

Numerical Study of the Aerodynamic Characteristics of a Multi-Element Airfoil NACA 23012

Asya Gabbasa, Badih Jawad, Liping Liu and Selin Arslan
Lawrence Technological Univ.

ABSTRACT

This work aims to numerically investigate the aerodynamic characteristics of a multi-element airfoil NACA 23012. The investigation was conducted through Computational Fluid Dynamics (CFD), using ANSYS FLUENT software. The Navier-Stokes equations were solved for turbulent, incompressible flow using k-epsilon model and SIMPLE algorithm. The study was carried out for both take-off / landing conditions and the results were compared to experimental data of the NACA 23012 from wind tunnel tests. The experimental and computational results for drag and lift coefficients match effectively up to pre-stall attack angles. The pressure coefficients, velocity distribution, and wall Y^+ data were presented for different angles of attack (0 deg, 4 deg, and 8 deg). The CFD analysis could help acquire a closer and detailed understanding of airfoil performance, which is usually not easy through normal experimentation. Also, the simulation work could predict the design outcome under various circumstances and help people modify the design for improved or optimized performance.

CITATION: Gabbasa, A., Jawad, B., Liu, L. and Arslan, S., "Numerical Study of the Aerodynamic Characteristics of a Multi-Element Airfoil NACA 23012," *SAE Int. J. Aerosp.* 6(1):2013, doi: 10.4271/2013-01-1410.

INTRODUCTION

Computationally expensive simulation codes based on mathematical models of a relevant system are in wide-spread use throughout the engineering industry. For instance, in the field of computational fluid dynamics (CFD) a single evaluation of a model may take several hours of computer run time. With the help of modern computer facilities, numerous complicated and mathematically intensive problems are now readily solved. The solutions of the Reynolds-averaged Navier-Stokes (RANS) equations in the application of CFD to aircraft design, involve challenges in the following three areas:

Computational Efficiency: There has to be a reduction in the computing time required to achieve appropriately resolved solutions. This is an urgent need in aerodynamic and multidisciplinary design optimization, as an outcome of the trend towards an integrated product and process development environment. For complete integration into the design process, the time required for solution of the RANS equations should be restricted to a few minutes. For three-dimensional configurations there are two orders of magnitude faster than current capability, a new development at the start of the new millennium. Although there is a definite advantage of parallel architectures and increased computer speeds, there is a need

for improvements in algorithms. In design optimization also, algorithm reliability has greater significance. "Modern design optimization algorithms, such as adjoint methods cannot be effective if the flow solver does not converge in relevant areas of the design space" [1].

Human Efficiency: There is a pressing need for a reduction in the "human effort and expertise required for computing flows over complex configurations since humans are not governed by Moore's law" [1] which states that computer power doubles every eighteen months, and the rapid increase in power on this scale is a unique and tremendous advancement in the history of technology [2]. Though expertise in CFD and aerodynamics is vital for carrying out the computations, there should be a minimization in the expertise required in the selection of solver and grid parameters. Another issue to be resolved is the estimation of global numerical error towards significantly reducing user expertise requirements [1].

Accuracy of Physical Modeling: Through methods such as appropriate grid resolution, numerical errors can be carefully controlled. However, errors resulting from "physical models, including turbulence models and prediction of laminar to turbulent transition are more difficult to estimate and control" [1]. The eddy-viscosity turbulence models used at present for

computing aerodynamic flows are unable to predict with accuracy, subtle phenomena such as Reynolds number or flap gap effects on high-lift configurations. The most optimal approach appears to be Reynolds stress models or second moment closures. It is essential to considerably increase the speed with which such models are incorporated into aerodynamic flow solvers for production use at this stage and to assess and guide the development of such models. Further development of turbulence models requires reduction of time needed to compute three dimensional flows; also there is a need for more high quality experimental datasets which include all the boundary condition data needed for computations. To increase the computational efficiency of RANS solvers, two new developments are considered: an inexact Newton-Krylov algorithm which decreases the time needed for computing towards achieving a steady state solution; and “a higher order spatial discretization which decreases the grid resolution requirements for a given level of numerical accuracy” [1], consequently also reducing computing expense. Moreover, objective measures of algorithm performance are required.

TECHNICAL PROCESSES

1. Geometry construction

In this study, the base geometry of a NACA 23012 airfoil with a 23012 external airfoil flap was used ($0.2 C_w$) as illustrated in [Figure 1](#). The vertex information was imported from information obtained by Abbot and Doenhoff [3], and used to define the basic geometries within the program. Next, the vertices were joined using the Non-Uniform Rational B-Spline (NURBS) tool that is part of Gambit. This tool creates a smooth curve over several different vertices. When this is done numerous times for different sets of vertices, edges are created. These can be joined to created faces. The next component of geometry construction is the creation of a far field. A far field is an important component of CFD analysis, as it allows for the calculation of drag and lift coefficients. For this paper, creation of a far field involved creating and connecting three vertices.

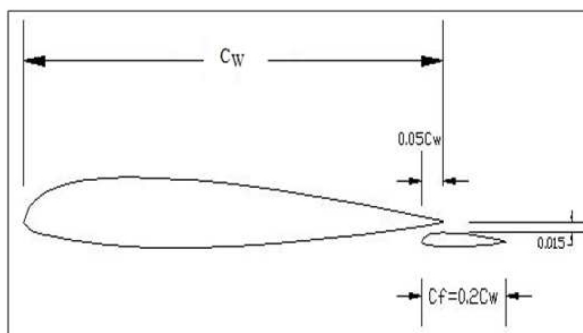


Figure 1. NACA23012 airfoil with external flap

2. Mesh generation

Once the initial geometry for the airfoil has been created, the next stage is the generation of the mesh. A mesh (or grid) refers to the way in which a domain is split up into sub domains, known individually as elements or cells. The creation of a mesh is essential for describing the flow of fluid in a domain through an analytical solution. Once individual cells are created, equations can then be solved for the cells.

Mesheres can be considered in one of two categories, either structured or unstructured. If a mesh is structured, the cells comprising it have a regular pattern. In two-dimensions, this type of mesh makes use of quadrilateral elements. In comparison, there is no order to the elements which are part of an unstructured domain. Using Gambit, meshing in two-dimensions makes use of triangular and unstructured quadrilateral meshing options.

The meshing is created in the order: edges and faces. In Gambit, the resolution of the mesh is related directly to the size along the edges. Another important factor is the boundary layer, which is meshed after the edge meshing. This type of meshing uses any of the three criteria: growth factor, number of rows, and total depth or height. Thus, in this paper all edges along the nose of the airfoil, the bottom and the top were fully meshed. Additionally, all edges of the far field were meshed. Following these steps, a boundary layer was created around the top, bottom and airfoil nose as well as the airfoil flap. The faces created were then meshed. [Table 1](#) shows the number of mesh faces, nodes and cells used in the two-dimensional model, while [Figure 2](#) shows the two-dimensional final mesh for the airfoil and external flap used in this paper.

Table 1. Mesh entities for two-dimensional model

Total mesh cells	172,691
Total mesh faces	347,113
Total mesh nodes	174,421

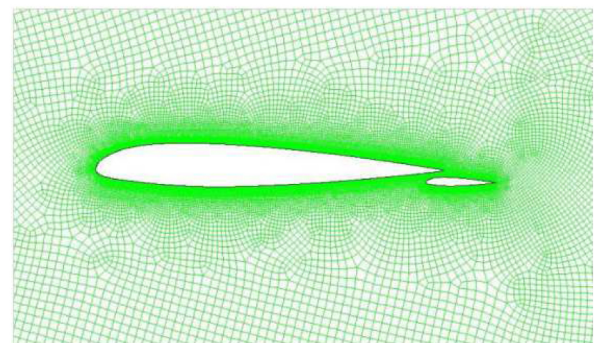


Figure 2. Structured domain meshes for NACA23012 with flap

3. Boundary types

The boundary specification is an important aspect of generating a model. This is crucial both for external and internal boundaries. As shown in Figure 3 the faces that contained the airfoil and flap outline of the far field can be specified as symmetry. This was done to avoid the tangential shear stresses that would be created if these aspects were specified as wall. The inlet boundary condition was specified as inlet velocity, while the outlet boundary condition was specified as pressure outlet. Additionally, the airfoil and flap were also defined as wall. This concluded the work using Gambit, and the file was exported to Fluent.

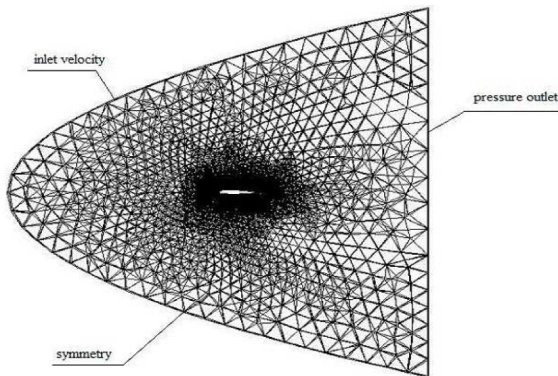


Figure 3. Specification zones for the model

FLOW EQUATION FORMULATION

1. Governing equation

Computational Fluid Dynamics (CFD) is a means of finding a solution to the equations that govern the way in which fluid flow. Three equations form the basis of any statement or calculation of fluid flow. These are the conservation of mass, conservation of energy and Newton's second law. The equations derived from Newton's second law are known as Navier-Stokes equations, which form the basis of most CFD problems. These equations are used to determine outcomes from any fluid flow that occurs in the single-phase. Consequently, the equations for governing incompressible turbulent flow can be represented in vector form, as follows:

$$\nabla \cdot \mathbf{V} = 0 \quad (1)$$

$$\rho \frac{D\mathbf{V}}{Dt} = \rho \mathbf{g} - \nabla p + \nabla \cdot \boldsymbol{\tau}_{ij} \quad (2)$$

where

$$\boldsymbol{\tau}_{ij} = \underbrace{\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)}_{\text{Laminar}} - \underbrace{\rho \overline{u_i u_j}}_{\text{Turbulent}}$$

When there is significant fluctuation in both pressure and velocity, as

$$\begin{aligned} u &= \bar{u} + u' & v &= \bar{v} + v' \\ p &= \bar{p} + p' \end{aligned} \quad (3)$$

The two-dimensional expression for steady state incompressible viscous turbulent fluids can be expressed as follows [4]:

$$\begin{aligned} \rho \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) &= - \frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \\ &\quad - \rho \overline{u^2} - \rho \overline{uv} \\ \rho \left(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) &= - \frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \\ &\quad - \rho \overline{uv} - \rho \overline{v^2} \end{aligned} \quad (4)$$

where

$\rho \overline{u^2}$, $\rho \overline{v^2}$ are two normal stresses

$\rho \overline{uv}$ is stress

and

- (u, v) : velocity components in (x, y) coordinates
- (u', v') : fluctuation velocities
- ρ : density
- p : pressure
- μ : fluid viscosity
- (g_x, g_y) : gravitational components in the (x, y) coordinates.

2. Flow equation solver

The Fluent software package is used to solve CFD problems, and for this work the model created in Gambit was imported into Fluent for analysis. This software operates using a method of finite-volume, which allows for many different types of physical models to be used. For example, the model can exist in any of the following dichotomies:

- Viscous or inviscid
- Turbulent or laminar
- Transient or steady-state
- Compressible or incompressible

For this paper, a two-equation turbulent model was used to solve the governing equations. Here, the Navier-Stokes equations were used in their integral form, with the assumption of constant viscosity and incompressible flow. The realizable k- ϵ model for a single transport equation was used. This allowed for the individual determination of the turbulent kinetic energy and the rate at which it dissipated. This model was particularly relevant for this study, as it made use of an enhanced wall treatment.

Incompressible steady state two-dimensional flow was assumed for this paper which allowed for variables to be solved using a Semi-Implicit Method for Pressure-Linked Equation as the algorithm. The interpolating scheme used in