AIAA SciTech
5-9 January 2015, Kissimmee, Florida
53rd AIAA Aerospace Sciences Meeting

AIAA 2015-1711

Comparison of CFD Hover Predictions on the S-76 Rotor

Jennifer N. Abras[1]
NAVAIR Applied Aerodynamics and Store Separation Branch, Patuxent River, MD, 20670

and

Nathan Hariharan[2]
HPCMP CREATE-AV, Lorton, VA, 22079

The S-76 rotor is used as a baseline case to assess predictions made by different CFD solvers. These predictions are compared to the available test data as well as to one another. Both grid and parametric studies of the options available within each code are included. The grid studies look into not only blade grid density, but also Cartesian and unstructured far field grid computations, and the use of AMR. The results show that, in addition to blade tip grid refinement, leading edge and trailing edge grid refinement are important to compute the hover performance. The dual mesh methodology is shown to preserve the wake for a longer distance when compared to the fully unstructured methodology. This has some impact on the final wake structure.

I. Introduction

THE CREATE^{TM}-AV program is focused on developing state-of-the-art software for use with both fixed wing and rotary wing aircraft analysis. These codes are HPCMP CREATE^{TM}-AV HELIOS[1] for rotary wing analysis and HPCMP CREATE^{TM}-AV KESTREL[2] for fixed wing analysis. Release versions of both of these codes are currently in use in government, industry, and academia. However, the development process to expand the capabilities of these codes and to improve on the present capabilities is ongoing. While these changes and improvements encompass a wide range of topics, one specific area of improvement is the addition of a new unstructured CFD solver in HELIOS.

Evaluation of major changes in the configuration of any CFD code are pursued through comparisons to both available measured data as well as to the predictions produced by established CFD solvers. Planned changes to the HELIOS solver will be evaluated using these types of comparisons. The unstructured CFD solver in the current release version of HELIOS is NSU3D. Future versions of HELIOS are planned to also include the kCFD solver which currently resides in KESTREL. It is of interest to compare both kCFD and NSU3D to better understand the possible impact that the transition will have on the HELIOS solver. The overall plan is to compare the predictions produced by both of these codes to one another as well as to available wind tunnel data and predictions from the FUN3D[3] solver. It should be noted that running a hover performance study properly takes time. Each case requires more resources than many other typical CFD studies. As a result, this work is ongoing and only a subset of the work completed towards these goals will be presented in this paper.

The Rotorcraft Simulation Working Group established by the AIAA Aerodynamics Technical Committee is an ideal venue in which to perform these assessments[4,5]. This group was established with the express purpose of combining the efforts of multiple organizations using a standardized format to assess the current state-of-the art and identify the needs and future direction of hover prediction technology. This format allows the products of the current effort to be evaluated not only independently, but also against cases run by other organizations. This workshop utilizes wind tunnel test data produced by Sikorsky in the 1980s[6,7] on the S-76 rotor in hover. The goals of the workshop are not only to assess the predictions on the baseline rotor, but also to look at the impact of different blade tip geometries.

This paper compares the hover performance predictions of the HELIOS solver against available wind tunnel data and predictions made by FUN3D. The comparisons against KESTREL are ongoing and were not ready for

[1] Aerospace Engineer, jennifer.abras@navy.mil, AIAA Professional Member.
[2] Deputy Project Manager, nathan.hariharan.ctr@hpc.mil, AIAA Associate Fellow.

publication at the time this paper was written. The current comparisons include grid density studies, overall solver comparisons, and timing and convergence comparisons. The final comparisons primarily focus on integrated thrust, torque, and figure of merit as these are the only parameters provided in the test results. However, more detailed analysis comparing the wake preservation and structure produced by each solver are also included.

II. Methodology

A. HELIOS v4.2

HELIOS is a multi-function code designed specifically for rotorcraft analysis. It contains unstructured near-body and Cartesian off-body CFD solvers, overset methodology using PUNDIT[8], rigid grid motion, and rotor blade structural deformations through RCAS[9] or through prescribed deformations. The near-body solver is NSU3D[10], a node-centered unstructured CFD code that solves the Reynolds Averaged Navier-Stokes (RANS) equations applying 2nd-order spatial accuracy. The Spalart-Allmaras (SA) and SA-DES turbulence models are available. The unstructured grid can contain a mixture of tetrahedral and prismatic cells. SAMARC[11] controls the off-body computations, where ARC3D[12] is the off-body Cartesian solver which applies up to 5th-order spatial accuracy using a central differencing scheme and uses a Runge-Kutta explicit time integration scheme which provides 3rd-order temporal accuracy. This solver may be run inviscid or with the SA-DES turbulence model. Automatic mesh refinement (AMR) may also be applied to the off-body solution. This is accomplished through the SAMRAI[13] framework and guided adaptive mesh refinement[14] (GAMR) methodology. The latest version of HELIOS also includes the option to use OVERFLOW[15] as the near-body solver. Results using this version of HELIOS on the S-76 rotor have been previously presented[16]. The performance predictions are extracted for comparison later in this paper. Cases using NSU3D alone and dual NSU3D and ARC3D are presented. All cases apply the SA turbulence model to the near-body and the Euler solution in the off-body.

B. FUN3D v11.3

The NASA Langley code FUN3D is an unstructured CFD code that solves the RANS equations using a node-centered 2nd-order upwind implicit scheme. The unstructured grid can contain a mixture of tetrahedral and prismatic cells. This code is able to perform a variety of analyses including rigid and elastic grid motion, 6DoF analysis, design optimization, and grid adaptation. Overset methodology is applied using SUGGAR[17] and DiRTlib[18]. This code also contains a variety of rotorcraft specific options including coupling with external comprehensive rotorcraft codes. There are a wide variety of flux schemes and turbulence models available, but for this study the Roe flux scheme and the SA turbulence model with the Dacles-Mariani rotation correction have been selected.

III. Conditions

The parameters applied in this study are provided as part of the AIAA workshop guidance. There are a series of three different tip geometries provided for analysis. The baseline blade has a tip with sweep and taper as shown in Figure 1. The other blades are a straight tipped blade and a blade with sweep, taper, and anhedral. The baseline case is the focus of the current study. All cases for the current paper are run in hover with a tip Mach number of 0.65 at standard sea level conditions. A range of collectives are requested from 0° to 10°. In the present paper only 8° through 10° at intervals of 1° are investigated. Test data for thrust, torque, and figure of merit are available for comparison.

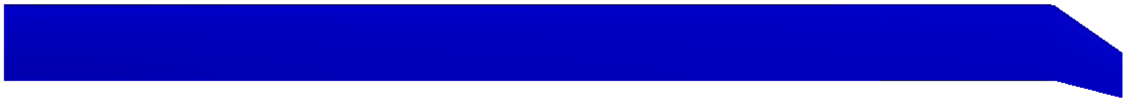

Figure 1: Baseline blade planform.

IV. Grids

The underlying CAD used for this model originated from the files provided through the AIAA workshop. This includes both the blades and the hub. These files were used within the NASA Langley TetrUSS tools, GridTool and VGrid, or CREATE^{TM}-MG Capstone to generate the tetrahedral unstructured grids. Since two different solvers are being used for this analysis that use different overall grid systems (FUN3D employs fully unstructured grids, HELIOS is run with the Cartesian off-body), the grids applied to each code will be different. The similarities are in the region immediately around the blade grid. It should be noted that the same initial viscous spacing was applied to all of the blade meshes which computes a maximum y^+ on the surface of about 0.42. The differences are in the region away from the blade and in the background grid. The FUN3D background grids are fully unstructured and refinement was explicitly added in the wake region. These grids varied from 35.7M nodes to 40.1M nodes as the grid study progressed. The HELIOS background grids, also referred to as off-body grids, are automatically generated by the solver. The details for the blade grids are provided in Table 1.

Table 1. Baseline blade grids used during the study.

	Grid Generator	# Nodes/Blade	# Nodes/Blade (modified)	Description
Grid c	Capstone	690K	N/A	Initial case, no specific refinement regions
Grid v1	VGrid	3.4M	5.1M	Interactional aero grid, refined tip region
Grid v2	VGrid	10.9M	N/A	Rotor performance grid 1, refined LE/TE
Grid v3	VGrid	14.7M	15.7M	Rotor performance grid 2, refined LE

The first grid (grid c), generated in Capstone, simply applied a visibly acceptable surface refinement with no apparent facets in the mesh then applied a volume mesh of increasing size to the outer boundary. Cell stretching was applied spanwise along the inboard regions of the blade using a very nice surface boundary layer growth feature, which created a semi-structured grid pattern around the leading edge and trailing edge of the blade which allowed for more highly stretched cells along the inboard region which accounts, in part, for why this mesh is so much smaller. While this mesh is not expected to produce good hover performance results, it does provide a good contrast with the results produced using the more refined meshes.

The next mesh (grid v1) refined the whole blade, but primarily the blade tip region. The difference between the original and the modified node counts is in how far the refinement extends away from the blade. For the cases that use the Cartesian off-body with AMR it is advantageous to extend the refinement to the outer boundary so that the near-body wake dissipation is minimized before the vortex transitions to the off-body mesh. The returning vortex region is also refined below and inboard of the blade tip for the same reason.

The third grid (grid v2) is the first genuine attempt at computing acceptable hover performance. This mesh has extra refinement at the leading edge and the trailing edge with 8 to 10 cells across the trailing edge. The last grid (grid v3) added even more refinement around the leading edge. This step was taken to investigate the applicability of research performed in adjoint-based design optimization[19] that predicted optimized airfoil lift using additional grid refinement ahead of the leading edge to the current study. Grid v4 was planned to add additional refinement to the trailing edge to enforce and expand upon the 10 cell rule of thumb more consistently, but grid v3 was found to produce sufficiently accurate results for the current analysis.

V. Results

All FUN3D solutions were run using the Spalart-Allmaras turbulence model with the Dacles-Mariani rotation correction. Time step sizes equivalent to 1° of rotor rotation were used. The subiteration count was determined based on the convergence of the forces and moments on the subiteration level. The HELIOS solutions were run with the Spalart-Allmaras turbulence model in the near-body and apply the Euler equations in the off-body. The time step sizes applied were equivalent to 0.1° of rotor rotation for grid c and grid v1, a step size of 0.05° is used for grid v3 for stability reasons. The subiteration count was determined based on a density residual drop of one order of magnitude. Further discussion of the convergence characteristics may be found in the computational metrics subsection.

A. Grid Independence Study

The grid independence study was conducted primarily within FUN3D. All grids were run for 9° and 10° collective settings. A subset of cases was also run with HELIOS at a collective setting of 9°. The results of each study are discussed below.

1. HELIOS

Three cases so far have been completed with HELIOS. These are the grid c, grid v1, and grid v3 cases. While the first two of these are not considered to be hover performance grade grids, and the results support this assessment, there is important information that may be derived from these cases. The first point is the impact of the near-body dissipation on the off-body AMR. Figure 2 compares the final adapted grids for the first two cases. In these images

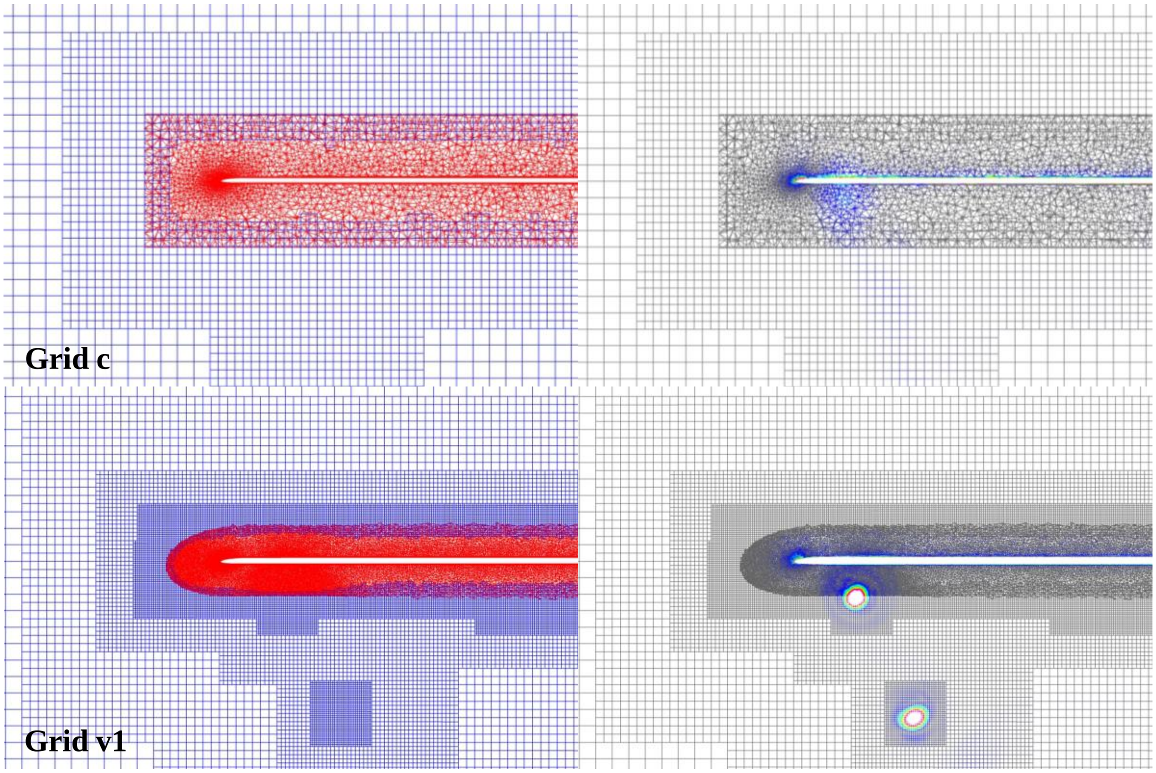

Figure 2: Cartesian grid comparison of the meshes (left) and vorticity contours (right).

the red represents the unstructured near-body grid and the blue represents the Cartesian off-body grid. The grid c case has a very coarse grid around the blade and out to the boundaries. This grid was left untrimmed in the simulation. The grid v1 case has a more refined blade and adds extra refinement around the blade tip and in the returning vortex region. This refinement extends to the near-body boundaries and is later trimmed for the simulation. It should be noted that the finest off-body cell size was tailored to the cell sizes on the near-body boundaries in both cases. Thus, the 'c' case has a coarser off-body mesh as well as a coarser near-body mesh. In the 'c' case, the tip vortex has dissipated to the point that the AMR is unable to detect this vortex resulting in an unstructured refinement pattern. For the 'v1' case, the near-body dissipation is minimized to a greater extent and the AMR is able to detect and refine the rotor wake. The 'v3' case yields images that look very similar to the 'v1' case in part because the Cartesian mesh settings are identical for the grid v1 and the grid v3 cases. The primary difference is the presence of a hub in the 'v3' case. This seems to be influencing the formation of a root vortex wake. It is important to note that for the finer 'v1' and 'v3' cases the resulting off-body mesh applies about 11

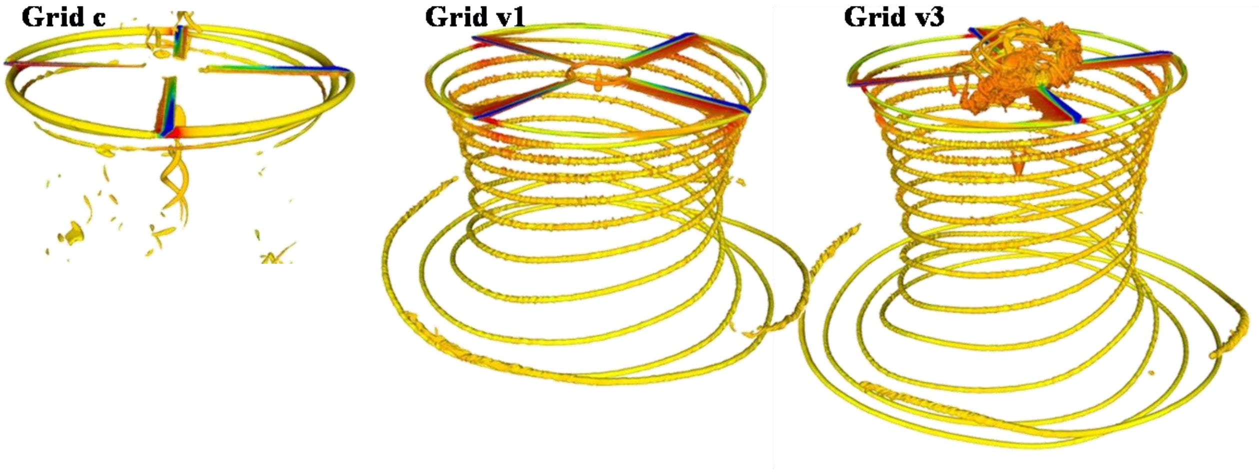

Figure 3: Comparison of isosurfaces of vorticity of magnitude 0.05 colored by density.

nodes across the vortex core as measured across the vertical velocity profile. The resulting wake structure computed using each grid is shown in Figure 3. In support of the previous argument, the isosurfaces of vorticity show that the 'c' case does not preserve the wake for the same distance as the 'v1' case.

The resulting performance predictions are plotted in Figure 4. The more refined cases produce better performance predictions with the 'v3' case producing the best results. These results show that not only is refinement around the blade tip needed, but the extra refinement added to the leading edge and trailing edge of the blade is also necessary in order to compute the correct hover performance parameters.

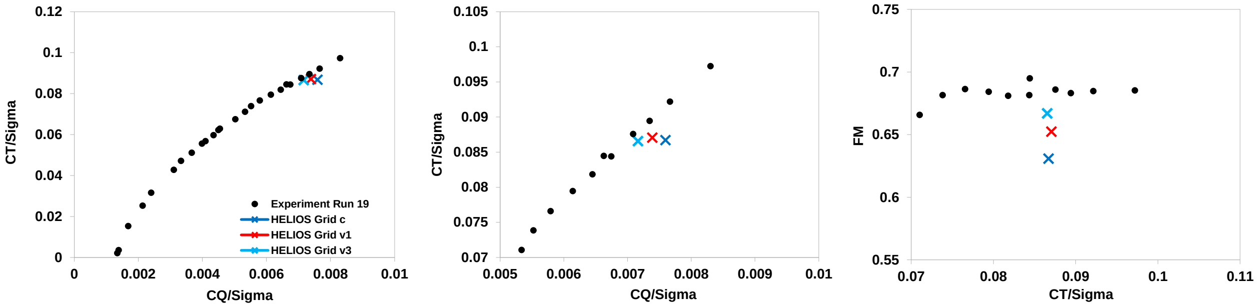

Figure 4: Comparison of figure of merit, thrust, and torque in HELIOS.

2. FUN3D

The FUN3D grid study compares all the grids from grid c to grid v3. Figure 5 compares the performance parameters computed using each grid. The 'v2' and 'v3' grids better predict the performance parameters. As with the HELIOS grid study these results support that inboard blade refinement is necessary when computing the hover performance. In particular, the extra refinement added around the leading edge in the 'v3' case does help to refine the blade loads.

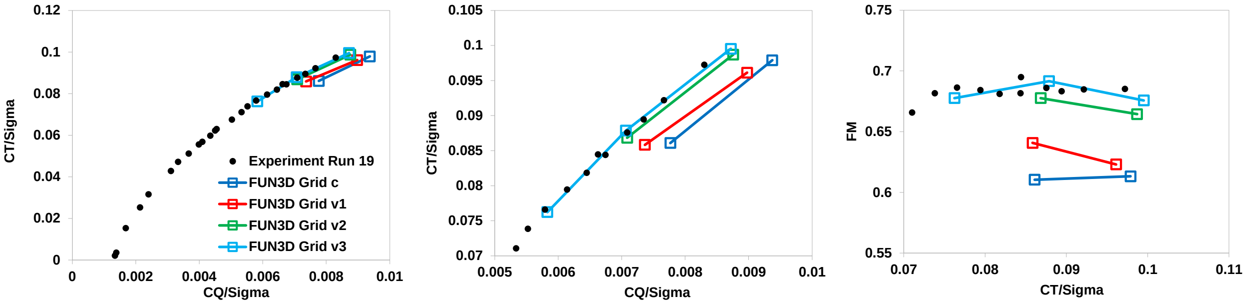

Figure 5: Comparison of figure of merit, thrust, and torque in FUN3D.

It should be noted that the blade grids are not the only factor that changes in the FUN3D comparisons. Modifications to the background grids were also made. For instance the 'c' and 'v1' cases both use the same background grid. This grid has a generic hub body and only a little refinement in the wake region. As the study progressed to the 'v2' case, more refinement was added in the wake region and for the grid v3 case the hub was modified to reflect the hub geometry later provided to the workshop participants. Thus, in addition to the blade grid

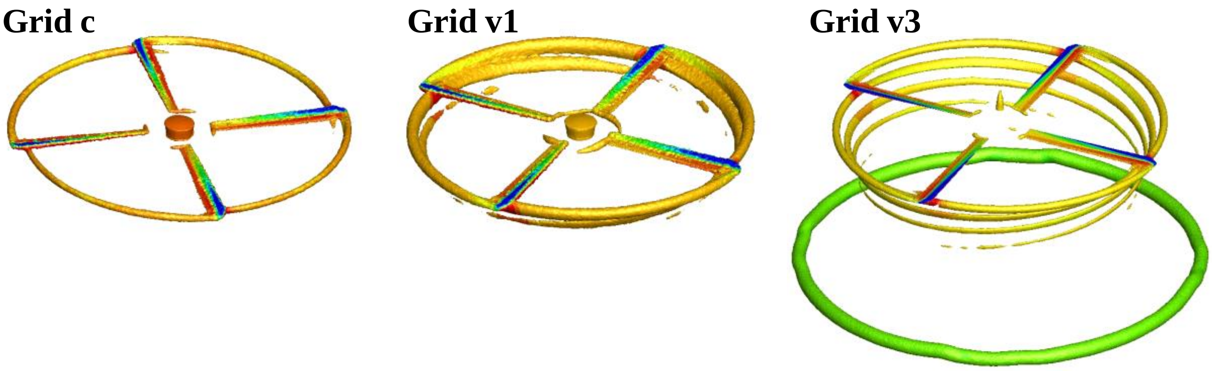

Figure 6: Comparison of isosurfaces of vorticity of magnitude 0.05 colored by density.

improvements, background grid improvements were also made as the study progressed. Figure 6 compares the wake preservation experienced by three of the stages in the grid study. Grid c and grid v1 use the same background grid, but grid v1 has better blade refinement which does have an impact on the wake preservation. Grid v3 has the same blade tip refinement as the grid v1, but much more refinement in the wake region and thus preserves the tip vortex for a longer distance. In particular, for the 'v3' mesh the wake refinement was such that the initial transient was preserved for a longer distance below the rotor.

B. FUN3D Extended Run Analysis

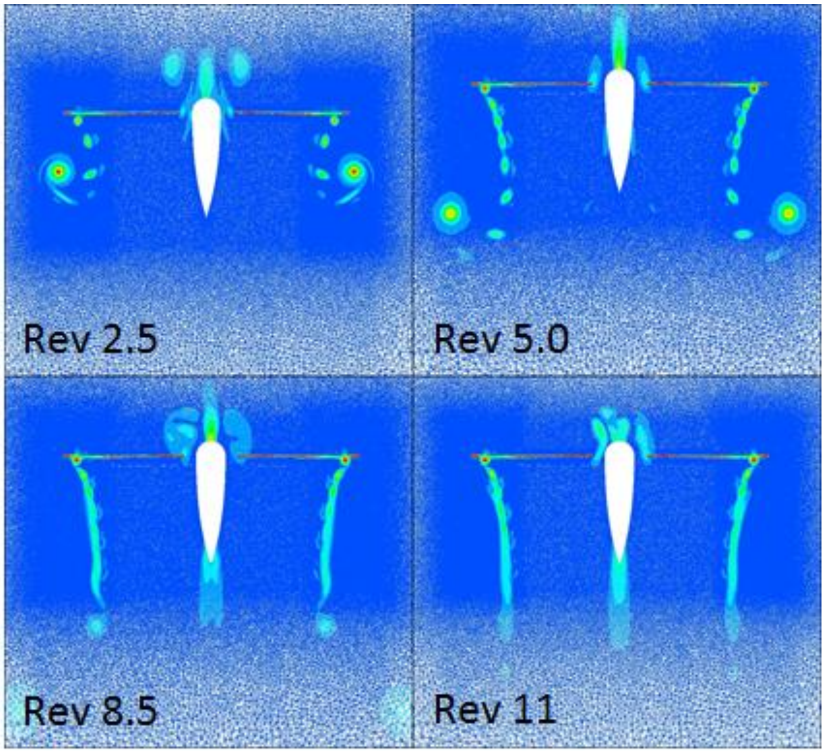

Figure 7: Extended FUN3D run showing wake dissipation over time.

An additional study was conducted in which the 'v3' case at a 9° collective angle was run for an extended period of time past convergence using FUN3D. This analysis was conducted to test the wake stability over time. Prior findings[4] have shown that many codes exhibit this behavior if hovering rotor cases are run over an extended period of time. Figure 7 shows the evolution of the wake using vorticity contours overlayed on the mesh. These plots show the region where the mesh was refined below the rotor. Note that after the transient leaves this region there are fluctuations seen in the wake around this boundary. These fluctuations are seen to dissipate the wake in this region. The dissipation then propagates back towards the rotor washing out most of the wake by 11 revolutions. In this case, the hypothesis is that this is a grid dependency issue caused by the change in grid density from fine to coarse in the region below the rotor. This causes a pseudo boundary effect that disrupts the wake formation the effect of which then propagates back towards the rotor. The current plan to test this hypothesis is to create an additional grid that extends the refined region further below the rotor and to more gradually coarsen the cells at the lower boundary of this region.

C. HELIOS Root Vortex Wake

An interesting side effect of the Cartesian off-body methodology is the development and preservation of a root vortex wake. In an unstructured methodology this root wake will typically diffuse as seen in Figure 7. After 11 revolutions there is still some upwash in the root cutout region with diffuse flow around the hub. No root vortices are preserved in this region. However, with the enhanced flow preservation capabilities of the off-body Euler solver this wake is preserved in the Cartesian mesh. Figure 8 illustrates the evolution of this phenomenon over time for the HELIOS 'v3' case. These images show that as the run progresses the root vortex wake extends above the rotor for some distance before spilling over onto the inboard sections of the blades after about eight revolutions. This vortex

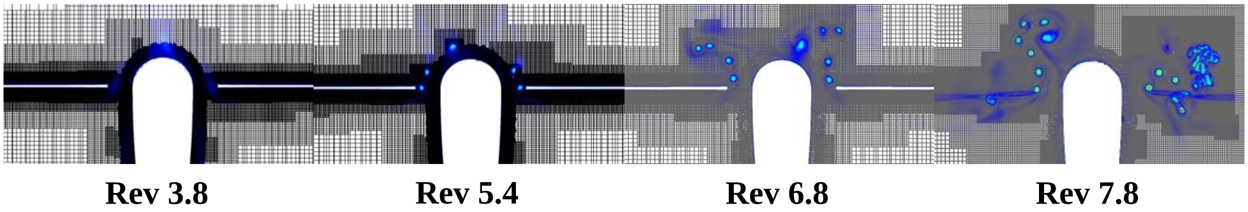

Figure 8: Development of the root vortex wake in HELIOS 'v3' case.

is initially above the rotor because of the initial upwash in the root cutout region. It is hypothesized that as the run progresses further that the wake will continue to move downward and eventually establish itself below the rotor. To prove or disprove this hypothesis this case is continuing to run for an extended period of time.

It is interesting to note that the matching 'v1' case that did not model the hub also did not develop the root vortex wake. Figure 9 shows the same vorticity contour slice for the HELIOS 'v1' case that did not model the hub. This case did not show any root vortex wake development even after six revolutions. However, it is not certain that the absence of the hub is the cause of this difference. Cases have been noted where even without a hub the root vortex wake was generated and preserved. Instead, this difference may possibly because the blade grid used for the 'v1' case has a coarser unstructured mesh in the root region than the blade grid used in the 'v3' case. This would cause the root vortex to diffuse in the unstructured mesh before it reaches the off-body mesh.

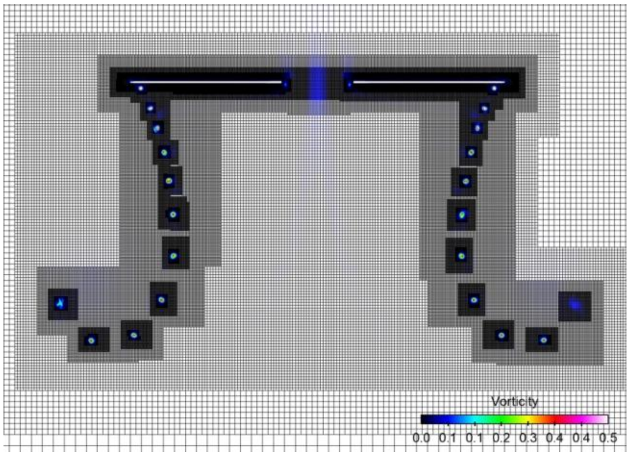

Figure 9: Root vortex wake in HELIOS 'v1' case.

D. Overall Solver Comparison

The results comparing FUN3D and HELIOS are shown in Figure 10. These results were obtained using grid c, grid v1, and grid v3 within both codes. Additional cases using HELIOS with OVERFLOW as the near-body solver are also included from reference 16. In general, the change from grid c to grid v1 to grid v3 is a positive change within both CFD codes. The HELIOS computation for a given grid provides a better prediction than the matching FUN3D computation with the exception of the 'v3' case where FUN3D seems to be predicting a better result. For the earlier cases this is hypothesized to be because the background grids used in FUN3D do not have much wake refinement in the coarser cases and the HELIOS off-body solver adds additional refinement through the AMR and also has higher spatial accuracy. Thus, the HELIOS cases consistently have better wake refinement relative to the matching FUN3D case. Looking specifically at the 'v3' cases, the FUN3D case has improved wake refinement and the HELIOS case uses the same settings as the 'v1' case. Thus, even though the HELIOS case still has the wake preserved over a longer distance, this creates a situation where the wake refinement is less of a factor. The fact that the performance predictions are not as good in the 'v3' HELIOS case is likely because the root vortex wake seen in Figure 3 has begun to spill over the inboard section of the blades. This probably stalled the loads convergence at a non-ideal location. If this wake spillage eventually moves below the rotor the converged performance parameters may improve. The HELIOS with OVERFLOW cases predict very good results on the order of the 'v3' cases for FUN3D. However, it should be noted that the OVERFLOW case did not model the hub.

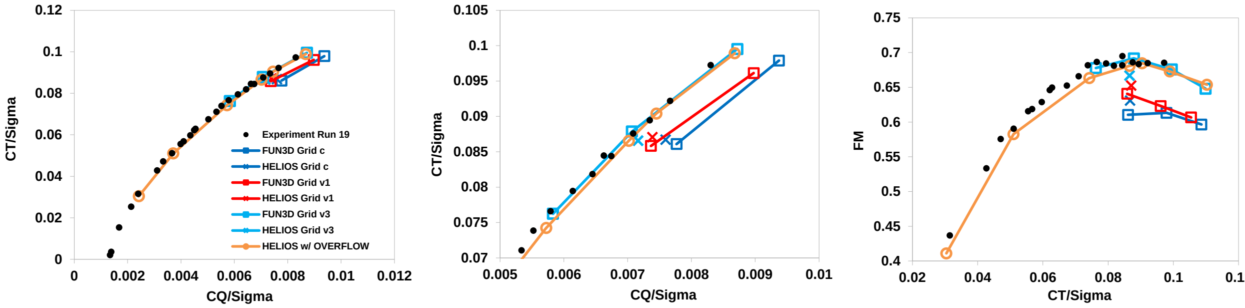

Figure 10: Comparison of figure of merit, thrust, and torque for matching cases in HELIOS and FUN3D.

A more detailed approach will be taken to look at the 'v3' cases in both FUN3D and HELIOS. Table 2 documents the wake position and the core size below the rotor as a function of wake age. The wake position was extracted by taking a slice of the solution through the center of the rotor intersecting the blade spanwise axis. The vortex centers were extracted from this slice by taking multiple horizontal slices through each vortex to identify the center more precisely. The results show that the position of the wake using each code is similar. Any differences are more easily seen by looking at the plots in Figure 11. These plots contain the same data as the table, but allow for a more direct comparison. Some differences to note are that the core size is consistently larger in the FUN3D solution than in the HELIOS solution and is increasing more rapidly with wake age. The wake positions in both cases are similar, but the HELIOS case has a slightly less wake contraction and remains closer to the rotor over time by a small amount. These differences are primarily a function of the differences in the solvers applied to the off-body solutions in each code; especially in the case of the core sizes. The HELIOS solution applies the Euler equations in a Cartesian mesh using 5th-order spatial accuracy with AMR. The FUN3D solution applies the Navier-Stokes equations using the SA turbulence model with the rotation correction in an unstructured mesh using 2nd-order spatial accuracy in a fixed mesh with additional manual refinement in the wake region. Thus, the HELIOS solution is less dissipative than the FUN3D solution as a function of wake age.

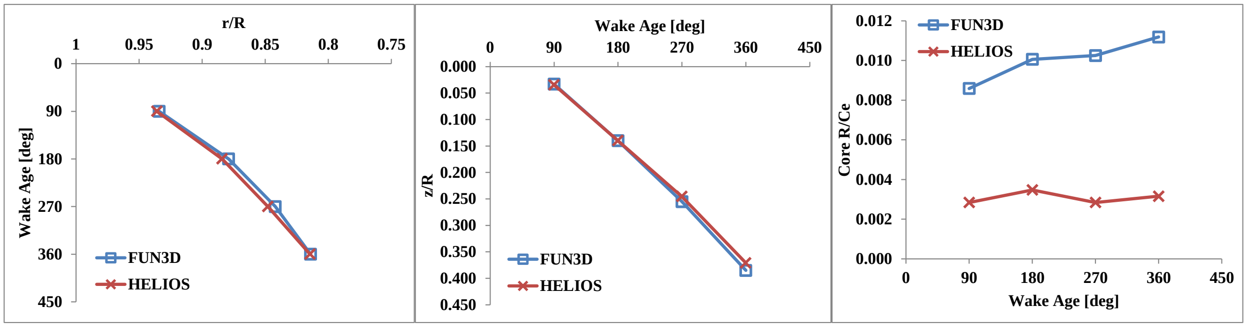

Figure 11: Code comparison of spanwise wake position, vertical wake position, and core size as a function of wake age.

E. Computational Metrics

The computational metrics give a good idea of how fast each code runs. However, the timing alone does not always give the full picture. This section compares both the computational time used as well as the solution convergence characteristics to provide a full picture of how fast a solution may be expected to complete. In order to provide a complete picture of the timing characteristics the convergence characteristics are discussed first. The ‘v3’ cases are compared for each code. Table 3 provides the time step size and subiteration parameters applied to the converged part of the cases. For the FUN3D case these parameters are not run throughout. The first two revolutions were run with 10 subiterations to accelerate the transient dissipation before a properly converged solution was obtained. The reason for using 50 subiterations in FUN3D is based not on the residuals, but on an analysis of the subiteration convergence of the forces and moments. While this analysis is a function of iteration, these loads were generally considered to have stopped changing at around 50 subiterations. The basis for selecting the 40 subiterations in HELIOS was a desire to reach one order of magnitude density residual convergence. This is the minimum acceptable convergence criteria to achieve 2nd-order temporal accuracy in the solution. However, the loads have not completely converged on the subiteration level after 40 subiterations, visually they look close to convergence. Currently, it is unclear how many more subiterations would be required to completely converge the forces and moments on the subiteration level or how much this would change the final solution. This analysis is currently considered future work.

Table 2. Wake position for the ‘v3’ cases in FUN3D and HELIOS.

	FUN3D			**HELIOS**		
Wake Age	r/R	z/R	Core R/Ce	r/R	z/R	Core R/Ce
90°	0.934	0.033	0.0086	0.936	0.035	0.0028
180°	0.879	0.140	0.0101	0.884	0.140	0.0035
270°	0.842	0.255	0.0103	0.848	0.245	0.0028
360°	0.814	0.385	0.0112	0.814	0.371	0.0032

Table 3. Convergence comparisons using grid v3 at a collective of 9°.

Code	Degrees of Freedom	Step Size	# Subiterations	Subiteration Convergence [orders of magnitude]
FUN3D	59M (nodes blades) 40M (nodes background)	1°	50	6
HELIOS	59M (nodes near trimmed) 80M (nodes off preAMR)	0.05°	40	1.4

The timing comparison shows a large difference between the two codes. FUN3D runs slower that HELIOS, but it takes 36% of the time HELIOS takes to run each revolution. Thus, even though FUN3D runs slower it takes less time to compute the final solution. It should be noted that there are multiple factors that would impact this overall run time. One is the number of processors chosen; FUN3D was run on almost double the number of processors as HELIOS. It is expected that if more processors were applied to the HELIOS case that faster run times would be obtained. The machine chosen is another consideration; each code was run on a different machine. Garnet is a Cray XE6 with 2.5 GHz AMD Interlagos Opteron processors and 64 Gb of memory per 32 core node while Lightning is a Cray XC30 with 2.7 GHz Intel Xeon E5 processors and 64 Gb of memory per 24 core node. The convergence criteria enforced should also be considered. If similar criteria had been applied to determine solution convergence in both codes then this would have caused HELIOS to complete later or FUN3D to complete sooner. The grids applied to each code also impact this run time. The Cartesian solver runs faster per node than the unstructured solvers. A final consideration is the possibility of running HELIOS in rotational mode. This would reduce the problem to a steady-state case in the hub frame of reference and may result in a quicker time to solution than using the time-accurate methodology. This option has not been pursued to date.

Table 4. Computational timing comparisons.

Code	Procs	Machine	Wall Clock [hrs/rev]	Iterations per rev	Time/proc/it/DoF [nanosec]
FUN3D	2048	Garnet	28	360	1.38
HELIOS	1032	Lightning	78	7200	0.27

VI. Conclusion

A study of two locally available CFD solvers (HPCMP CREATE^{TM}-AV HELIOS and FUN3D) are compared against experimental data. These two solvers are compared using variations in the grid and solver parameters. The conclusions from this study include,

- Grid independence studies show that having a blade surface with no apparent facets is not sufficient to compute the hover performance. The additional refinement added to the tip region, leading edge, and trailing edge of the blades is necessary to properly compute the hover performance.
- The advantage of the Cartesian solver is that it better preserves the rotor wake over a longer distance. This is achieved both through the higher spatial accuracy as well as the AMR methodology. This is shown to improve the performance predictions for a given blade grid. However, if insufficient near-body refinement is present in the blade grid, the dissipation will not provide enough information to the off-body solver for the AMR to work effectively.
- The direct comparison of the two solvers shows that the reduced wake dissipation in the HELIOS solution predicts a smaller core size overall than the FUN3D solution.
- Timing comparisons reveal that HELIOS runs faster, but a solution is obtained quicker using FUN3D. However, there are factors that could change the run time in both codes such as methodology chosen, machine run on, number of processors selected, and subiteration convergence criteria selected. Overall, the improved wake preservation available using the dual mesh methodology in HELIOS may make any extra time worthwhile.

Future work will be focused on completing analyses of alternate tip shapes. Matching KESTREL solutions will be run for comparison, but since KESTREL is a CFD code not previously validated for rotorcraft simulations a more comprehensive study will be pursued. This will include an independent grid study (since KESTREL is cell centered) as well as parameter studies. It is hoped that the availability of an implicit off-body solver will allow for larger time steps thus accelerating the time to solution on the order of the FUN3D results. This expanded comparison will take a closer look at the solutions including distributed parameters on the blades.

Acknowledgments

Material presented in this paper is a product of the CREATE-AV Element of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office. In addition, the authors would like to acknowledge the support of the supercomputing resources provided by the HPCMP, in particular the Air Force Research Lab (AFRL) and the Army Engineer Research and Development Center (ERDC). The blade models, test reports, and standardization guidance provided through the AIAA workshop are also gratefully acknowledged.

References

[1]Wissink, A.M., Sankaran, V., Jayaraman, B., Datta, A., Sitaraman, J., Potsdam, M., Kamkar, S., Mavriplis, D., Yang, Z., Jain, R., Lim, J., and Strawn, R., "Capability Enhancements in Version 3 of the Helios High-Fidelity Rotorcraft Simulation Code," *50th AIAA Aerospace Sciences Meeting*, Nashville, TN, January 2012.

[2]Morton, S., McDaniel D., Sears, D., Tillman, B., and Tuckey, T., "Kestrel: A Fixed Wing Virtual Aircraft Product of the CREATE Program," *47th AIAA Aerospace Sciences Meeting*, Orlando, FL, January 5-8, 2009.

[3]Lee-Rausch, E. M., Park, M., Nielsen, E., Jones, W., and Hammond, D., "Parallel Adjoint-Based Error Estimation and Anisotropic Grid Adaptation for Three-Dimensional Aerospace Applications," *23rd AIAA Applied Aerodynamics Conference*, Toronto, Canada, June 6-9, 2005.

[4]Hariharan, N. S., Egolf, T. A., and Sankar, L., N., "Simulation of Rotor in Hover: Current State and Challenges," *52nd Aerospace Sciences Meeting*, National Harbor, MD, January 13-17, 2014.

[5]AIAA Applied Aerodynamics Rotorcraft Working Group, "https://info.aiaa.org/tac/ASG/APATC/Web%20Pages/RotorSim-DG_Info.aspx," Last Accessed November 3rd, 2014.

[6]Balch, D. T., and Lombardi, J., "Experimental Study of Main Rotor Tip Geometry and Tail Rotor Interactions in Hover. Volume 1: Text and Figures," NASA Contractor Report 177336, February 1985.

[7]Balch, D. T., and Lombardi, J., "Experimental Study of Main Rotor Tip Geometry and Tail Rotor Interactions in Hover. Volume 2: Run Log and Tabulated Data," NASA Contractor Report 177336, February 1985.

[8]Sitaraman, J., Floros, M., Wissink, A., and Potsdam, M., "Parallel Domain Connectivity Algorithm for Unsteady Flow Computations Using Overlapping and Adaptive Grids," *Journal of Computational Physics*, 229 (2010) pp. 4703-4723.

[9]Saberi, H., Khoshlahjeh, M., Ormiston, R., and Rutkowski, M., J., "Overview of RCAS and Application to Advanced Rotorcraft Problems," *4th AHS Decennial Specialists' Conference on Aeromechanics*, SanFancisco CA, January 21-23, 2004.

[10]Mavriplis, D. J., and Venkatakrishnan, V., "A Unified Multigrid Solver for the Navier-Stokes Equations on Mixed Element Meshes," *International Journal for Computational Fluid Dynamics*, Vol. 8, 1997, pp. 247-263.

[11]Wissink. A., Kamkar, S., Pulliam, T., Sitaraman, J., and Sankaran, V., "Cartesian-Adaptive Mesh Refinement for Rotorcraft Wake Resolution," *28th AIAA Applied Aerodynamics Conference*, Chicago, IL, June 2010.

[12]Pulliam, T. H., "Euler and Thin-Layer Navier-Stokes Codes ARC2D and ARC3D," *Computational Fluid Dynamics Users Workshop*, University of Tennessee Space Institute, Tullahoma TN, March 1984.

[13] Wissink, A. M., Hornung, R. D., Kohn, S., Smith, S., and Elliott, N., "Large-Scale Parallel Structured AMR Calculations using the SAMRAI Framework," *Proceedings of Supercomputing 2001*, Denver, CO, November 2001.

[14] Kamkar, S. J., Jameson, A., Wissink, A. M., and Sankaran, V., "Automated Off-body Cartesian Mesh Adaptation for Rotorcraft Simulations," *49th AIAA Aerospace Sciences Meeting*, Orlando, FL, January 4-7, 2011.

[15] Buning, P. G., Jespersen, D. C., Pulliam, T. H., Klopfer, W. M., Chan, W. M., Slotnick, J. P., Krist, S. E., and Renze, K. J., "OVERFLOW User's Manual Version 1.8m," Tech. rep., NASA Langley Research Center, 1999.

[16] Jain, R. K., and Potsdam, M. A., "Hover Predictions on the Sikorsky S-76 Rotor using HELIOS," *52nd Aerospace Sciences Meeting*, National Harbor, MD, January 13-17, 2014.

[17] Noack, R., "SUGGAR: A General Capability for Moving Body Overset Grid Assembly," *17th AIAA Computational Fluid Dynamics Conference*, Toronto, Canada, June 2005.

[18] Noack, R., "DiRTlib: A Library to Add an Overset Capability to Your Flow Solver," *17th AIAA Computational Fluid Dynamics Conference*, Toronto, Canada, June 2005.

[19] Park, M. A., "Adjoint-Based Three-Dimensional Error Prediction and Grid Adaptation," *AIAA Journal*, Vol. 42, No. 9, 2004, pp. 1854-1862.