Project n°5
Interpolation and integration methods / Cubic splines and surface interpolation

Group n°3 - Team n°1
Responsible: lpfleger
Secretary: pgaulier
Programmers: ndehaiesdit, paubreton, lhing

Introduction :

The aim of this project was to implement interpolation and integration methods in order to build a model of the pressure surrounding the cross section of an aircraft wing. To do so, the air foil was first refined in a sufficiently smooth curve, and then the pressure map was computed.

Air foil refinement

A complete set of couples $(x_i, y_i)_{\in \mathbb{N}}$ was given, describing positions on the upper and the lower surface of an air plane wing. To achieve this project, it was needed to be able to interpolate those positions. To do so, cubic splines were used, which were giving parts of third degree polynomials on each fragment of curve.

Understanding

Chapter 3.3 of *Numerical Recipes* gave information and explanations on cubic spline interpolation and on how to implement it. Indeed, cubic splines are a useful tool of interpolation that preserves the continuity of the two first derivatives of the function, within an interval and its boundaries. Next formula was provided by *Numerical Recipes*:

$$y = Ay_j + By_{j+1} + Cy_j^{''} + Dy_{j+1}^{''} \tag{1}$$

In the equation (1), A, B, C and D are coefficients calculated by algorithms given in *Numerical Recipes*. $y_j^{''}$ and $y_{j+1}^{''}$ are derivatives on specific points, one following the other. With this equation, it was possible to obtain the interpolation polynomial on each point of the curve.

Implementing algorithms

Numerical recipes provided also two algorithms that calculatesa cubic spline given a function. The first one, *spline*, which is used only once per set of points, was used to compute and obtain second derivative on each point of the set.
This algorithm was first needing derivatives of the two first points, and the two last points. However, the algorithm was made to use a "natural" mode if derivatives values are higher than $0.99e30$. This mode considers $y^{''}(x_0) = y^{''}(x_n) = 0$. It provided a simpler way of calculating and was sufficient. This function returns an array containing second derivatives on each point.

Second algorithm, called *splint* for *spline interpolation* is the one which was doing the interpolation between each point. It calculated the cubic-spline interpolated value of a given x.

In order to obtain a smooth interpolation curve, *splint* function is called for a certain quantity of points, for instance 200, equally distributed between 0.0 and 1.0.

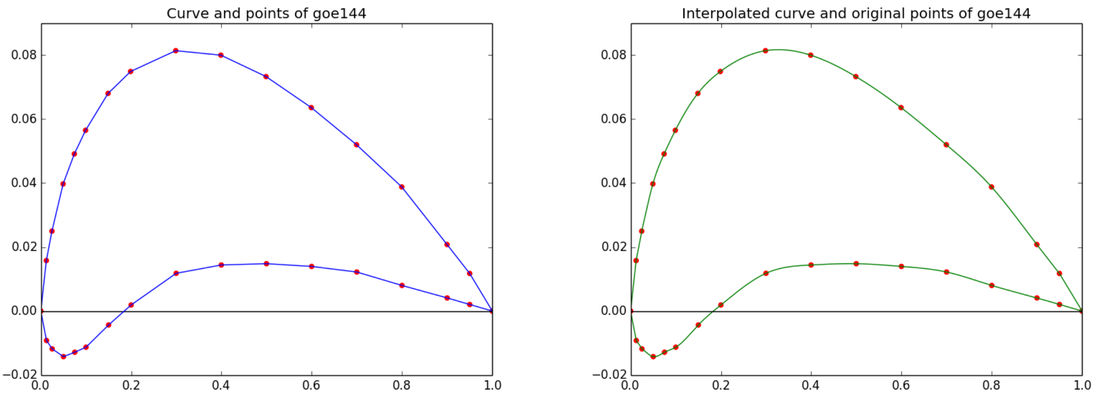

Figure 1: (2 figures) Original curve and points, and interpolated curve with original points

On the figure above are displayed the first curve defined from the set of points extracted from a data file, and the interpolated curve with the same set of points. Interpolation gives a smoother and continuous curve.

For the next parts of this report, a function returning a lambda expression was created in order to compute values of third degree polynomial in certain points.

Computing the length of plane curves

The approximation of the pressure on each side of the foil requires the length of the function f's spline's graph that can be obtained with the formula:

$$L([a,b]) = \int_a^b \sqrt{1+(f'(x))^2}dx \tag{2}$$

In order to compute this integral, it is necessary to implement at least one integration method. Since there are a lot of such methods and that some are faster and more accurate than others, it should be able to chose which one should be used among several, and thus more than one must be implemented.

In this project, it has been chosen to implement four methods for numerical integration:

- The rectangle method: This method consists in splitting the interval $[a,b]$ in n subdivisions $([x_i,x_{i+1}])_{i\in[\![1,n-1]\!]}$ of width $h=\frac{b-a}{n}$ and calculating the area of the rectangle with either the value on the left $f(x_i)$ or the value on the right $f(x_{i+1})$. Then summing those values and multiplying the result with h gives an approximation of the area underneath the curve of f.

$$\int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f(x_i) \qquad \int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f(x_{i+1})$$

- The midpoint method: This method is very similar to the rectangle method, except that the value of f that is considered is the value at the exact middle point of the sub interval $[x_i, x_{i+1}]$.

$$\int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f(\frac{x_i + x_{i+1}}{2})$$

- The trapezoid rule: This method is an upgrade to the rectangle method. Rather than computing the area of a rectangle, it calculates the area of the trapezium between x_i and x_{i+1} as if the function was linear.

$$\int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} \frac{f(x_i) + f(x_{i+1})}{2}$$

- Simpson's rule: This method uses a Lagrangian polynomial interpolation to approximate the function f by a parabol, for which it is easier to calculate the integral. It has been proven that this method is more effective on "smaller" intervals, so it is possible to use a splitted interval like in the other methods.

$$\int_a^b f(x)dx \approx \frac{h}{3}[f(a) + 2\sum_{i=1}^{\frac{n}{2}-1} f(x_{2i}) + 4\sum_{i=1}^{\frac{n}{2}} f(x_{2i-1}) + f(b)]$$

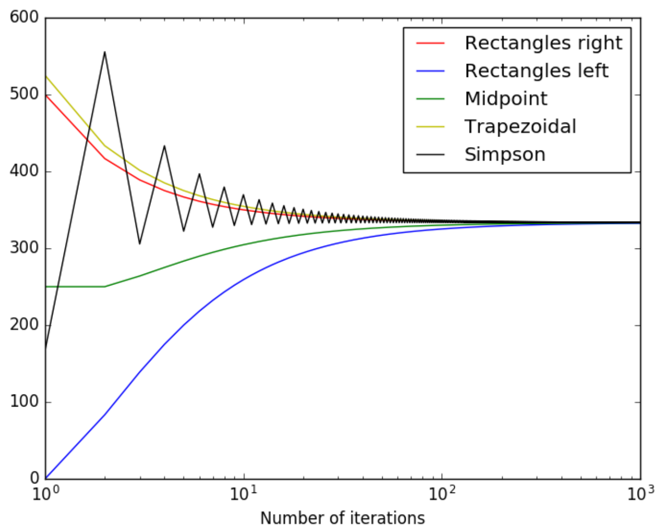

Figure 2: Comparison of the different methods on the function $x \mapsto x^2$ on $[0, 10]$

Each curve represents the result of one specific method in function of the number of values of the function f known, and thus the number of iterations. It is clear that the rectangle methods are less accurate than the others, and take more iterations to give a satisfying approximation of the integral. On this graph, the most accurate method seems to be Simpson's rule though it has a oscillatory behaviour, it converges faster towards the real value. As a consequence, it is the method that is going to be used for the rest of the project.

Modelling the airflow

In order to model the airflow and to draw the pressure map, some approximations were made. First, the airflow was supposed to be laminar, that is to say that the latter could be split into non-intersecting slices.Then, near the airfoil, the air moves along a curve very similar to the airfoil, whereas when it is situated further, it moves horizontally, without disturbance. The airflow was supposed to be disturbed by the wing only in a vertical interval $[3h_{min};3h_{max}]$ where h_{min} and h_{max} were respectively the minimal and the maximal heights of the airfoil. Out of this interval, the air flowed in a rectilinear way. The family of curves describing the airflow above the wing were determined by the following equations:

$$y = f_\lambda(x) = (1-\lambda)f(x) + \lambda * 3h_{max} \qquad \forall\lambda \in [0;1] \tag{3}$$

Precedent functions made were used to have a lambda expression and also to provide the equation above the function f. This equation has been used to model airflow on the upper and lower part of the wing.

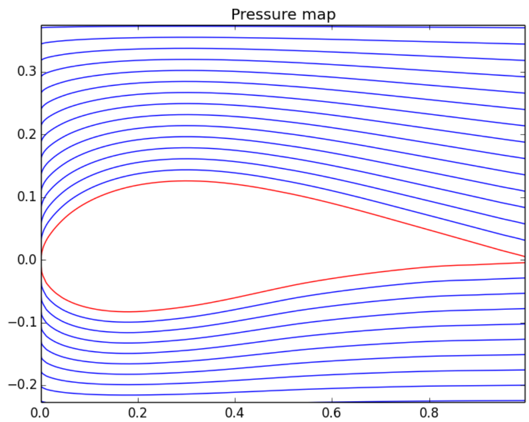

Figure 3: Laminar flowing of the air

Calculating pressure

The air is supposed to flow along surfaces on the wing. To calculate pressure on each side, air was considered as a moving fluid, in order to apply **Bernoulli law** :

$$P = \underbrace{P_s}_{\text{static pressure}} + \underbrace{P_d}_{\text{dynamic pressure}} \quad \text{where} \quad P_d = \frac{1}{2}\underbrace{\rho}_{\text{density of the air}} \times \underbrace{V^2}_{\text{speed}} \tag{4}$$

To simplify calculations, P_s the static pressure was supposed to be constant. Pressure variations were then linked to variations speed of the air. However, air speed is not known. Because it is laminar, the time the air takes to pass through the zone is the same everywhere. So, it is possible to compute air speed as a simple function of the length of the wing.

$$V = \int_a^b \sqrt{1+(f'(x))^2}dx \tag{5}$$

The force induced by air is slightly due to static pressure. Indeed, the force is normally weaker on the upper part of the wing than on the lower part. It is this difference which induces the force sustaining the plane in the air during flight. In order to obtain a colored map of pressure, a pixel map was created. Each pixel on the image corresponds to the value of the length of the upper curve, relatively to pixel's position. *Python pyplot*'s *cmap* permitted to draw accordingly to a map of colors.

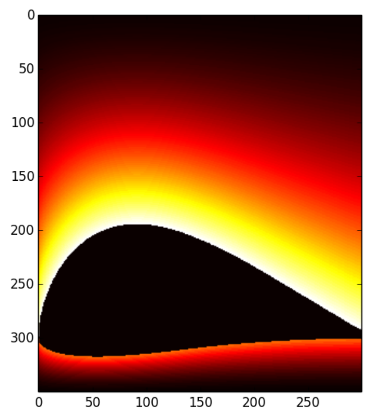

Figure 4: Colored pressure map

Conclusion

In this project, different methods to interpolate and to integrate were developed in order to solve a precise problem. These methods can be generic and also applied to many other fields. It was an interesting project to do. We "discovered" the "mathematics" behind air plane wings.

References

1. Press, William H., Saul A. Teukolsky, William T. Vetterling, and Brian P. Flanner. *Numerical Recipes: The Art of Scientific Computing*. Cambridge: Cambridge University Press, 1986.

2. "Spline". Wikipedia. https://en.wikipedia.org/wiki/Spline.