Numerical Simulation of Dynamic Stall

Debojyoti Ghosh* James Baeder†(Adviser)

AMSC 663 - Advanced Scientific Computation I (Project Proposal)

Abstract

A numerical algorithm will be developed to study the phenomena of dynamic stalling, which is frequently encountered in rotorcraft operations. The governing equations for this problem are the Navier Stokes equations, which describe the behavior of a viscous, compressible fluid. A finite volume algorithm is being proposed which will use a characteristic-based decoupling to compute the fluxes, during the reconstruction stage. The dissipative terms are going to be computed using central differencing of the required order. A time-marching algorithm is planned where the Runge-Kutta class of ODE solvers will be used for the evolution stage. The resulting algorithm is intended to be validated through various stages, beginning with simple benchmark problems involving inviscid flows to viscous, unsteady flows around airfoils.

Introduction

The aim of the present study is to numerically simulate the phenomenon of dynamic stall [1], which is commonly encountered in rotorcraft operations. It is one of the major factors which limit rotorcraft performance, especially at high forward flight speeds and in high g maneuvers. Dynamic stalling on the rotor blades is symptomized by high torsional forces on the blades as well as a drastic loss of lift. Both of these are undesirable and therefore, accurate prediction of the occurrence and effect of dynamic stalling during rotorcraft operation is important to the design process. The distinctive feature of dynamic stalling which sets it apart from the aerodynamic stalling encountered by fixed-wing aircraft is the unsteady nature of the flow. This is due to the time-varying angle of attack of the rotor blade as well as the free-stream velocity that the blade cross-section sees.

Proper computation of the flow during stalling requires a model which can accurately predict boundary layer transition as well as separation. Due to the nature of stalling, the flow is dominated by reverse flow and a convecting vortex. Additionally, even at moderate free-stream velocities, local pockets of supersonic flow can form over the blade upper surface, thus causing shocks in the flow. The presence of shocks further complicates the process of boundary layer separation. To accurately capture all the constituent phenomena in dynamic stalling, we need to solve the Navier Stokes equations, which describe the behavior of a viscous, compressible fluid, along with an appropriate turbulence model. The present study is aimed at solving the 2D flow around blade cross-sections (airfoils) and thus getting an insight into the basic mechanisms behind dynamic stall. Thus, a 2D Navier Stokes solver is planned to be developed which will be coupled with a turbulence model and used to simulate the flow around 2D cross-sections of typical blades. The algorithm will initially be validated through various stages as described later and then applied to the problem of dynamic stalling.

Dynamic Stall

In a steady flow around an airfoil, the flow, and thus the pressure distribution on the surface, is dependent on the angle of attack, i.e, the angle between the free-stream velocity and the airfoil chord (line joining leading and trailing edges). The higher the angle, the greater is the perturbation to the flow, causing higher velocities and lower pressures over the upper surface. The pressure minimum (suction peak) occurs towards the leading edge of the airfoil and the flow over the upper surface

*Department of Mathematics, University of Maryland, College Park, Email: ghosh [at] umd.edu
†Department of Aerospace Engineering, University of Maryland, College Park, Email: baeder [at] eng.umd.edu

downstream of the suction peak experiences an adverse pressure gradient. Under usual conditions, the momentum of the flow helps overcome the pressure gradient. However at high angles of attack, the adverse pressure gradient become too high, causing the flow to separate from the surface of the airfoil. The region of flow right next to the upper surface sees reverse flow and the pressure distribution over the upper surface causing the lifting force is destroyed. This phenomena is termed “stalling” and is characterized by a sudden loss of lift at a high angle of attack. The design of a fixed wing aircraft involves making sure that the maximum angle of attack is never exceeded in any flight condition.

The flow around a rotor blade is complicated due to its unsteady nature. While the flow around a fixed wing can be modeled using steady state flows, modeling flows around rotors has to take into account the time-varying angles of attack as well as free-stream velocities. The rotor is constructed such that during one full rotation, the angle of attack changes sinusoidally. It is at a minimum when the blade is advancing and at a maximum when the blade is retreating. The free-stream velocity that a given cross-section of the blade sees is also a function of time. While the blade is advancing, the free-stream velocity is at a maximum since it is a sum of the forward flight speed as well as the linear blade speed. On the other hand, while the blade is retreating, the free-stream velocity is at a minimum since the blade movement and the rotorcraft motion are in opposite directions. In addition to these unsteady effects, the blade is also subject to the wake from the preceding blade. However, this will not be modeled in the present study.

The unsteady nature of the flow around the rotor blade makes it susceptible to dynamic stalling. The general outline of the process can be describes as follows. As the angle of attack increases from minimum, the lift generated increases as the pressures over the upper surface decrease. As the stall-point is reached, the flow starts to separate from the trailing edge. As the angle of attack increases further, beyond the static stall limit, the flow separation moves upstream to the leading edge. By this time, the flow over the upper surface can separated completely and a vortex is formed in the separated region. The vortex creates a region of low pressure and thus augments the lift, beyond its static stall limit. However, as the vortex starts convecting downstream, it causes a high nose-down pitching moment. This stage is referred to as the “moment stall” and causes high torsional loads on the blade. As the angle of attack increases further, the vortex convects downstream and is shed off the trailing edge, thus causing a sudden loss of lifting force. This stage is called as the “lift stall”. By this time, the airfoil is at its highest angle of attack and starts pitching downwards. As the angle of attack decreases, the flow starts reattaching and by the time the airfoil is at its minimum angle of attack, the flow is attached and well-behaved.

Governing Equations

The behavior of a compressible, inviscid fluid is governed by the Euler equations of gas-dynamics. They consist of the mass, momentum and energy conservation laws applied to a fluid element. They can be expressed in the differential form, for flows involving smooth density and pressure distributions, or in the integral form for flows containing shock waves and contact discontinuities. The Euler equations can be solved to obtain the inviscid flow-field around a given body and thus, they can be used to get a first approximation of the lift and pressure drag generated by the body. However, for proper prediction of dynamic stall and all the accompanying phenomena, the Navier Stokes equations need to be solved. They are a more general case of the Euler Equations, incorporating lossy terms in the momentum and energy conservation equations. These are the terms involving viscous forces and thermal dissipation. Similar to the Euler equations, the Navier Stokes equations can be expressed in a differential form as well as in an integral form. Since the flow involves the possible transition from laminar to turbulent flow, a suitable turbulence model needs to be incorporated into the governing equations.

From the mathematical standpoint, the Euler equations form a set of hyperbolic partial differential equations. The present study aims at solving the flow around airfoil cross-sections and thus, the 2D Euler equations are going to be considered. They form a 4×4 system, with a complete set of eigenvalues and eigenvectors. Each eigenvalue-eigenvector pair represent a characteristic field and there are two acoustic and two entropy characteristic fields. At each point, the acoustic waves propagate in each direction at the speed of sound, relative to the fluid element, while the entropy waves convect with the fluid element at the same velocity. The Navier Stokes equations has the same hyperbolic flux function as the Euler equations. In addition, it has a dissipative source term, thus forming a mixed hyperbolic - parabolic system.

Numerical Scheme

The 2D Euler equations can be expressed in conservative form as

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla . \mathcal{F} = 0 \tag{1}$$

where the flux is $\mathcal{F} = \mathbf{f}\hat{\mathbf{i}} + \mathbf{g}\hat{\mathbf{j}}$, and

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \ \mathbf{f}(\mathbf{u}) = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho u v \\ (E+P)u \end{bmatrix}, \ \mathbf{g}(\mathbf{u}) = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + P \\ (E+P)v \end{bmatrix} \tag{2}$$

Here, ρ is the fluid density, u, v are the Cartesian components of the velocity, P is the pressure and E is the internal energy. As mentioned before, the system is hyperbolic with four eigenvalues as $u, u, u \pm a$ and a complete set of eigenvectors. In the present study, a finite volume approach is going to be used were the fluxes are resolved along the normals to the cell interfaces. The semi-discrete equation, using the finite volume formulation is given as:

$$\frac{d\mathbf{u_{ij}}}{dt} V_{ij} + \sum_{faces} \mathbf{F}.\hat{\mathbf{n}} dS = 0 \Rightarrow \frac{d\mathbf{u_{ij}}}{dt} = \mathbf{Res}(i,j) \tag{3}$$

Here, V_{ij} is the area of the cell. The residual is given by (for a quadrilateral cell)

$$\mathbf{Res}(i,j) = \frac{-1}{V_{ij}} [\sum_{l=1}^{4} \mathbf{F}.\hat{\mathbf{n}}_l dS_l \tag{4}$$

dS_l is the length of the cell interfaces. The semi-discrete equation, as given by eq. (3) is marched in time using the multi-stage Runge-Kutta (RK) algorithm (the order of the RK scheme is chosen to match the spatial accuracy of the flux reconstruction). Note that $\mathbf{F}.\hat{\mathbf{n}} = n_x \mathbf{f} + n_y \mathbf{g}$ is a vector representing the normal flux at a given interface and thus, it can be reconstructed in a 1D manner, along a dimension normal to the cell interface. To exploit the wave nature of the solution, the flux is reconstructed through a characteristic-based process, as outlined below for the flux at the interface between the cells (i,j) and $(i+1,j)$. It is easily extensible to all other interfaces of a given cell.

To compute the interface flux $\mathbf{F}_{i+1/2,j} = (n_x \mathbf{f} + n_y \mathbf{g})_{i+1/2,j}$, the eigenvalues and the left and right eigenvectors at the interface ($\lambda^k_{i+1/2,j}$, $\mathbf{L}^k_{i+1/2,j}$ and $\mathbf{R}^k_{i+1/2,j}$ respectively for $k = 1, ..., 4$) are evaluated at an averaged state (obtained from Roe averaging). The flux is evaluated as:

$$\mathbf{F}_{i+1/2,j} = \sum_{k=1}^{4} fc^k_{i+1/2,j} \mathbf{R}^k_{i+1/2,j} \tag{5}$$

where $fc^k_{i+1/2,j}$ is the component of the flux vector along the kth characteristic direction, evaluated numerically. For a scheme using a stencil S, the characteristic flux at the interface is a function of those evaluated at cell centers lying in the stencil,

$$fc^k_{i+1/2,j} = Rec(fc^k_{i,j};\ i \in S) \tag{6}$$

where Rec is the reconstruction procedure, dependent on the scheme used and the stencil lies along the i-coordinate. In the present study, the Roe-Fixed (RF) formulation [3] is used to evaluate the characteristic flux in an upwinded fashion. The RF formulation is given as

$$\begin{aligned} fc^k_{i+1/2,j} &= fc^k_L, \text{ if } \lambda^k_{i,j}, \lambda^k_{i+1/2,j}, \lambda^k_{i+1,j} > 0 \\ &= fc^k_R, \text{ if } \lambda^k_{i,j}, \lambda^k_{i+1/2,j}, \lambda^k_{i+1,j} < 0 \\ &= \frac{1}{2}[fc^k_L + fc^k_R + \alpha_{i+1/2,j}(u^k_R - u^k_L)], \text{ otherwise} \end{aligned} \tag{7}$$

where $\alpha_{i+1/2,j} = max(|\lambda^k_{i,j}|, |\lambda^k_{i+1/2,j}|, |\lambda^k_{i+1,j}|)$. The terms u^k_R, u^k_L represent the characteristic components of the state vector $\mathbf{u}$, decoupled in the same way as the flux. The RF formulation uses the LLF flux formulation [3] as an entropy fix to the Roe's scheme by introducing extra dissipation and thus breaking up non-physical expansive shocks. Using the RF formulation is also computationally cheaper than the LLF flux formulation since reconstruction of the state vector is required only in cases where entropy fix is required.

Computation of the decoupled fluxes $f^k_{L,R}$ is done using the Essentially Non-Oscillatory (ENO) and Weighted Essentially Non-Oscillatory (WENO) class of reconstruction schemes [3]. The subscripts L, R denote the function interpolated using a left and right biased stencil respectively. The RF formulation, described above, thus represents an upwinding where the left-biased interpolated value is used if the eigenvalues are positive while a right-biased interpolated value is used if the eigenvalues are negative. If the eigenvalue changes sign across the interface, an averaged value is used.

The Navier Stokes equations can be expressed in the same form as Equation (1), except with a source term in the right hand side, which represents the dissipative mechanisms. It is a parabolic term involving the second derivatives of the flow variables. The computation of this source term is typically done through a central differencing procedure, whose order is the same as the overall spatial order of the algorithm. Thus, the Navier Stokes solver involves modifying the Euler solver, described above, by adding an additional subroutine to compute the source term at each iteration.

The above numerical scheme is planned to be implemented in C/C++. As is expected out of most finite volume solutions of the Navier Stokes equations, the solver will be computationally intensive. Since, the aim of the study is to solve for 2D flows around airfoils, it is felt that a serial code will suffice and it is planned to run the code on single CPU machines. The author has available three such machines, with processor speeds 2.4 GHz and 3.6 GHz, and 4 GB memory each. Any one of these machines will do for the desired computations.

Validation

The validation process is going to be in stages. The first part involves the validation of the 2D Euler code. The simplest benchmark problems for the 2D Euler equations require a square/rectangular domain (with a Cartesian mesh). The first set of such problems are the 2D Riemann problems formulated in [4]. They involve a square domain, divided into four quadrants. The initial conditions involve four different constant states in each of the quadrants. The second Cartesian problem is the shock wave reflection which consists of an oblique shock wave reflecting off a solid wall. This problem tests the ability of the algorithm to capture shocks in a supersonic flow and the imposition of solid wall boundary conditions. These problems have been studied exhaustively in literature and serve as standard test problems for any 2D solver. Thus, it will suffice to compare the obtained results with those in literature. Additionally, the shock wave reflection problem has an analytical solution to which the computed results can be compared to.

The next stage involves validation of the Euler code on non-Cartesian meshes. The main advantage of the finite volume formulation is its ability to be applied to a body-fitted curvilinear mesh without the use of transformation metrics. Thus, it is expected that the same algorithm will be able to compute solutions on non-Cartesian meshes without further modifications. The first non-Cartesian case is the compression ramp problem, which involves supersonic flow negotiating an inward ramp, thus forming a shock wave at the beginning of the ramp and an expansion wave at the end of the ramp. The mesh for this domain is mildly non-Cartesian. Following this, it is intended to use the algorithm to solve for supersonic flow around a cylindrical blunt body, which involves the capturing of a detached bow shock in the solution. Finally, the algorithm will be used to solve for steady flows around standard airfoil shapes, for which experimental and computational results are available. The algorithm will yield the flow quantities (pressure, velocity, density, etc) at each point in the domain and thus, a post-processing stage is required to compute the lift, drag and moment coefficients from the flow data.

Following the validation of the Euler solver, the Navier Stokes solver will be developed. This requires an additional module computing the dissipative source terms at each iteration. Additionally, a turbulence model needs to incorporated such that physically relevant solutions are obtained which model the transition of flow from laminar to turbulent. To start with, it is planned to validate the Navier Stokes solver for steady flow around airfoils and prediction of the lift, drag and moment coefficients. Since the Navier Stokes equations include the dissipative mechanisms present in the flow, the drag coefficients are expected to be much more realistic and thus, easily validated by experimental results.

After the above mentioned validation stages, the Navier Stokes solver will be used to solve for unsteady flow around airfoils, which includes cases with well-behaved flows as well as dynamic stalling. There are a number of experimental and computational results for unsteady flow around standard airfoil shapes and thus, the obtained results can be easily validated. It is planned that the airfoils studied will include the symmetric NACA 0012 and 0015 airfoils as well as airfoils designed specifically for rotor blades [2]. The comparison of the results for these different classes of airfoils is expected to further the author's understanding of this complicated phenomenon.

Timeline

- **October** - Submission of proposal, writing the basic 2D Euler code, validating it on Cartesian benchmark problems
- **November** - Application of 2D Euler code to non-Cartesian meshes (compression ramp, blunt body problem and steady flow over airfoils), Beginning of literature survey on the Navier Stokes equations and algorithms used to solve them
- **December** - Modifying Euler code to solve the Navier Stokes equations, preliminary validation, Submission of mid-year progress report, literature survey various turbulence models used and the selection of one specific model for the present study
- **February** - Incorporation of turbulence model, application of Navier Stokes solver to steady flows around airfoils, computing aerodynamic coefficients for various airfoils, comparison with available results, correction of code if necessary
- **March & April** - Study of unsteady flow around airfoils, including dynamic stall, comparison with experimental and computational results, study of the flow around various airfoils (general purpose as well as rotor-specific shapes)
- **May** - Submission of final report

References

[1] Leishman J.G., *Principles of Helicopter Aerodynamics*, Cambridge University Press, 2006

[2] Piziali R.A, *2-D and 3-D Oscillating Wing Aerodynamics for a Range of Angles of Attack including Stall*, NASA-TM 4632, 1994

[3] Shu C.W., Osher S., *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*, ICASE report 97-65, 1997

[4] Lax P.D., Liu X.D., *Solution of Two Dimensional Riemann Problems of Gas Dynamics by Positive Schemes*, SIAM Journal of Scientific Computing, **19** (2), 1998, 319 - 340