Scuola Politecnica DIME

Ingegneria Meccanica - Energia e Aeronautica

Anno scolastico 2017-2018

Fluidodinamica Avanzata

Aircraft S-shaped duct geometry optimization

Professor
Jan Pralits
Supervisor
Joel Guerrero

Student
Valerio Lanata

October 1, 2018

1 Introduction

This project was developed inside the Advanced Fluid dynamics course to complete the practical session. The aim was to optimize the shape of an aircraft S-shaped duct, the analysis was performed via the open-source codes OpenFOAM and Dakota, using a multi-objective optimization.

2 Optimization Method

2.1 Multi-objective optimization

Multi-objective optimization is an area of multiple criteria decision making, that is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. Multi-objective optimization is often applied in many fields where optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives. For a nontrivial multi-objective optimization problem, no single solution exists that simultaneously optimizes each objective. In that case, the objective functions are said to be conflicting, and there exists a (possibly infinite) number of Pareto optimal solutions. A solution is called non-dominated if none of the objective functions can be improved in value without degrading some of the other objective values: without additional subjective preference information, all Pareto optimal solutions are considered equally good. The result of a MOO is the approximated Pareto front at the end of the optimization, which differ from the true front that could be reached in the limit [1].

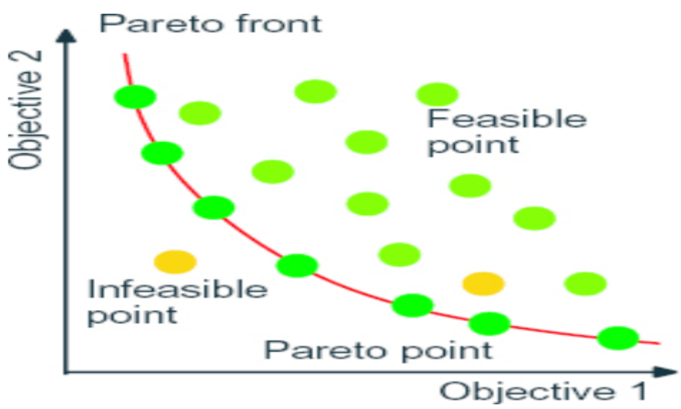


Figure 1: Example of pareto frontier

2.2 Evolutionary and genetics algorithms

Evolutionary Algorithms (EA) are based on natural evolution systems; each iteration is called, generation, the solutions are named individuals, the whole set of solutions is the population and the goal functions are named fitness. The method starts with a random generation, called Generation 0, then all solutions objectives are calculated and finally a fitness value is assign to each individual. All individuals take part in the creation of the new generation, but as in nature, those with the best fitness values have greater chance to survive during the evolution. Generation after generation, the algorithms tends to get closer to the real

Pareto Front and, theoretically, the last solution should lie on the Front. A genetic algorithm can act in three different ways to generate a new set of solution, applying Selection, Cross-Over and Mutation operators.

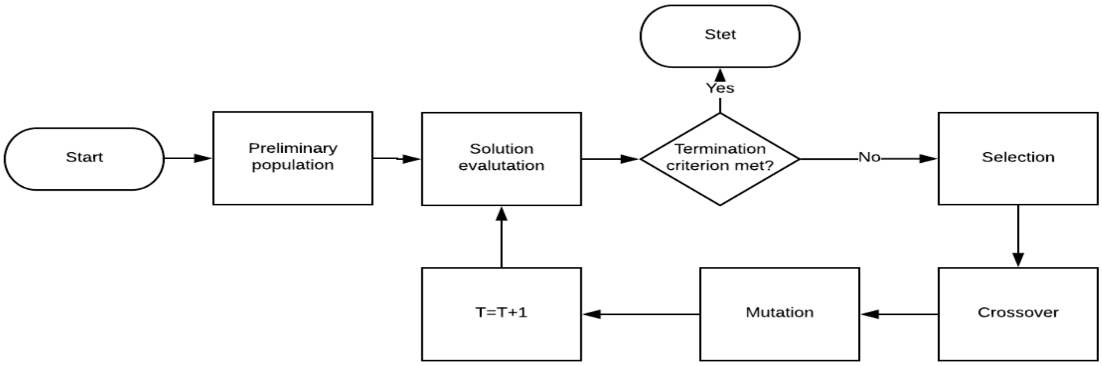


Figure 2: Flux diagram of optimization

Selection: this operator randomly chooses two solutions, the parents, which are used by the next operators to generate two new individuals, called offsprings. Actually this choice depends on the fitness value assigned to each solution: those with a better fitness have a greater chance to be chosen. In theory, this method allows to create a better generation than the previous one, also maintaining the same size. There are different types of selection operators, as proportionate, ranking or tournament.

Cross-Over: together with the mutation, the cross-over creates the offsprings randomly exchanging part of parents' information; the genetic material is not lost, just recombined. There are different methods for cross-over operation that can be found in literature, like single-point, double-point or non-homologous.

Mutation: this statistical operator maintains the genetic diversity of a generation; for each offspring, a small part of the information is randomly modified. It permits the code to escape from local minima, but it does not guarantee the offspring to be better than its parent.
These three methods are performed in sequence until a new generation is created that will be evaluated by the calculation model, hence the loop has been closed.

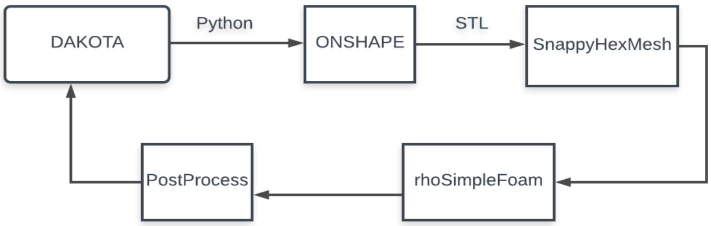
2.2.1 Genetic algorithms

In GAs the cross-over operator assumes particular importance: the method starts with initialization and evaluation of a m individual population. After, a generation n, m new offspring indidivuals are created as follows and select a couple of parents and apply the cross-over giving birth to two children where the chromosomes of the parents is swapped. The new generation is evaluated and replaces completely the old one. For this project every generation is created for 36 individuals for every 5 generations.

3 Utilized tools

3.1 Dakota and OpenFOAM

Dakota and OpenFOAM are the principal applications utilized, both of them are opensource. The large amount of optimization methods preinstalled in Dakota's library allows the user to perform an optimization study, furthermore Dakota can interface itself to every codes. The optimization loop starts with Dakota that modifying the geometry saved on Onshape and exporting it as an STL file. OpenFOAM processes the new geometry, meshes it and then run the rhoSimpleFoam solver. As postprocessing the values of the 2 objective functions of the last 30 iterations are averaged to prevent errors from fluctuations that may occur in the simulation.



4 Case preparation

4.1 S-shaped duct

An S-shaped (shortened S-duct) is a common component in both military and civil airplanes or in automotive industry. In the aircraft application is used to supply the core engine with air at a pressure level higher than the external one.

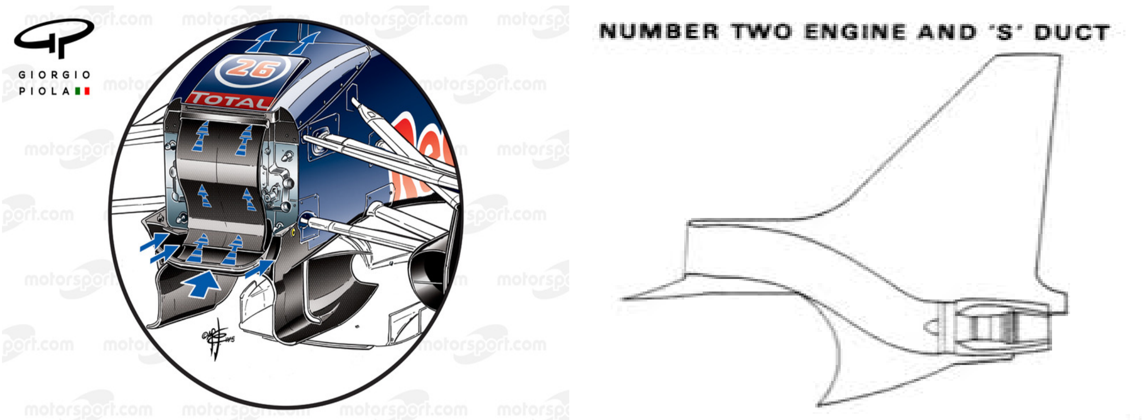


Figure 3: Automotive (left) and aircraft (right) S-ducts

Because of the diffusion happening inside itself, the S-duct is inclined to flow separation and great total pressure losses. These two negative effects can be reduced via a proper design [2].

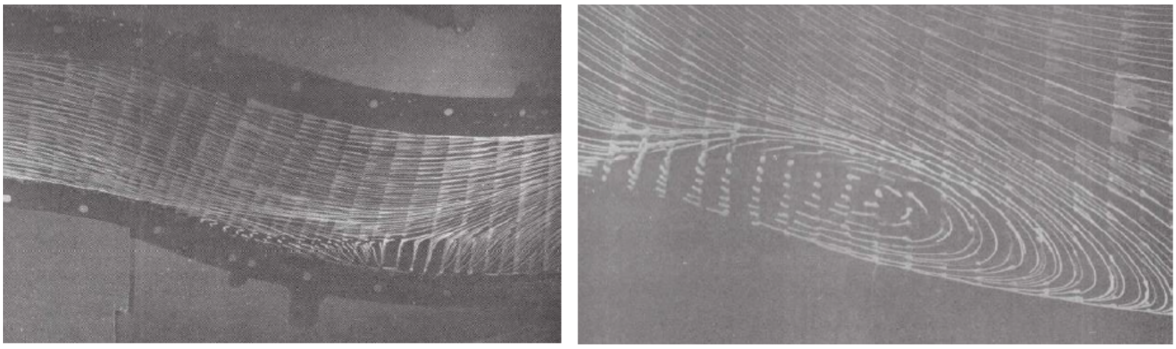


Figure 4: Stall and recirculation bubble inside the ONERA S-duct

The starting geometry was based one the one presented by Delot in 2006, designed at ONERA, French Center of Aerospace Research, that is provided to the CAD by an analytical expression.

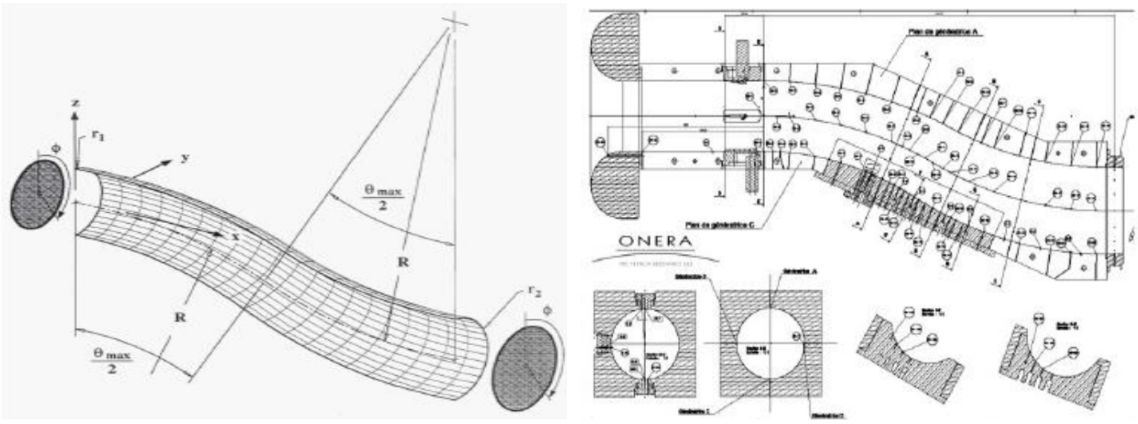


Figure 5: Onera analytical shape(left) and test rig (right)

4.2 Geometry parameterization

In this project the optimization was carried on the geometry of the duct, keeping the external conditions constant and subsonic. To manipulate the geometry a spline curve with 7 control points has been used as the spine of the entire geometry: the 2 coordinates of the 3 central control points were the input variables for the optimization loop. Subsequently from the central spine the outer area was created.

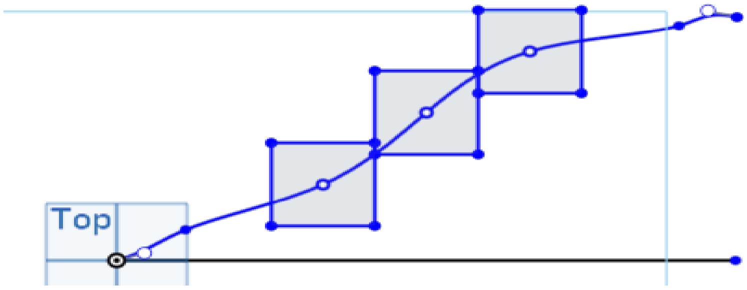


Figure 6: Example of base spline curve and variables boundaries

4.3 Simulation Parameters

The simulation was conduced in the conditions reported in table 1, to compare the CFD base's results to Delot's experimental data, [2] and [3]. From the boundary condition it is calculated that the inlet Mach number is 0,6: the incompressible flow assumption wouldn't modelize the real flow in a good manner.

Boundary Conditions	
Parameter	value
Inlet total pressure	88,744 Pa
Inlet static pressure	69,575 Pa
Inlet total temperature	286,2 K
Outlet static pressure	78,982 Pa

Table 1: Boundary conditions used for this project

4.4 CFD Setup

As a result of the Boundary conditions a steady, 2D, compressible, adiabatic flow simulation has been carried on. The CFD calculus simulated the experimental setup reported in figure 7.

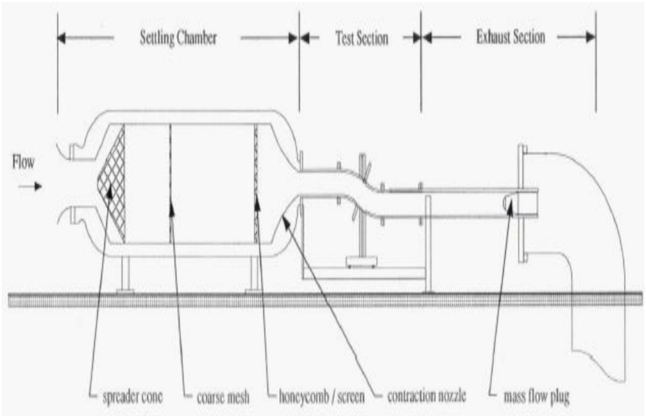


Figure 7: Experimental test rig

The OpenFOAM solver chosen was rhoSimpleFoam with the SIMPLE pressure-velocity coupling. As turbulence model k$\omega - SST$ was selected for its behaviour in flows with strong separation regions as the flow in the S-duct [4].
The mesh was created by snappyHexMesh and then refined at wall by the function refineWallLayer; the final number of cells was around 300'000.The duct presents an upstream straight part of $4D_1$ and a downstream straight part of $3D_2$: this was done to ensure uniform inlet conditions and to guarantee that the outlet conditions did not have any influence on the upstream flow. To obtain a result similar to the Delot's experimental data the number of iterations of the solver was 500. After the iteration n° 500 the U_x reaches its asymptotic value, as shown in Fig.8: a larger number of iterations wouldn't affect the simulation output.

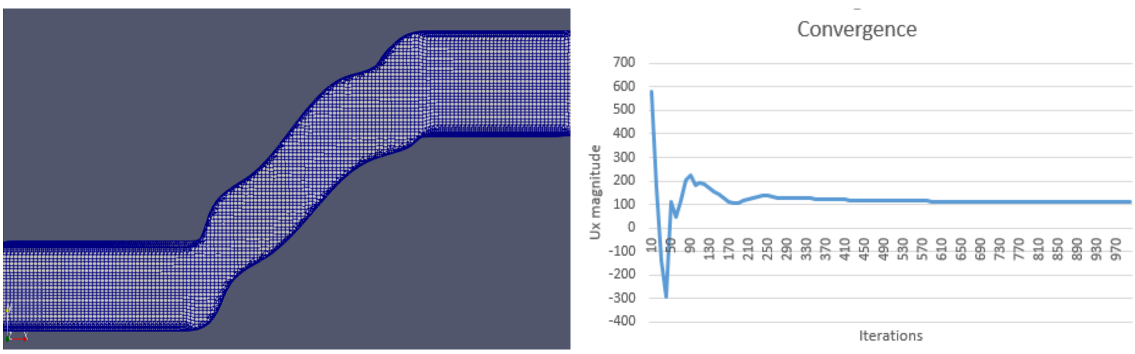


Figure 8: Example of mesh (left) and Ux magnitude in a probe against the iteration number (right)

4.5 Optimization Setup

The Dakota suite has been used because it's opensource and it can interface with every black box solver as OpenFOAM or Onshape, using a simple python script. For the MOGA optimization 2 objective functions have been used: the total pressure loss coefficient Cp_t (eq.:1) and a distorsion factor DC (eq.:2) modified from the Bissinger and Breuer[5] original form. The distortion coefficient is calculated as the difference between average total pressure of the whole plane and the minimum total pressure calculated in one of the

12 horizontal sectors in which the outlet area is divided, calculated via 12 probes, using the local density and velocity, normalized with the inlet dynamic pressure.

$$Cp_t = \frac{Cp_{t\infty} - Cp_{tlocale}}{q_\infty} \tag{1}$$

$$DC = \frac{Cp_{tavg} - Cp_{minlocale}}{q_\infty} \tag{2}$$

5 Results

In the Figure 9 is reported the pareto front with the best solution for DC, for Cp_t and for the trade off.

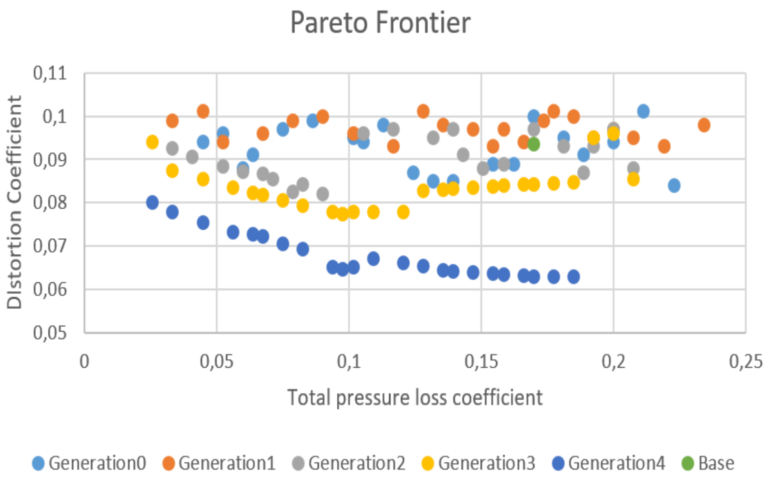


Figure 9: Pareto frontier of the project

In table 2 is reported the percentage improvement from the original geometry.

	minimum DC condition	minimum Cp_t condition	trade-off
DC	-20.39	-13.56	-16.397
Cp_t	+1.734	-13.269	-7.2397

Table 2: Percentage differnce from datum geometry

The next 2 figures show the original outlet area total pressure distribution and the trade-off one.

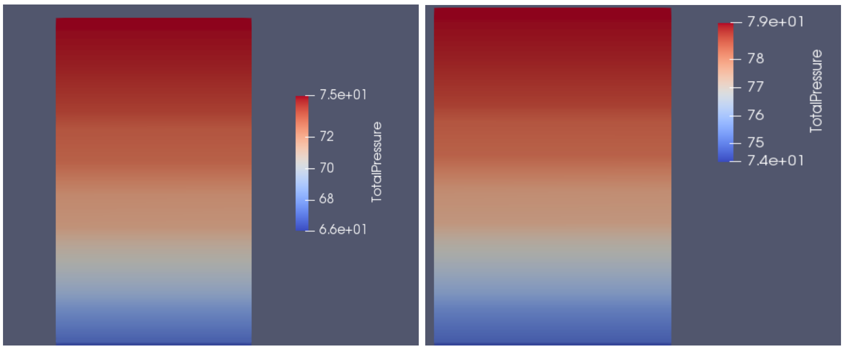


Figure 10: Original outlet area (left) and trade-off one (right)

6 Final discussion

The optimization loop granted a good improvement either for DC and Cp_t both; but the Genetic algorithm, to be successful, needed a lot of cfd evaluations. This was possible, in a short time and with a low budget pc, only because of the 2D problem: to perform a 3D optimization, a parametric study and a surrogate model could be more suitable.

7 Bibliography

[1] Pralits J., dispense Fluidodinamica avanzata
[2] Delot A., A comparison of several CFD codes with experimental data in a diffusing s-duct, AIAA, 2013.
[3] Rigobello A., A Multi-objective shape optimization of an S-Duct intake through NSGA-II genetic algorithm, http://tesi.cab.unipd.it/53996/1/Rigobello$_{A}urora_{t}esi.pdf$
[4]$GuerreroJ.$, ”$Module8advancedphysics$”; $UNIGE$
[5]$BissingerN.C.andBreuerT.$, $Basicprinciples-gasturbinecompatibility-intakeaerodynamicsaspects$, 2010