Visualization of Large-Scale Unsteady Computational Fluid Dynamics Datasets

EARL P.N. DUQUE
and
STEVE M. LEGENSKY
Intelligent Light

This paper represents an entry to the SC|05 Analytics Challenge and describes a new scientific visualization tool designed to allow investigators to more effectively interrogate large unsteady datasets. By utilizing techniques such as data extracts and programmable graphics processing units, this new capability allows users to more freely interrogate time accurate Computational Fluid Dynamics (CFD) solutions in excess of $10^8$ grid points. This tool known as the Advanced Technology Viewer (ATViewer) has been incorporated into Intelligent Light’s FIELDVIEW 11 product and applied here to interrogate two large unsteady datasets on the order of 60 gigabytes in size. The first is an unsteady Reynolds Averaged Navier-Stokes simulation of a wind turbine operating at a high yawed position. The other is a hybrid Large-Eddy simulation of an elastic helicopter rotor in forward flight.

Categories and Subject Descriptors: J.2 PHYSICAL SCIENCES AND ENGINEERING – Aerospace; I.3.7 Three-Dimensional Graphics and Realism; I.3.6 Methodology and Techniques – Graphics data structures and data types; I.3.6 Methodology and Techniques – Interaction Techniques

General Terms: Performance

Additional Key Words and Phrases: Computational Fluid Dynamics, scientific visualization

1. INTRODUCTION

Intelligent Light, through an Army Phase II SBIR, has developed a new scientific visualization tool designed to allow investigators to more effectively interrogate large unsteady datasets. By utilizing techniques such as data extracts and programmable graphics processing units, this new capability allows users to more freely interrogate time accurate Computational Fluid Dynamics (CFD) solutions in excess of $10^8$ grid points. This tool, known as FIELDVIEW Advanced Technology Viewer (ATViewer), has been applied to interrogate two large unsteady datasets on the order of 60 gigabytes in size. The first is an unsteady Reynolds Averaged Navier-Stokes simulation of a wind turbine operating at a high yawed position. The other is a hybrid Large-Eddy simulation of an elastic helicopter rotor in forward flight illustrated in Figure 1. By utilizing unsteady iso-vorticity surfaces and streak point patterns the development of blade

* This research was supported by a U.S. Army SBIR Phase 2 Contract W911W6-04-C-0007 entitled “Next Generation system for Efficient Remote Parallel Visualization of Large Scale Time-Dependent CFD Data”

Authors’ addresses: Author 1, Manager of Applied Research,; Author 2, General Manager/Founder, Intelligent Light, 301 Rt 17N – 7th Floor, Rutherford, N.J., 07070.

Permission to make digital/hard copy of part of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication, and its date of appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee.

SC/05 November 12-18, 2005, Seattle, Washington, USA
© 2005 ACM 1-59593-061-2/05/0011 $5.00

-1-
stall regions and the development of the rotor wakes are readily investigated. Unlike other methods that allow only single, fixed camera angles, the ATViewer readily allows exploration of the flowfield at user interactive rates.

2. METHODS AND TOOLS

Intelligent Light has been actively researching the topics of large data management since the mid ‘90s and has consistently deployed new technologies into its product - FIELDVIEW. Intelligent Light’s new Advanced Technology Viewer (ATViewer) represents a new philosophy in post-processing large and unsteady CFD cases, in that it is designed from the ground up for unsteady data. The goal is interactive exploration so that the engineer or researcher can gain insight into complex unsteady phenomena. The primary roadblock to achieving this goal is the size of the CFD data. Because the size of these datasets is commonly in the tens of gigabytes (and sometimes up to several terabytes), it is impractical to consider having them resident in memory and it is often prohibitive to have them on disk! (Of course, there are large scale architectures such as the SGI Altix or Prism that can support that much physical RAM in a single system image).

The ATViewer uses a principle recognized by Robert Haimes of MIT [2] and Al Globus of the NAS group at NASA Ames[1,2]: “extracts”. CFD datasets require a large number of grid points and cells in order to obtain a high quality solution. However, in the post-processing phase, researchers are usually interested in a much smaller volume of data. For example, if the subject of study is surface pressure behavior, only the surface data is needed in the post-processor. So, one can extract the surfaces from each timestep of the simulation and post-process only those. This amounts to a form of data management – taking direction from the user as to the data needed for post-processing and then handling only those extracts that are required.

As illustrated by Figure 2, a large CFD dataset will consist of upwards of 100’s of GBytes of data that a user will need to access throughout the visualization process. “Extracts” are created of the dataset. These extracts may consist of graphical elements such as iso-surfaces, cutting planes, computational surfaces and/or particle paths. These
extracts may contain multiple scalar and vector values so that the user may have some control on what values to
display ranging from contour lines, surfaces or other scalar coloring schemes. The extracts are created via the
FIELDVIEW product which saves the extracts to disk. ATViewer then uses these extracts or extract databases to
render interactive images.

Another important feature of the ATViewer is the use of programmability features in commodity graphics cards.
The technology in these cards is advancing as rapidly or perhaps more rapidly than that of general purpose CPUs.
We take advantage of that processing power for more than just drawing polygons whose vertices are colored
corresponding to a scalar color map. If we use vertex colored polygons, they need to be retransmitted from
workstation memory into the graphics card each time the user changes the scalar variable or the color map range. In
the ATViewer, texture mapping techniques are used to accomplish the vertex color changes – if the variable remains
the same but the color map range changes, only the texture map need be resent to the card, which is infinitely
smaller than the vertex data. Similar techniques are used in the ATViewer for streamline animation and surface
thresholding, in which geometry can be clipped away depending on the value of a scalar variable on the surface.

Data management is critical to the productive use of large datasets. CPU cache access speeds are many 10s of
thousands of times faster than disk I/O speeds. Graphics bus speeds can be many thousands of times slower than
graphics processing unit (GPU) speeds. Without proper management, the flow of excessive data can greatly limit
the response speeds that a user experiences when interacting with data. Figure 3 illustrates the relationships between
data size and data access speed for the operations of FIELDVIEW and the ATViewer.

There have been implementations of the extract model in the past. Some have contained ‘hardwired’ contents
for each extract (for example, a cutting plane
has fixed color contours corresponding to a
specific scalar variable). Others have
allowed multiple variables to be stored with
the extract, but the extract format is not
optimized for interactive access speed. ATViewer extracts can contain as many
variables as the user desires and they are
stored in a database that is optimized for
graphics speed. In a sense, this is another
level in the hierarchy of data management.

Finally, ATViewer is designed as a multi-threaded application. Multi-threading is a technique in which an
application is organized as separate ‘sub programs’ all sharing the same virtual data space. There are two reasons
why multi-threading is important in the ATViewer design: interactivity and parallel operation. In order to be able
to zoom and rotate the data while a transient sequence is playing, one can put the time stepping into one thread and
the mouse/zoom/rotate handler into a separate thread. These two threads communicate with one another via queued
requests, which permit asynchronous operation. This is an oversimplification, but the point is clear – you can do more than one operation at what the human operator thinks is the ‘same time’.

The multi-threading capability gives the appearance of simultaneous execution if only one CPU is available. However, if there is more that one CPU in a workstation, threads will automatically migrate to available CPUs in a given system. So, if you have a dual CPU system, two threads from the list of all available threads can be operating in parallel. In this challenge project submission a dual processor-dual core AMD Opteron™ workstation was used where four threads executed simultaneously. This provides greater throughput to the user and it takes advantage of the trends in current computer development: multiple CPUs in a system or even on a single chip.

![Figure 4 – Wind Turbine Grid Details](image)

3. DATASET DESCRIPTION

The unsteady datasets used to demonstrate the capability of these tools are unsteady Navier-Stokes and Hybrid Large Eddy Simulations (HRLES) of wind energy applications and rotorcraft, respectively. The flow solver code used is the OVERFLOW code by Buning Et.al. [5]. The baseline turbulence simulation method utilizes a Spalart-Allmaras RANS turbulence model [6] in the near-wall and an HRLES method following Kok et al [7].

The accurate prediction of the noise characteristics of current and next generation wind turbines and helicopters require a detailed knowledge of the unsteady aerodynamic flow field around the rotor blades. In turn, the aerodynamics and aeroacoustics modeling both require an accurate description of the blade position, which involves an accurate modeling of the aeroelastic response of the blade surfaces. The blade position is affected by the control
setting imposed by the pilot or automatic control systems to trim the machine. For most machines, the trim conditions do not remain constant, particularly during aggressive maneuvers or high load operations. Thus, the aerodynamics, aeroelastic response, and trim, that ultimately affect the aeroacoustics of the machine, are very tightly coupled and interdependent.

The visualization challenge presented here consists of handling large unsteady datasets with dynamic and elastic motion between component parts. The method needs to allow the user to quickly interrogate the dataset and to explore the unsteady fluid dynamic behavior of an HRLES predicted flowfield whilst the object rotates, bends, twists and translates. The datasets used in this challenge project utilize 4.0 to 18.6 million grid points to properly resolve and capture the aerodynamic loads.

The grid systems used in these rotor computations consist of an overset grid system. These systems consist of near body and off body grids as illustrated in Figure 4 for a wind turbine rotor. This particular near body curvilinear grid system consists of 12 grids to fully describe two blades and a simplified hub attachment. Figure 4b and c illustrate the rotor’s surface grid near the tip region and the curvilinear mesh for the main parts of the rotor. The near body grid is then surrounded by a series of off body Cartesian grids which carry the solution to the farfield.

During the flow solving process, the near body grids are caused to move through the off body grids. As the near body grids move relative to the off body grids, the grids communicate and pass boundary information between corresponding overset grids. The motion of the near body grids includes both rigid body rotation modes and elastic motions that cause deformation of the near body grids. The body motion and grid deformations place an exceptional challenge to the visualization techniques employed.

4. VISUALIZATION RESULTS

To date, the FIELDVIEW ATViewer has been tested on various platforms. The following table lists a few of the platforms.

<table>
<thead>
<tr>
<th>Graphics Board</th>
<th>System</th>
<th>OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nvidia Quadro FX3400</td>
<td>Dell 470N Workstation</td>
<td>Red Linux</td>
</tr>
<tr>
<td>Nvidia Quadro FX1400</td>
<td>Dell 470 Workstation</td>
<td>Windows/XP</td>
</tr>
<tr>
<td>Nvidia Quadro FX3000</td>
<td>AMD Dual Opteron™</td>
<td>SuSe Linux</td>
</tr>
<tr>
<td>ATI Mobility FireGL V3100</td>
<td>Dell M20Laptop</td>
<td>Windows/XP</td>
</tr>
<tr>
<td>ATI Radeon 9700 Pro</td>
<td>Dell Workstation</td>
<td>Windows/XP</td>
</tr>
<tr>
<td>GeForce FX5700LE</td>
<td>Dual Processor- Dual Core AMD Opteron™</td>
<td>Redhat EL4</td>
</tr>
</tbody>
</table>

The visualization challenge presented here utilizes data extracts that were created within the FIELDVIEW product and then explored utilizing the new ATViewer. Two datasets were considered. Both sets consist of large unsteady datasets with dynamic and elastic motion between component parts. These datasets utilize overset structured grids and are solved using an unsteady Reynolds-Averaged Navier-Stokes and a version of the same code.
that utilizes a hybrid Large Eddy Simulation methodology. Some details of the grids utilized are provided in the extended abstract. The following table lists the grid sizes.

<table>
<thead>
<tr>
<th></th>
<th>Number of Overset Grids</th>
<th>Total Grid points (million)</th>
<th>Data Size per rotor rotation (GBytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wind turbine</td>
<td>46</td>
<td>18.6</td>
<td>59.7</td>
</tr>
<tr>
<td>Helicopter Rotor</td>
<td>59</td>
<td>4.61</td>
<td>31</td>
</tr>
</tbody>
</table>

Rotor wake development and dynamic blade stall effects are important fluid dynamic phenomena that affect the performance of both wind turbine as well as helicopter rotors. On any lifting device, a tip vortex and vortical wake sheet will form behind the device. For a rotor, this wake structure forms a helicoidal shaped structure behind it. The interaction between the wakes and the rotor blades has a large effect upon the unsteady air loads. Experiments have been performed to visualize and understand these interactions, but the fidelity of wind tunnel experiments and any in-field experiments is highly limited. In stalled flow conditions, a region of separated flow will form, shedding vortices from the lifting surface in a coherent manner allowing for local lift coefficients to exceed the static maximum lift coefficient. By utilizing CFD solutions and visualization tools, such as FIELDVIEW, one can explore the CFD data to understand the wake interactions. This process is very tedious however using a static viewpoint such as was used in previous versions of FIELDVIEW. Using data extracts, that were created in FIELDVIEW, the ATViewer allows one to more readily and interactively explore the unsteady datasets from any camera angle desired.

The following images illustrate the use of three fundamental data extracts available within FIELDVIEW and the ATViewer – computational boundary surfaces as illustrated by the blade surface, point particles as illustrated by fixed release rakes and moving rakes, and iso-scalar surfaces such as vorticity.

Figures 5 illustrate the fixed point release rake and the moving rake. In the fixed release rake, rakes are formed in the form of a cross just ahead of the rotor. Note that the wind flow is into the paper. Particles are then released for a total of five rotor revolutions. In the second streak image, rakes...
were defined such that they were attached with the motion of the rotor blade. Rakes were defined at the rotor tips and just above the rotor surface, within the boundary layer and just aft of the blade trailing edge. In both rake releases, on the order of 400 particles are released every 5 degrees of rotation and propagated through the simulation for 5 rotor revolutions. These extract particles required on the order of 1 hour of wall time to generate on the dual-core AMD Opteron™ for a total particle count of 144,000 particles.

Once the extracts are rendered in ATViewer, the investigator can move the rotor to any arbitrary position to gain better understanding of the flowfield. These streak patterns lead to the better understanding of the wake path of this particular rotor while operating in yaw. It brought to light that an inboard vortex may be affecting the rotor airloads. The particles themselves do not completely reveal the wake interactions and the source of these interactions. These observations lead to the need for iso-vorticity surfaces.

The isovorticity surfaces shown in the Figure 6 were generated at two levels of vorticity magnitude. The outer resulting surface is rendered using transparency and colored by the scalar velocity magnitude while the other value uses a solid color of red. This technique allows one to visualize the formation of the wake structure; both the vortex wake sheet and the resulting tip vortices. Sweeping through the extracts in time yielded even further evidence that there was an interaction between the wake from one blade onto the other. However, the fact that the sweeps through time were performed in an inertial reference frame made it difficult to see the interaction; after all, the rotor is spinning at interactive rates as it would in real life.

![Figure 7 – Isovorticity surfaces in rotating frame](image)

To get around this difficulty, the dataset was recast in a rotating reference frame by transforming the grid and the solution by an appropriate angle that was keyed to the time step. Figure 7 illustrates the resulting surfaces. The image on the left shows a tip vortex structure off the extreme tips and a wake sheet formation. The image on the right shows another time instance and reveals the formation of an inboard vortex forming on the lower blade that tends to curl up and flow into the inboard sections of the upper blade. The resulting forces that were computed from this dataset indicated a wake interaction, the visualizations that were generated by ATViewer allowed this investigator to rapidly find the cause.
The data extract, parallel threads, programmable GPU's capabilities all result in an interactive CFD visualization capability that greatly assists the engineer and scientist in the exploration of large unsteady datasets. All the solutions rendered for this paper were produced at frame rates ranging from 15-20+ frames per second using extracts from unsteady CFD datasets totaling upwards of 60Gbytes. Many of the graphics consisted of up to 140,000 triangles, 45,000 quads and up to 121,000 vertices. Figure 8 illustrates another solution requiring these high interactive rates to fully understand the flow- the HRLES solution of a helicopter rotor in forward flight. This unsteady data, without the use of ATViewer, would require many weeks to obtain arbitrary viewpoints and to fully explore the dataset at truly interactive rates.

5. CONCLUSIONS

A new CFD visualization tool specifically designed for large unsteady datasets has been successfully developed and deployed. The ATViewer, part of Intelligent Light’s FIELDVIEW 11 product, makes use of data extracts and multi-threading to maintain interactive interrogation of unsteady data. Two large unsteady datasets: a wind turbine and an helicopter rotor in forward flight, demonstrated the code’s capabilities by maintaining interactive frame rates in the range of 15-20 frames per second on surfaces extracted from CFD datasets on the order of 60Gbytes in size. The methods used here shall greatly improve the ability to readily visualize and gain better understanding of the even larger datasets we face in the near future.

ACKNOWLEDGMENTS

The authors extend gratitude to their colleagues and the development team at Intelligent Light: Mike Schechterman, Principle Investigator; Developers: Gordon Acocella, Dan Hiepler, Don Leich, Liz Martin and Ken Wall.

REFERENCES


