Shape Evolution in Computational Fluid Dynamics

Cristian Rendón
Universidad EAFIT
Medellín, Colombia
crendo11@eafit.edu.co

Mauricio Toro
Universidad EAFIT
Medellín, Colombia
mtorobe@eafit.edu.co

Keywords

B-Spline; Isogeometric analysis; Laminar Flow; Navier-Stokes equation; NURBS

1. GLOSSARY

- **Ω:** Rectangular orthogonal simulation domain with center $\in$ (0,0) and $x \in$ [-L,L] and $y \in$ [-D,D].
- **Γ:** PL simple closed curve $\in R^2$ (CCW with respect to Z) with center of gravity $\in$ (0,0).
- **F_L:** Lift force, generated by the difference of pressures across a object and acting perpendicular to flow direction.
- **F_D:** Drag force, generated by the relative movement between a object and a fluid and acting parallel to flow direction.

2. ACRONYMS

- **CAD:** Computer Aided Design.
- **CAM:** Computer Aided Manufacturing.
- **CAE:** Computer Aided Engineering.
- **CFD:** Computational Fluid Dynamics

3. INTRODUCTION

Among the disciplines of computer sciences, CAD/CAM/CAE is one of the most useful in the engineering due to its possibility to create simulations of the behaviour of a object under certain conditions without the complication of materialize it.

One of the advantages of CAD packages is the possibility to emulate fluid flow conditions in an object. This information can be used to change the shape of it under a optimally criteria.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

4. PROBLEM SPECIFICATION

Given:

- A incompressible Newtonian fluid $\in R^2$.
- Constant density and viscosity.
- Steady laminar flow.
- $v(x = \text{-L}) = (V_\infty, 0)$.
- Γ_0: Initial object, submerged in the fluid domain Ω.
- A target F_L.

Goal:

- To obtain a Γ_f by modifying its shape such that generates the target F_L and minimizes F_D.

5. RELATED WORKS

5.1 Flow Complex

The goal of this algorithm is to reconstruct a surface from a given a set of points, but they are constrained to lie on the surface of some solid [1].

5.2 Isogeometric shape optimization in fluid mechanics

This work uses isogeometric shape optimization to implement it in many applications, like body with uniform pressure, minimal drag or pipes with minimal pressure drop. The goal is thus to minimize the cost function, which depends on the velocity u, pressure p, and boundary domain Γ'.[2].

The method has two phases: Initialization and Optimization.

- **Initialization:** In the initialization phase, the algorithm uses B-spline a and NURBS for the geometry and parametrization of the body. In the flow analysis the Navier-Stokes equation and the incompressibility condition are solved.
- **Optimization:** In the optimization phase an iterative, gradient-based, non-linear optimizer is employed to reduce the cost function.

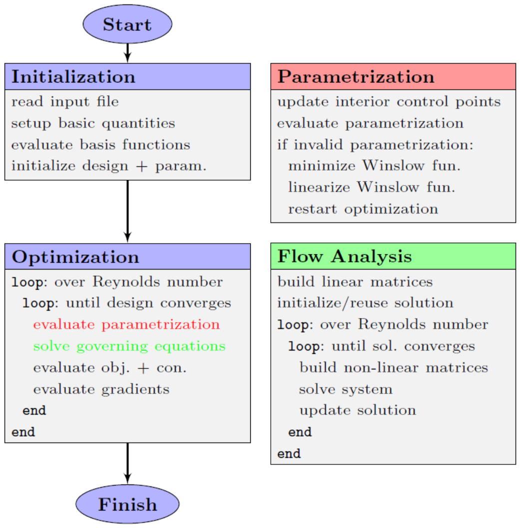

Figure 1: Flow chart for the optimization process (left) with details of the parametrization and analysis procedures (right) [2].

6. EVOLUTION FLOW

Figure 6 presents the flow diagram in the evolution of the shape.

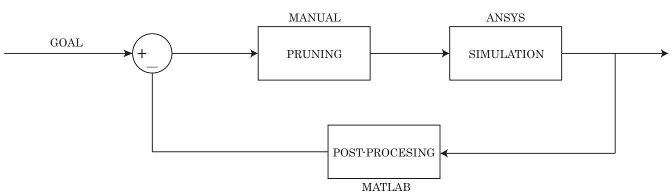

Figure 2: Evolution diagram of the object, based in the calssic control diagram

6.1 Experimental Values

In first iteration we have a Γ_0 with initial rectangular shape submitted to the flow.

- The chosen fluid is air with density $\rho = 1.225 kg/m^3$.
- Dimensions for Γ are $L = 35m$ and $D = 30m$.
- Magnitude of velocity $V_\infty = 80m/s$.
- Steady state time.

6.2 Simulation results

After running the simulation, there is available data of the velocity vector field, pressure scalar map, particles path to export for the post processing. Figure

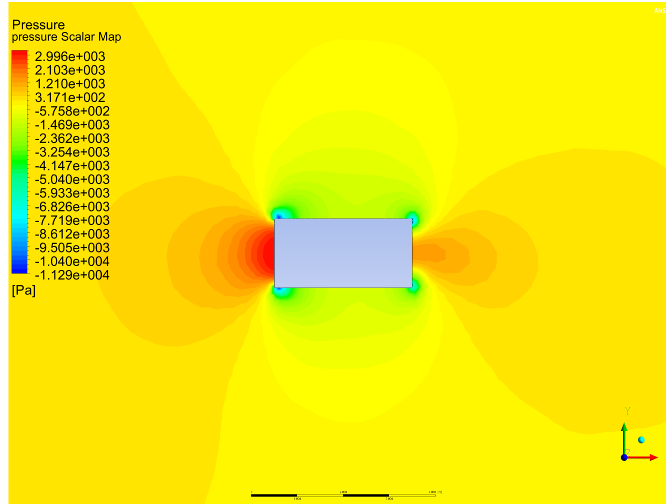

Figure 3: Pressure scalar map for initial object Γ_0

As shown in figure 6.2, due to the symmetry of pressures w.r. x axis there is not initial F_L.

7. MATLAB DATA STRUCTURES

7.1 Particles path

Table 3 shows the way that the path curves are exported with coordinates x_{ij} where j represents the j-th point of curve i.

Table 1:

begin curve		
x_{11}	y_{11}	z_{11}
x_{12}	y_{12}	z_{12}
x_{1n}	y_{1n}	z_{1n}
end curve		
begin curve		
x_{21}	y_{21}	z_{21}
x_{22}	y_{22}	z_{22}
x_{2n}	y_{2n}	z_{2n}

Table 2:

x_{11}	y_{11}	z_{11}
x_{12}	y_{12}	z_{12}
x_{1n}	y_{1n}	z_{1n}
inf	inf	inf
x_{21}	y_{21}	z_{21}
x_{22}	y_{22}	z_{22}
x_{2n}	y_{2n}	z_{2n}

Table 3: Manual replacing of the headers.

The data of table 2 is read nad sved in a matrix, then a loop is implemented to divide this information for each cuve in a Structure Array.

7.2 Structure array

Structures to storage CAD data are very simple, in this case the use of structure array from Matlab result very handy for the classification of path lines.

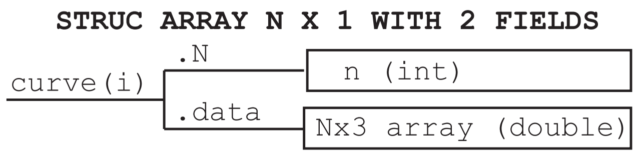

Figure 4: Diagrm of the path data saved in Matlab where .N gives the i-th curve and .data gives the matrix with the geometry of the curve [?]

This structure allows the insertion of new data of any type which is very useful for the post processing.

7.3 Execution times

Table 4 presents the execution times for the reading of the txt file, inserting in the structure and search in it. with the help of the tool "Run and time" of Matlab (fig. 7.3).

	Read	**Insert**	**Search**
n = 50	0.06	<0.01	<0.01
n = 500	0.6	<0.01	<0.01
n = 5000	6.3	<0.01	<0.01
Complexity	O(nxm)	O(n)	O(1)

Table 4: Execution times and complexity of the algorithm where n represents the numer of curves and n the number of points per curve

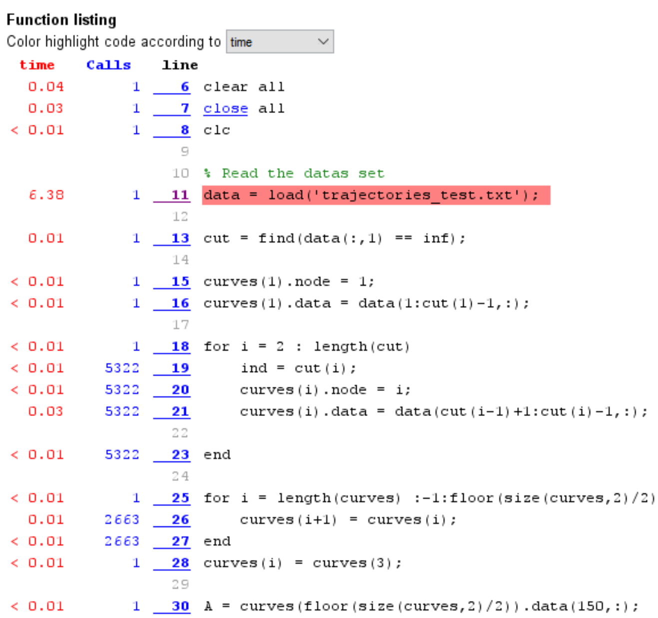

Figure 5: Run and time tool from Matlab

8. REFERENCES

[1] J. G. . M. John. The flow complex: A data structure for geometric modeling. *ScienceDirect*, 2007. doi:10.1016/j.comgeo.2007.01.002.

[2] Mathworks. Structures, apr 2018.

[3] J. Nortoft, Peter; Gravesen. Isogeometric shape optimization in fluid mechanics. *Structural and Multidisciplinary Optimization*, 2013. DOI: 10.1007/s00158-013-0931-8.