# P­ropeller Performance Modelling Using BEMT

The Blade Element Theory [1] attempts to estimate the performance of a propeller by dividing the blade into small sections called blade elements and then treating each element as a separate 2-D airfoil. The aerodynamic characteristics of each element are calculated based on the local flow conditions and are integrated in the end to give the performance of the complete propeller. The Momentum Theory is used to estimate the induced AOA which results from the induced velocity of air; effectively making BET into BEMT. BEMT was implemented in a MATLAB code and the verification of the code was possible thanks to the propeller databases made available by UIUC (University of Illinois Urbana Champagne) and APC propellers. The geometry file of a propeller was used to populate the MATLAB code and the results were compared to the corresponding performance file.

## Integration of XFOIL

A challenge with the BEMT code is that although the determination of aerodynamic characteristics of each section individually based on local flow conditions and AOA allows for high fidelity, the accurate determination of these characteristics using a singular lift-drag equation renders the solution inaccurate since the lift and drag equation is specific to the Reynolds number; which perceptibly changes along the blade span with increasing tangential velocities.

XFOIL is a well-known panel method solver for determining the airfoil aerodynamic characteristics and was automated into the BEMT code such that it could be called iteratively for each segment of the propeller. However, frequent convergence errors necessitated the use of some other aerodynamics software.

## Integration of VSPAERO

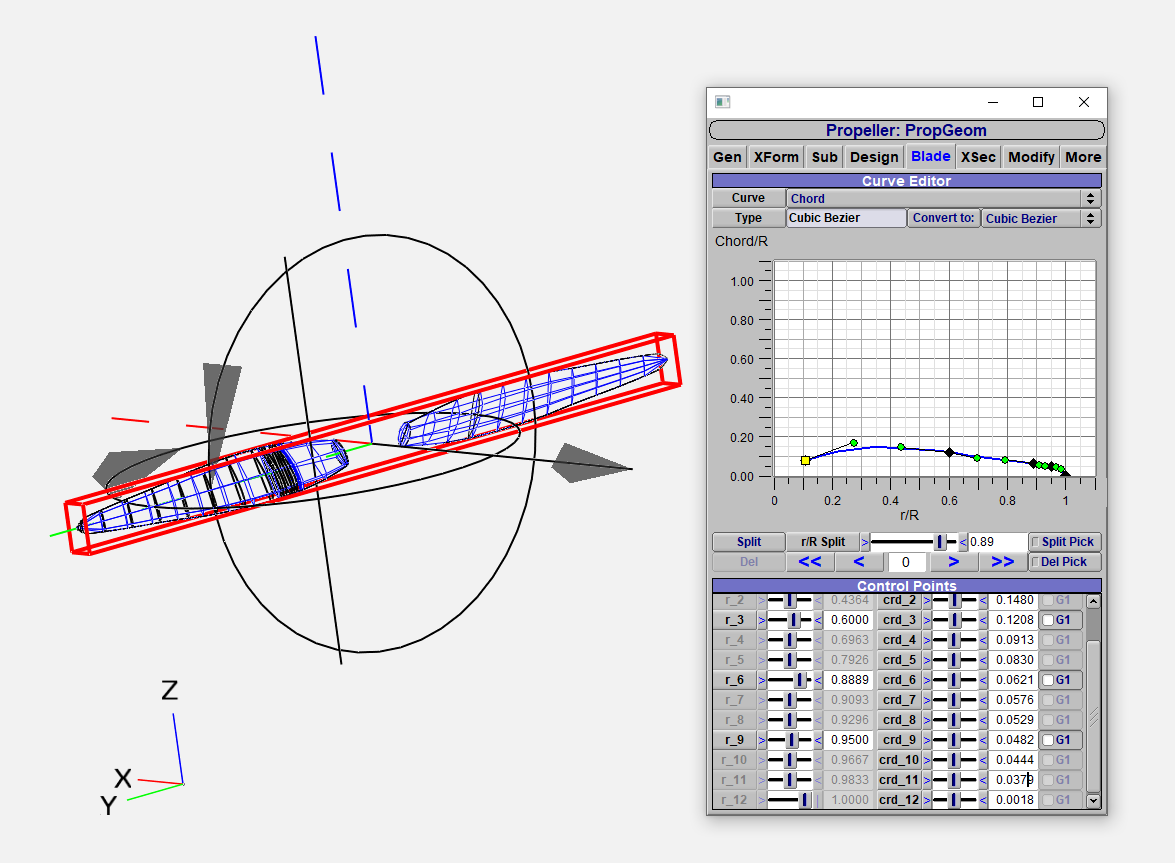
Although VSPAERO has been developed for aerodynamic analysis of 3-D bodies, airfoil analysis can be made possible by making a pseudo-infinite wing i.e. a wing with a very large span. The automation of VSPAERO into the BEMT code is still in progress and will hopefully bring the expectations of a high fidelity propeller performance tool into fruition.

For now, we are certain that the BEMT code is valid from an algorithm standpoint since even with the same two equations for lift and drag at each segment, we get a result which resembles the actual performance data which is a good indication of success once VSPAERO is integrated in the BEMT code.

|  |  |
| --- | --- |
| Propeller Efficiency Comparison | |
|  |  |
| Propeller CT Comparison | |
|  |  |

# P­ropeller Performance Modelling Using VSPAERO

Aside from wings and fuselage, the geometry module of OpenVSP allows modelling propellers. There are two ways to model propellers in OpenVSP; as a momentum imparting disk (similar to what we see in the development of the Momentum Theory); and as an actual propeller with blades, airfoils, etc. The propulsion team was able to use the geometry files available for the APC propellers, to recreate the propeller geometry in OpenVSP. The process for recreating the geometry is not trivial as it requires inserting the sections in the propeller object manually and then interpolating section characteristics from the geometry file in EXCEL since the control points for propeller pitch are predefined by VSP.



Once the propeller geometry was built in VSP, VSPAERO was used to determine the aerodynamic performance of the propeller. The short-term goal was to analyse the propeller performance at a few conditions for which the data exists in the performance files so as to get an idea of usability / accuracy. The long-term goal was to automate the use of VSPAERO to determine the propeller performance at multiple conditions using the scripting feature in VSP.

VSPAERO does an unsteady simulation of the propeller i.e. the boundary conditions are time dependant, the usable performance values of the propeller are from the last few iterations in the output files. VSPAERO has two solvers, i.e. Panel method and Vortex Lattice method (VLM).

## Panel Method

The panel method was first used to analyse the propeller performance and for the most part, the solver refused to run, reasons were unclear since the geometry was simplified as much as possible however the solver would simply stall in the very start. Although the team would have preferred to use the panel solver as it is an ideal method for airfoil analysis and generating Cp graphs, any efforts made to use panel method for analysing the propeller did not bear fruit.

## VLM Method

The VLM method was flexible and would always run with the propeller geometry, the problem however was that the VLM method required a 2-D / surface geometry for its analysis. Also known as the degenerate geometry, it reduces the propeller airfoils into curved lines which of course leads to losses in accuracy. The results of the simulation were in correspondence with the performance data and were used for further analysis.

# Engine Performance Modelling

Shaft Horsepower data for every prospective engine, along with specific fuel consumption in some cases, was available in engine data sheets. The Power data variation with altitude was determined using the rudimentary density ratio relation used in performance textbooks such as [2]. Future endeavours will include the development of an actual physics-based model of the Internal Combustion Engine similar to the BEMT model for propeller performance analysis. The following flow chart includes the current methodology used to build an engine deck which relies on engine data available from the data sheets and propeller performance data from either the performance files or the BEMT code.

