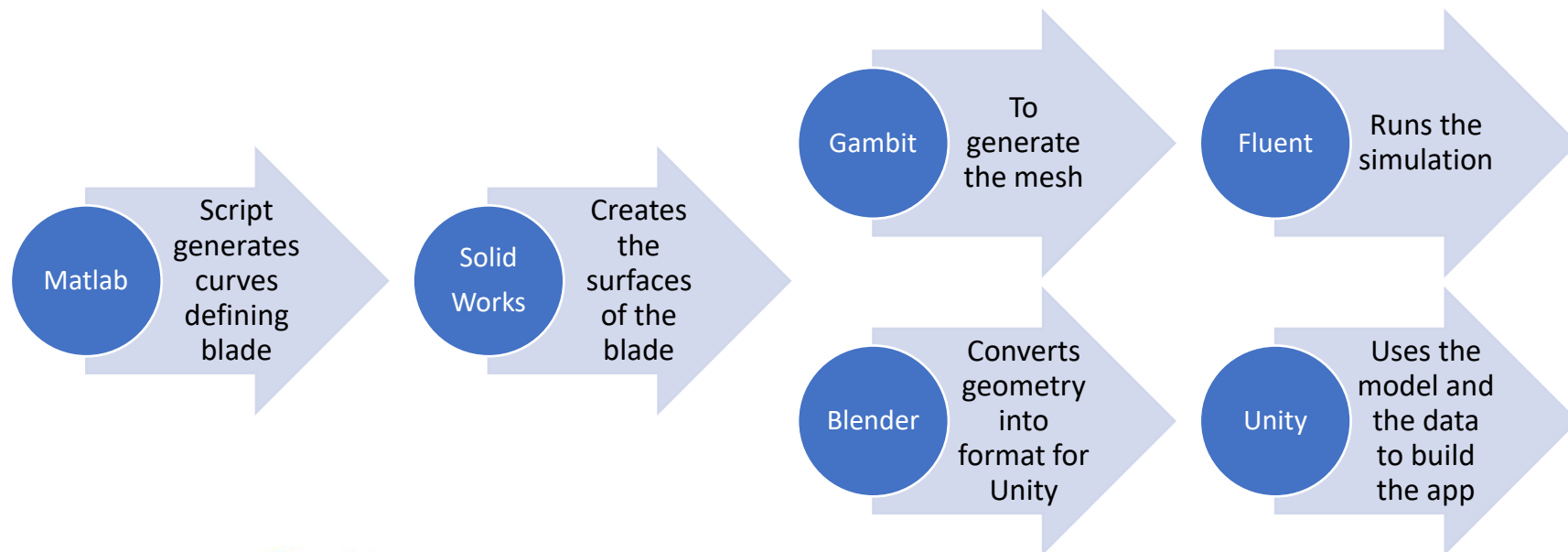
- Create the CAD model of the wind turbine blade
- Run the CFD simulation and collect the data
- Create the Android application that will be used to visualize the streamlines in VR (using Unity3D)



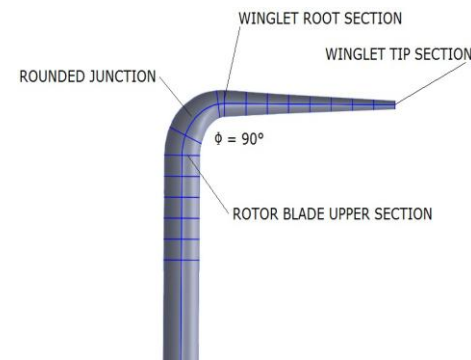
The Samsung Gear VR headset will allow the user to see the streamlines in VR. With the controller it is then possible to move around the simulation and select the turbine grade of rotation to visualize.

The phone, a Samsung Galaxy S6 will run the application

CREATION OF THE CAD MODEL



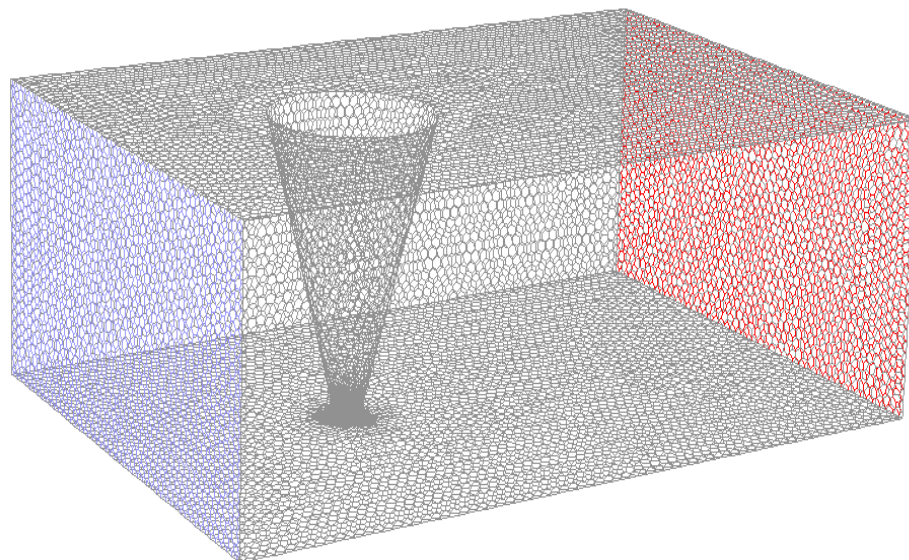
Winglet sections vary linearly from NACA 0021 to NACA 0013 from the winglet root station to the winglet tip station



SIMULATION MODEL

The overall computational domain is discretized into two macro volumes:

- a prismatic outer zone, covering the whole computational domain, presenting a cylindrical/conical opening centered on the turbine axis of revolution, identified as Fixed sub-grid;
- a cylindrical/conical inner zone, revolving at the same angular velocity ω of the turbine, identified as Rotating sub-grid.



$$\omega = 560 \text{ rpm}$$

Relative Specification		UDF	
Relative To Cell Zone	absolute	Zone Motion Function	none
Rotation-Axis Origin		Rotation-Axis Direction	
X (mm)	0 constant	X	0 constant
Y (mm)	0 constant	Y	0 constant
Z (mm)	0 constant	Z	1 constant
Rotational Velocity		Translational Velocity	
Speed (rad/s)	58.64306 constant	X (m/s)	0 constant
<input type="button" value="Copy To Frame Motion"/>		Y (m/s)	0 constant
		Z (m/s)	0 constant

Denomination	Value
R [m]	0.515
H [m]	1.716
Blade section	NACA 0021
Blade number [-]	1
c [m]	0.086
Location of blade centre of pressure	0.25 c
Location of blade-spoke connection	0.25 c

SIMULATION MODEL

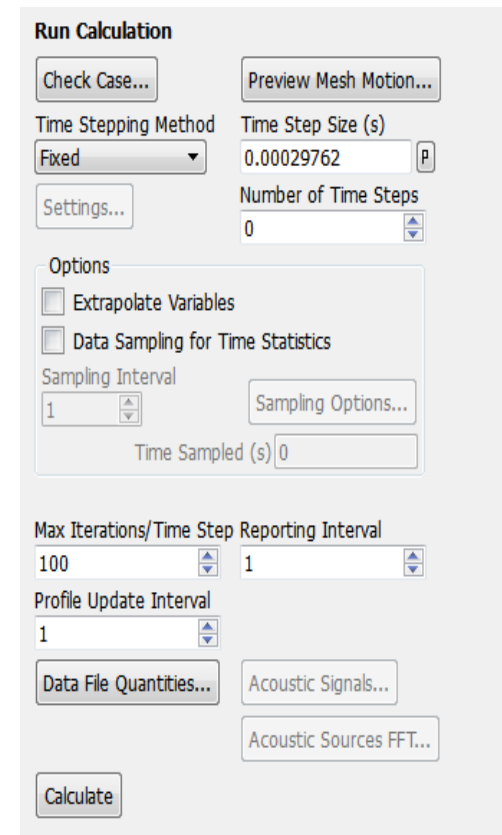
Boundary conditions:

- symmetry boundary condition for the equatorial plane
- freestream wind velocity $V_\infty = 9$ m/s at the domain inlet
- atmospheric pressure at the domain outlet

K- ω SST turbulence model is used

Denomination	Value
Pressure [-]	0.3
Density [-]	1.0
Body forces [-]	1.0
Momentum [-]	0.7
Turbulent kinetic energy [-]	0.8
Specific dissipation rate [-]	0.8
Turbulent viscosity [-]	1.0

Under-relaxation factors



Run Calculation

Check Case... Preview Mesh Motion...

Time Stepping Method: Fixed Time Step Size (s): 0.00029762

Settings... Number of Time Steps: 0

Options

Extrapolate Variables

Data Sampling for Time Statistics

Sampling Interval: 1 Sampling Options...

Time Sampled (s): 0

Max Iterations/Time Step: 100 Reporting Interval: 1

Profile Update Interval: 1

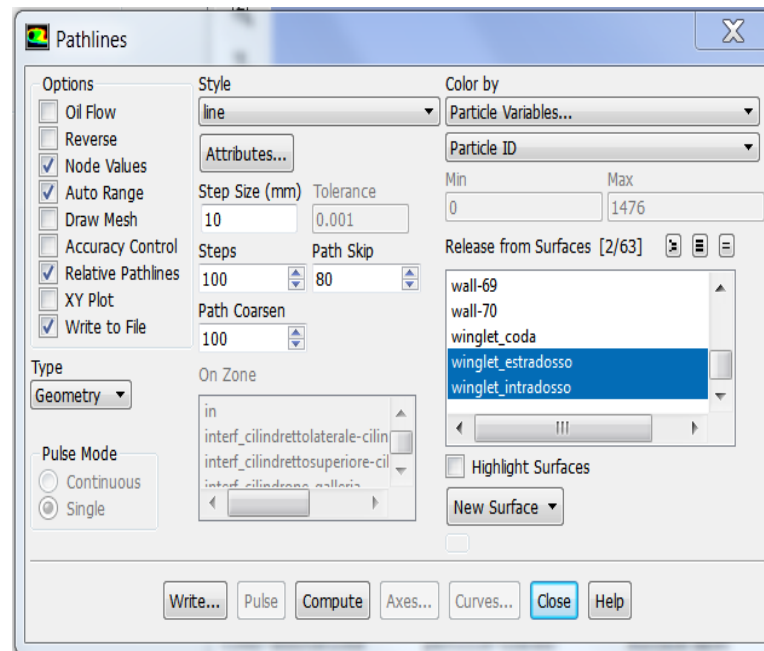
Data File Quantities... Acoustic Signals...

Acoustic Sources FFT...

Calculate

SIMULATION MODEL

Fluent allows user to export pathlines in different formats. In this case «Geometry» type is the easiest one to work with. This creates an .ibl file that contains particle paths in form of a curve. Other data, except the shape of the curve (e.g. pressure, velocity, etc.), are lost. This is done for every grade of rotation of the turbine, resulting in 360 .ibl files.

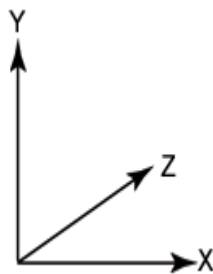
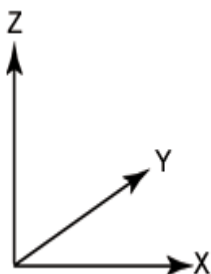


```
begin curve
0.0642322 0.49577 0.879477
0.0643292 0.495736 0.879464
0.0644502 0.495668 0.879395
0.0643746 0.495609 0.879366
end curve
```

The strings «begin curve» and «end curve» define the beginning and ending of each curve inside the file, and are therefore used to separate the file into an array of curves

IMPORTING THE DATA

Unity uses a left handed reference system, with gravity acting on the y axis, instead of the classic right hand system that is used by Fluent. Importing the data as it is would result in having the curves mirrored. This problem is easily overcome by inverting the second and third column while importing the data.



Fluent reference system: right handed Unity reference system: left handed

```
begin curve
0.0642322 0.49577 0.879477
0.0643292 0.495736 0.879464
0.0644502 0.495668 0.879395
0.0643746 0.495609 0.879366
end curve
```

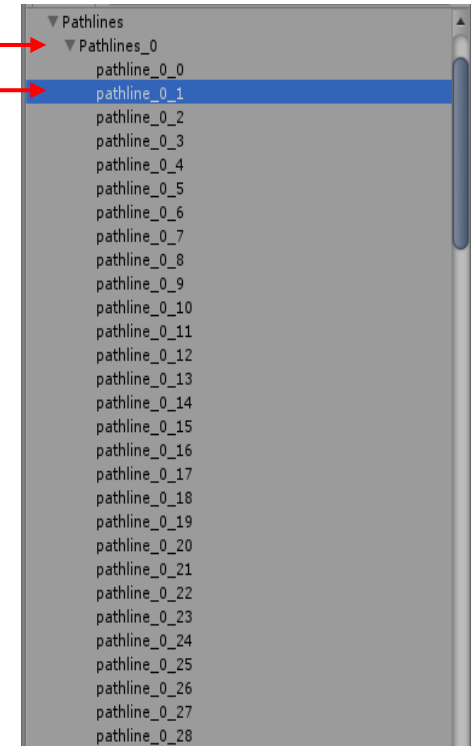
A script inside Unity is made to convert the text file into an array of vectors. Each array contains the points (x, y, z) of one curve.

VISUALIZING THE STREAM LINES

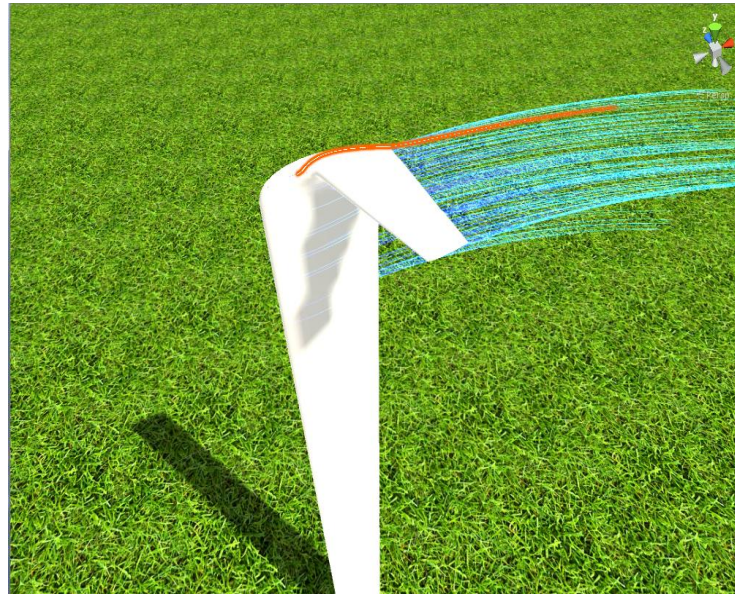
The construct used to visualize the streamlines is called “Line Renderer”. It’s a component that, attached to a GameObject, draws a line between 2 or more points. Since the line is always continuous, in case of completely separate lines it is necessary to use multiple GameObjects.

Empty GameObject that contain all the lines of that angle of rotation

Specific line



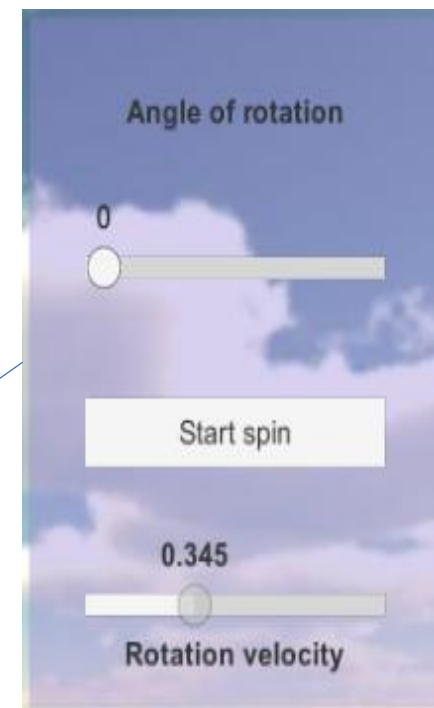
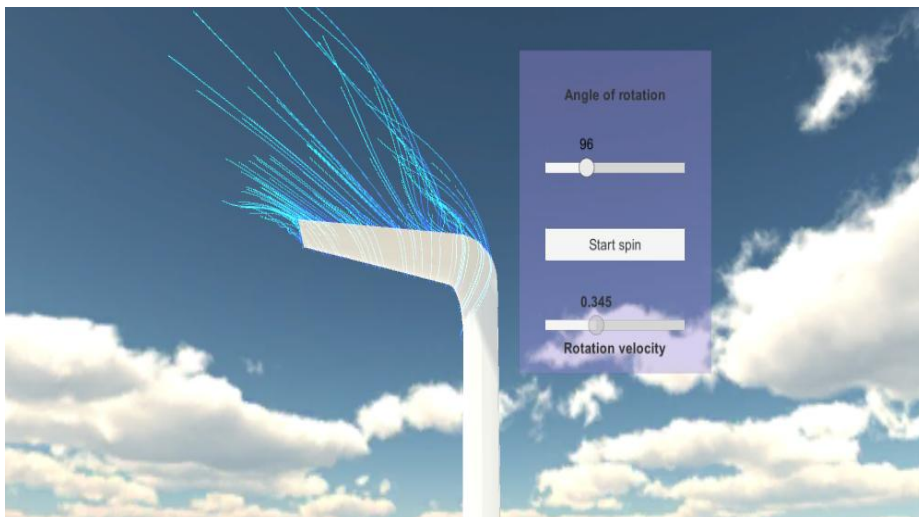
To keep the work organized, for each grade of rotation, an empty GameObject is created that is the parent. This contains all the GameObjects that represent the pathlines for that specific angle. During the visualization, the app will enable or disable the parent according to the angle of rotation of the turbine that is currently visualized.



USER INTERFACE

A simple user interface allows the user to rotate the turbine inside the app. Using the controller it is possible to select a specific angle to visualize, or start the spin of the blade setting the desired speed.

Using the controller it is also possible to move around the scene, getting closer or farther away from the blade and looking at the streamlines from different perspectives.





CONCLUSIONS

Problems:

- Time needed to collect the data: the CFD simulation is automatically saved every time step. Even so, to collect the data required for the VR app, it is necessary to open every time step of the solution and download the relative data.
- Amount of data: to avoid having too much data to import in the application, making it unavoidably slow, it is necessary to select only the data that is of most interest.

Future work will include:

- Visualization of pressures and other data on planes that can be moved around using the controller: to reduce the amount of data the best option appears to be the use of spatial data structures, such as the octree. In this way it should be possible to reduce the fluid domain from all the domain of the simulation to the areas that are of interest and the level of detail to a set amount.
- Implementing an app for the visualization of FSI: since in this case the mesh changes at every time step it cannot be imported just once using the standard method. It will be necessary to export the mesh every time using the software used to run the simulation.



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Eleonora Seu, Università degli Studi di Padova
Sam Hewitt, University of Manchester
Lee Margetts, University of Manchester



THANK YOU
FOR YOUR
ATTENTION

6th European Conference on Computational Mechanics (ECCM 6)
7th European Conference on Computational Fluid Dynamics (ECFD 7)
11 - 15 June 2018, Glasgow, UK