12-21-2017

Reducing Computation Time of Vortex Position For Helicopter Flight Simulations

Matthew Tibbetts
Washington University in St. Louis

David Peters
Washington University in St. Louis

Reducing Computation Time of Vortex Position For Helicopter Flight Simulations

David Peters, Matthew Tibbetts

December 16, 2017

Abstract

The purpose of this investigation was to reduce the computational cost of vortex mapping for helicopter flight simulators, with the particular case of landing a helicopter on a moving ship. We pursued this goal by modeling the velocity using a series of Fourier transforms, linear fits, and Legendre polynomial fits. The end result was a matrix of 9 coefficient values which could be used to recreate the vortex behavior of the original data set with some accuracy. This technique will allow flight simulators to generate vortex data quickly to improve their effectiveness in training pilots.

1 Introduction

Flight simulators are an incredibly valuable tool for training pilots without risking expensive equipment or endangering the pilots. As computation power grows, simulators become more able to recreate the experience of flying in various conditions, presenting pilots with challenges before they have to face them in the real world so that they can be better prepared. As the fidelity of simulations rises, so too does the computing power required to calculate the hundreds of variables that affect an aircraft's flight. Any property which can be successfully modeled mathematically with a small number of inputs frees up computing power for other vital systems.

One of the most common challenges a pilot-in-training must prepare for is the effect of air vortices on the aircraft's flight. Helicopters in particular are vulnerable to flight instability when encountering vortices due to the large surface area and high speed of their rotors. Calculating the position and magnitude of vortices is, therefore, a vital component of any realistic flight simulator. Conventional methods for calculating vortices are time-consuming and computationally expensive.

Our goal with this investigation is to reduce the amount of inputs necessary to generate a velocity field mathematically. We used a computational fluid dynamic simulation as test data, and derived a method by which the velocity data can be modeled using only a small array of coefficients.

Our method of obtaining these coefficients consisted of five steps. First, we took a Fourier transform of the velocities with respect to time for each position. We then fit the resulting velocity data with a least squares fit line, and plotted the fitting parameters, U_0 and λ, with respect to x position. We then took a Fourier transform of this data with respect to x, and collated the resultant frequency plot so that the parameters fell into two peaks. We created an array of the peak values for the second peak with respect to y and z position and fitted this surface with a series of Legendre polynomials. The coefficients of these polynomial products is our final product.

The ultimate goal of this project is to use these coefficients to recreate the original data set and determine how accurate our method is at modeling velocity data. If successful, this would allow a flight simulator to model vorticity positions while only storing a small array of values, preserving fidelity while decreasing computation time and space.

1.1 Geometry

For this investigation we used as our original data set a computational fluid dynamics (CFD) simulation consisting of velocity in the $x(U)$, $y(V)$, and $z(W)$ directions at 43493 positions: 61 points in x, 31 points in y, and 23 points in z. The simulation takes place over 125 time steps of 0.373 seconds, and the position values are in steps of 2 feet. The field of interest is above the landing pad and in the wake of a Simple Frigate Shape (SFS-2), as seen in Figure 1.

Values in x are positive behind the ship with the origin at the hangar/superstructure wall, ranging from -30 to -150. Values in y are positive starboard centered at the mid line of the ship, ranging from -30 to +30. Values in z are positive down with the origin at the water line, ranging from -15 to -59. The velocities are measured with respect to the ship.

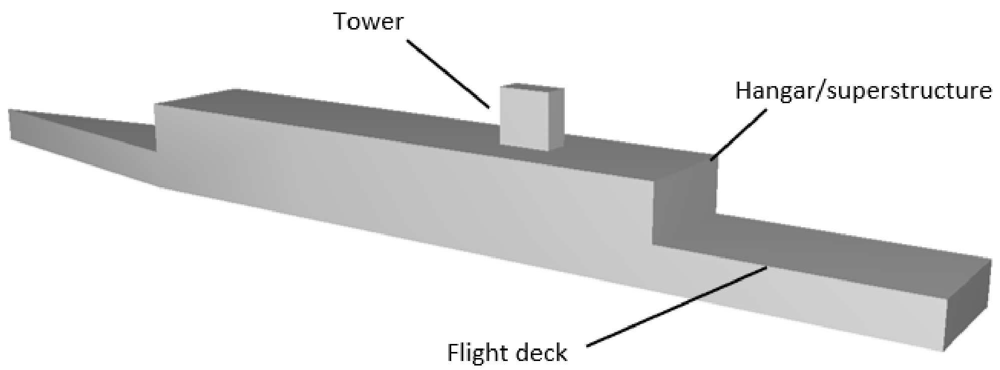

Figure 1: Simple Frigate Shape (SFS-2) used in the computational fluid dynamics simulation that produced the original data set

Description	Value
Length	455
Beam	45
Waterline	Z = 0
Flight deck elevation	Z = -15
End of flight deck	X = -120
Flight deck extent	Y = -22.5 $\rightarrow$ 22.5
Superstructure elevation	Z = -35
Tower elevation	Z = -55
Hangar face	X = -30
Tower longitudinal extent	X = 20 $\rightarrow$ 40
Tower lateral extent	Y = -5 $\rightarrow$ 5

Table 1: Important values defining the SFS-2 geometry

1.2 Mathematical Theory

This investigation primarily uses two mathematical modeling techniques. Fourier transforms are used to reduce the number of dependent variables by expressing values as a function of frequency rather than the variable of interest. These frequency plots are often useful for determining a characteristic frequency which captures most of the behavior of the value. Legendre polynomial fitting is here used to further reduce an array of values in two dimensions to a much smaller matrix of coefficients. The Legendre polynomials are a set of orthogonal polynomials which are often used to model uneven surfaces.

The Fourier transform takes the form

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i x \omega}\, dx$$

Where x is the variable of interest and ω is the frequency of the function. The

Fourier transform can be reversed with the inverse Fourier transform, of the form

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\omega)\, e^{2\pi i x \omega}\, d\omega$$

In this way a function can be transformed into its frequency form, analyzed, and then returned to its original dependency. In our investigation this process is repeated twice: first on the velocity with respect to time, then on the fitting coefficients with respect to x.

The Legendre polynomials are a set of orthogonal polynomials of the form

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n}(x^2 - 1)^n$$

With the normalization of $P_n(1) = 1$. In our investigation, these polynomials are used as fitting functions for a surface dependent on both y and z. The fitting surface is of the form

$$U_0, \lambda = \sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij} P_i(z) P_j(y)$$

Where N is the degree of the fit. So, for example, the 2nd order Legendre fit of the U_0 parameter would take the form

$$U_0 = c_{11} P_1(z) P_1(y) + c_{12} P_1(z) P_2(y) + c_{21} P_2(z) P_1(y) + c_{22} P_2(z) P_2(y)$$

With these two tools, we were able to take an array of more than 21 million values and reduce it to an $N \times N$ array of coefficients that a computer could use to reasonably recreate the original data set.

2 Procedure

Our process begins with selecting a component of velocity to analyze. In this investigation we chose U, the velocity in the x direction, because it is the most important in determining vortex positions. Our original data is an array of velocity values in

x,y,z, and t, as seen in Figure 2. Our first goal is to reduce the number of dependent variables we have to work with.

We start by analyzing each position independently, plotting the velocity with respect to time as seen in Figure 3. Our test point for these plots is at the front end of the landing pad ($x = -30$), on the centerline of the ship ($y = 0$), midway between the landing pad and the top of our area of interest ($z = -39$). Our process repeats each step for the full range of positions.

The first step of the process is to take a Fourier transform of the velocity data with respect to time, producing Figure 3. In Figure 3, we plot the natural log of the velocity versus the frequency in Hertz. We then find the least squares fit line for this data set and record the slope and intercept values. These will become our fit parameters. The fit line is of the form

$$U = U_0 \, e^{-\lambda f}$$

Where U is the velocity, U_0 is the exponential of the intercept of the fit line, and λ is the slope of the fit line.

We now have two data sets,U_0 and λ, each with respect to only x,y, and z. We want to reduce this dependency further, so we plot the fit parameters with respect to x, as seen in Figure 5. We take the Fourier transform of these parameters with respect to x, producing a plot of values with respect to the wave number, k. This can be seen in Figure 6.

We can see in Figure 6 that only two distinct peaks appear on this plot. The first and largest peak is at $k = 0$, representing the constant base velocity. We are interested in relative motion within the field, so we focus on the second peak, near $k = 4$. In order to capture the full behavior of the velocity field, we sum the values between $k = 2.5$ and $k = 5.5$ and place them at $k = 4$, as seen in Figure 7. The values of U_0 and λ at this point, then, are our values of interest.

We plot these two arrays of peak values in y and z, as seen in the interpolation plots of Figure 8. We then use the Legendre fitting polynomials to create fit surfaces of order 3 to the data, as seen in Figure 9. Each fit surface is characterized by a 3×3 array of coefficients. These coefficients can be found in Table 2 and Table 3.

3 Results

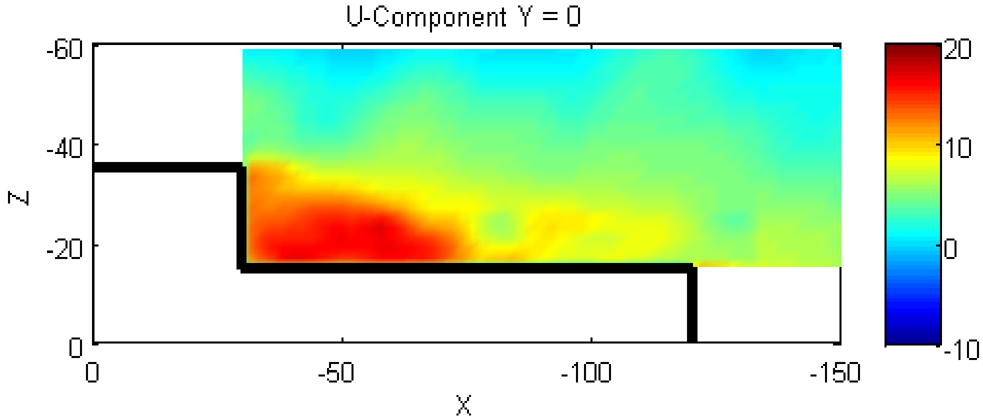

Figure 2: Contour plot of velocity in the x direction at the ship centerline in y at a given time

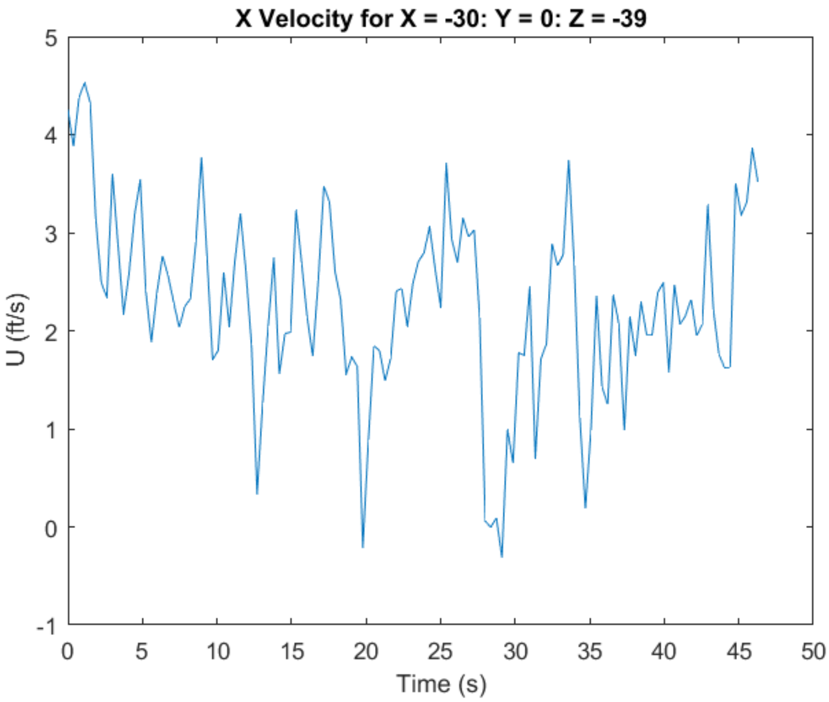

Figure 3: Velocity in the x direction plotted as a function of time for a particular location (x,y,z)

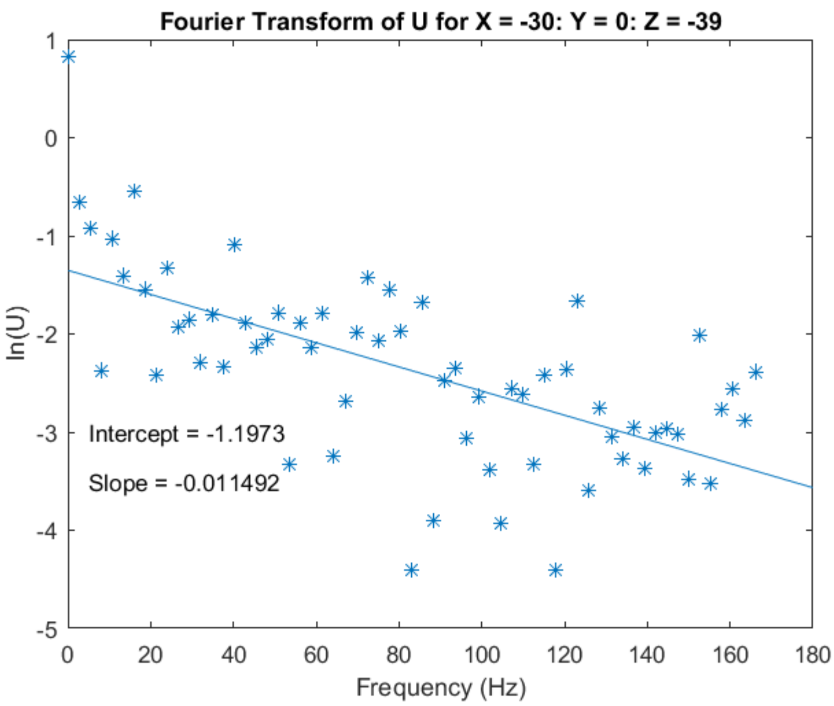

Figure 4: The natural log of the Fourier transform of velocity plotted with respect to f for a particular location (x,y,z)

Figure 5: Fitting parameters plotted with respect to x for a particular (y,z)

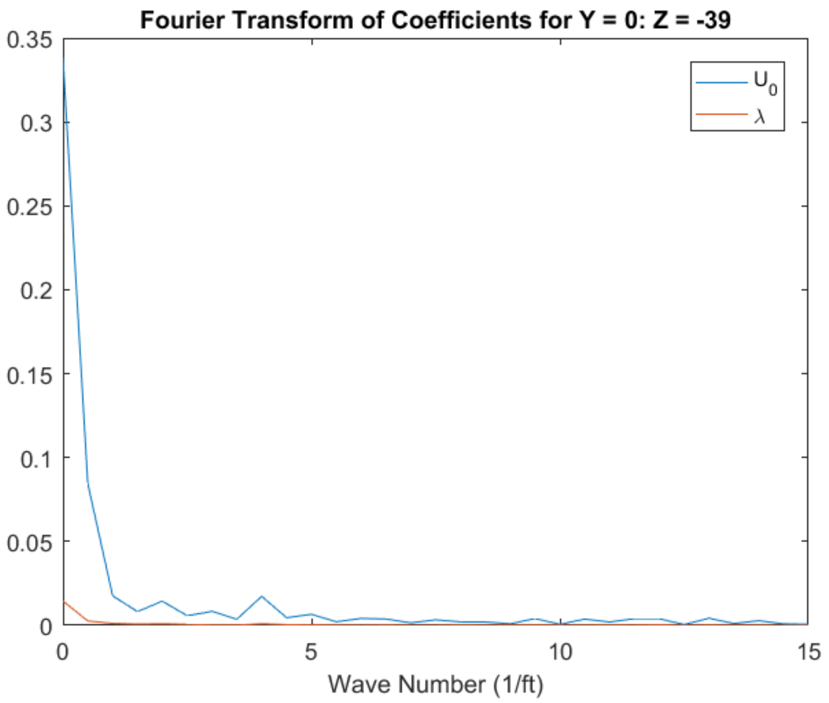

Figure 6: Fourier transform of the fitting parameters plotted with respect to k for a particular (y,z)

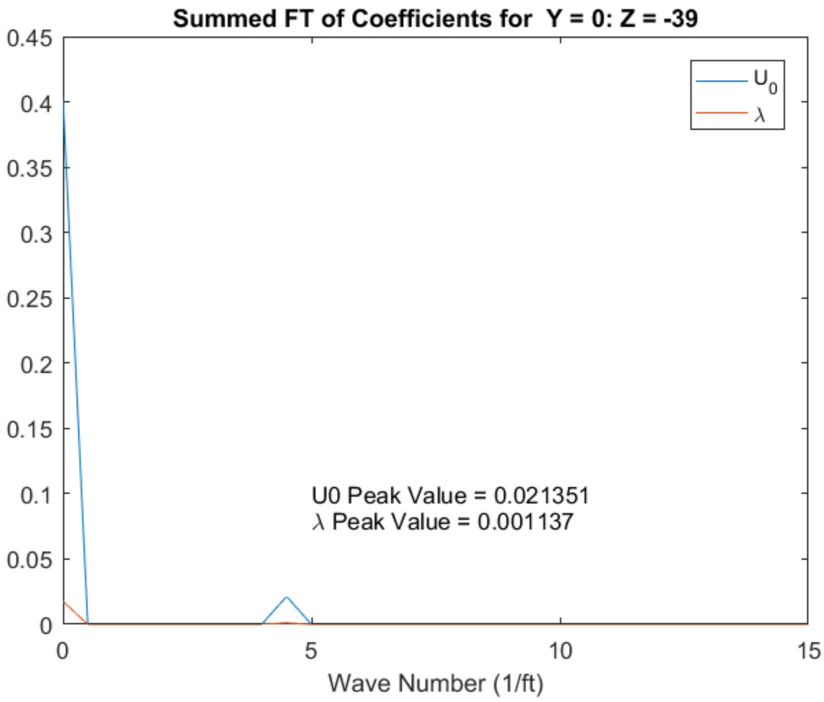

Figure 7: Summed Fourier transform of fitting parameters with respect to k for a particular y,z

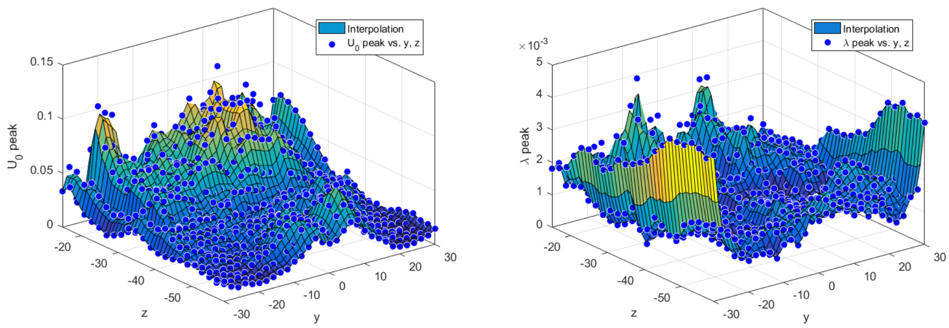

Figure 8: Interpolation plot of parameters U_0 and λ for all y, z

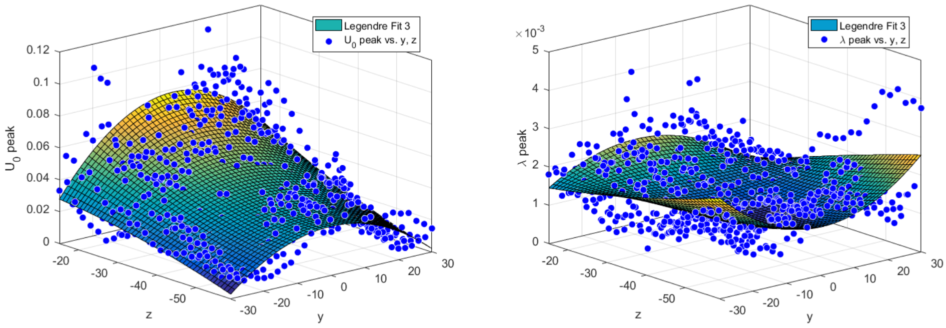

Figure 9: Legendre fit of order 3 of parameters U_0 and λ for all y, z

	$P_1(y)$	$P_2(y)$	$P_3(y)$
$P_1(z)$	1.16e-01	2.95e-03	1.702e-05
$P_2(z)$	6.93e-04	2.754e-05	1.749e-07
$P_3(z)$	-4.815e-05	-1.524e-06	-1.202e-08

Table 2: Coefficients of U_0 peaks for 3rd order Legendre polynomials

	$P_1(y)$	$P_2(y)$	$P_3(y)$
$P_1(z)$	3.62e-03	1.15e-04	8.517e-07
$P_2(z)$	-1.598e-05	-5.744e-07	-3.751e-09
$P_3(z)$	-2.112e-06	-1.057e-07	-6.026e-10

Table 3: Coefficients of λ peaks for 3rd order Legendre polynomials

4 Analysis

In this investigation, we successfully reduce a data set dependent on four variables to an array of coefficients through the use of Fourier transforms and fitting. This process makes several assumptions and introduces error at each reductive step. Our fit line to the Fourier transform of velocity captures only a trend, and the wide scatter seen in Figure 4 certainly introduces error into our final product. We also collate the range of values in the Fourier transform of the fitting parameters in Figure 7, which removes some of the nuance of the actual shape. Finally, our Legendre fitting surfaces do not line up with all the data, as seen in Figure 9.

Our investigation could improve the accuracy of our results primarily by increasing the order of the Legendre fit. In our process, the 3rd order Legendre surface was the highest order which produced usable results, but ideally a 10th or higher order fitting surface would capture the data better, and the increase in fidelity would only require a modest increase in data, from 9 values to 100. While that may seem like a large increase, these arrays are still replacing the millions of cells required to record the CFD simulation.

5 Conclusion

Our analytical process reduced a $61 \times 31 \times 23 \times 125$ array of values to a 3×3 array, requiring only the mathematical framework to restore the data set from these coefficients. We use Fourier transforms and fitting functions to decouple dependency on two of the four dependent variables from the original data.

Given our desired outcome, the error we introduced during the fitting process is acceptable. Vortex position and velocity are highly stochastic, and any mathematical model must necessarily provide the broad strokes of the shape of the velocity field, allowing the simulation to stochastically augment in order to provide a realistically challenging experience.

While we achieved the results we wanted from this exploration, there is still work to be done improving and verifying this process. We were unable due to time constraints to implement a reversal of the reduction process, taking the coefficients and generating a full velocity data set. We also only modeled a Legendre fit of oder 3, which does not capture the true shape of the surface. Future work on this subject could increase fidelity and demonstrate the full functionality of this process.