Visualization of a Simulated Long-Track EF5 Tornado Embedded Within a Supercell Thunderstorm

Leigh Orf
Central Michigan University
Mt Pleasant, MI, USA
leigh.orf@cmich.edu

Robert Wilhelmson
National Center for Supercomputing Applications
Urbana, IL, USA
wilhelms@illinois.edu

Louis Wicker
National Severe Storms Laboratory
Norman, OK, USA
louis.wicker@noaa.gov

***Abstract*—Tornadoes are one of nature's most destructive forces, creating winds that can exceed 300 miles per hour. The sheer destructive power of the strongest class of tornado (EF5) makes these tornadoes the subject of active research. However, very little is currently known about why some supercells produce long-track (a long damage path) EF5 tornadoes, while other storms in similar environments produce short-lived, weak tornadoes, or produce no tornado at all.**

In this work we visualize cloud model simulation data of a supercell thunderstorm that produces a long-track EF5 tornado. Several obstacles needed to be overcome in order to produce the visualization of this simulation, including managing hundreds of TB of model I/O, interfacing the model output format to a high-quality visualization tool, choosing effective visualization parameters, and, most importantly, actually creating a simulation where a long-track EF5 tornado occurs within the model, which only recently has been accomplished.

I. Introduction

Tornadoes remain the subject of research due to the damage they produce every year. Forecasting the occurrence of tornadoes prior to formation has proved very difficult, and the behavior of tornadoes once they are formed remains poorly understood. Field studies have increased our understanding of the inner workings of supercell thunderstorms that produce tornadoes. These observations have given researchers a closer, more detailed look at the physical (related to pressure, moisture, temperature, and cloud/precipitation) and dynamic (related to wind) features and associated with tornado formation (tornadogenesis) and have somewhat elucidated the structure of the tornadoes that have formed in observed storms.

While observations of tornadoes and the storms that form them are crucial in order to improve our understanding, numerical simulation is also a powerful tool that can be used to provide insight into the inner workings of tornado-producing thunderstorms. In this work we report on a cloud model simulation executed on the Blue Waters supercomputer where a simulated supercell produces a long-track EF5 tornado. "Long-track" refers to an unusually long damage path length of a tornado. Over the period from 2002-2012, 90% of tornado paths documented in the United States were shorter than 8 miles in length, and only 1% exceeded 30 miles in length (Harold Brooks, personal communication). Long-track tornadoes are of special interest researchers due to their potential for creating unusually large amounts destruction along their path. A long-track EF5 (winds exceeding 200 miles per hour, the strongest category on the Enhanced Fujita scale) tornado would therefore potentially be the most damaging type of tornado.

We present volume-rendered visualizations of an ultra-high resolution simulation of a supercell that produces a long-track EF5 tornado. The simulated tornado is on the ground for over 90 minutes, and produces a damage path exceeding 60 miles.

II. Simulation

CM1 [2] is three-dimensional, non-hydrostatic, non-linear, time-dependent numerical model designed for idealized studies of atmospheric phenomena. The simulation described herein utilized 20,000 computing cores. A hybrid MPI/OpenMP model was used where each shared-memory node contained 16 MPI ranks, each of which spawned two OpenMP threads. OpenMP was used primarily to pararellelize triply-nested loops which comprise much of the CM1 model (as it is integrating in three spatial dimensions). This configuration was found to be the fastest compared to other configurations, such as reducing of the MPI ranks and an increasing in the number of OpenMP threads per MPI rank.

The simulation was initialized with environmental conditions that occurred adjacent to an observed supercell thunderstorm that produced a long-track EF5 tornado on May 24, 2011, outside Oklahoma City, Oklahoma [3]. To the best of our knowledge, this work represents the first time a long-track EF5 embedded within a supercell has been simulated. The fact that the simulation is based upon an actual event that was observed in the field with research-class Doppler radar [1] presents an opportunity to compare an observed storm to a simulated storm in a meaningful way.

III. I/O MANAGEMENT

CM1 version 16, used in this study, contains several I/O options, including the option where each MPI rank writes a single history file (containing the three-dimensional model state) at selected intervals. With this method, if one were to save the model state every 5 model seconds, a total of 10,000 files would be written to a directory, 20 times per model minute, for approximately 180 model minutes, creating 36 million small files in a single directory for typical simulation. Even if one creates a new directory for each time data is dumped, the resultant number of files is staggering and potentially overwhelming. Furthermore, the latency involved in having each rank a single file per time dump to a Lustre file system on a heavily-utilized shared resource is very high, resulting in unsatisfactorily long wallclock model run times where the model spends the majority of its time doing I/O.

In order to address these issues, different approaches were attempted in order to both reduce the number and frequency of files written to disk. Experiments with parallel HDF5 indicated that while improvements could be achieved by utilizing parallel writes there were serious issues with latency and poor aggregate throughput where one or two ranks would sometimes "get stuck" and hold back the completion of the writing of a file, and consequently hold up the entire simulation.

Significant performance improvements were found by utilizing the core memory driver of (serial) HDF5 which enables the ability to write files to memory rather than to disk. This buffered-write approach eliminates the latency issues associated with frequently writing to disk, as memory writes are significantly faster. In order to take advantage of the large amount of available memory on Blue Waters, multiple time levels (on the order of 50) were written to each HDF5 file in memory before flushing to disk and freeing up memory. A further reduction in the number of files written per dump to disk was achieved by assigning one MPI rank per shared memory node as an I/O rank, and collecting 3D data on each node to the I/O rank which allowed the file to grow in memory until flushing to disk.

With this approach, we reduced the number of files per simulation by a factor 800, buffering 50 times per file and writing 1 file per shared-memory node rather than 16. This results in 45 thousand files per simulation, with file sizes around 1-2 GB per file. These file are not stored in a single directory but are spread out over dozens of directories. This configuration showed dramatic I/O throughput improvements on Blue Waters as compared to default CM1 output options. A more detailed description of the I/O approach utilized in this study can be found in [4].

IV. VISUALIZATION

VisIt [6] is a visualization and analysis tool supported on Blue Waters. VisIt can utilize massively parallel hardware such as Blue Waters by spawning compute engines across many compute nodes and displaying visualized data to a client running on another machine. VisIt also contains a robust python programming environment that allows visualizations to be created with scripts without use of a client GUI. We chose VisIt based upon its programming/development environment, features, performance, and the fact that the software is actively developed.

There are different paths available for those who wish to display their data with VisIt. The "plug-in" approach was chosen for this work, which requires the development of code (primarily C++) that provides VisIt with a direct interface to the CM1 HDF5 model output format without any file conversion, which would be unwieldy and require huge amounts of additional disk space.

After the plug-in was developed and tuned, VisIt was utilized interactively to explore the three-dimensional structure of the supercell and tornado, and then run in batch mode to produce frames for animation. The plug-in was specifically developed such that only a subset of the full model domain could be chosen to visualize. This drastically reduced the amount of computation resources required when focusing on a small subsection of the storm (i.e., the tornado and its surrounding environment), but also allowed for full-domain visualization.

In order to achieve high volume rendering fidelity with VisIt, 4,000 samples per ray were used for each frame. Using such a value caused VisIt to exhaust memory unless VisIt's engine ran with only one serial process. In order to render a total of 7,200 frames for both animations in a timely manner, parallelization was achieved by rendering many frames concurrently. On Blue Waters, a total of about 30 wallclock hours was required to render 7,200 volume rendered frames at 1080p resolution (1920x1080) using this approach.

Thus far we have focused our visualization efforts on cloud, rain, and vorticity fields. A storm's cloud and rain comprise much of what is visible to the human eye. This "photorealistic" approach is important, as photography and video are tools utilized in field studies of severe storms. Vorticity (the curl of the wind vector) is large in regions of large wind shear and rotation. Vorticity is a useful quantity to visualize in a tornadic supercell thunderstorm where rotation of the air (e.g., the mesocyclone, tornado, and other weaker but abundant vortices) is fundamental to the storm.

A. Cloud and rain

VisIt can only volume render one field in an image. In order to create imagery where both cloud and rain were visible, a VisIt expression was created that compared the cloud mixing ratio (qc) to the rain mixing ratio (qr) at each model grid zone and returned the largest of the two - but returned a negative value if cloud mixing ratio was larger. Because mixing ratios are positive-definite, this allowed a single three-dimensional field to contain both qc and qr

without overlap between the two fields. A different color and opacity map could therefore be applied to cloud and rain while visualizing both in a single frame. This approach, however, is not without limitations, as it operates under the assumption that only rain or cloud may be visible at any given location in space. However it succeeds because naturally, rain inside a cloud will be masked by the surrounding cloud as in nature, and rain shafts falling from a cloud will be visible as well. By choosing a slightly different color map for cloud and rain, the differences between the two are readily visible.

Figure 1, several minutes following tornadogenesis, demonstrates the effectiveness of this visualization technique for creating something resembling a photorealistic presentation. Several features observed in the field are presented in this figure, including a smooth, laminar cloud (part of the mesocyclone), shafts of rain in the storm's rear flank, a lowering in the cloud base called a wall cloud, a feature from which tornadoes typically descend as they form, a tail cloud, and the tornado. In addition, a very thin curtain of rain associated with the storm's hook echo can be seen encircling the tornado and the wall cloud. This thin sheet of rain is a feature that has been observed in recent field studies and which may play a role in some cases of tornadogenesis. Later in the simulation, when the tornado becomes engulfed by rain, the degradation in visual identification of the tornado is faithfully reproduced, producing only a vague representation of the tornado behind the heavy rain [5]. Tornadoes that are "rain-wrapped" are not uncommon when spawned by high-precipitation supercell thunderstorms, and this presents a special hazard as the tornado is not visible to the naked eye as the optically opaque rain masks the danger lurking within.

B. *Vorticity*

In Figure 2, the magnitude of the three-dimensional vorticity vector is visualized using volume rendering. The rendering focuses on moderate to large values of vorticity that are typically found only in regions of strong rotation (as opposed to shear). This approach highlights the presence of vortices such as those involved with the formation and maintenance of the tornado. In Figure 2 (a) the tornado vortex is undergoing the process of vortex breakdown in which the vortex near ground level "breaks down" aloft into twin intertwined cyclonic vortices. In Figure 2 (b) and (c), the presence of strong, "hairpin" vortices containing both vertical and horizontal vorticity are indicated. The presence of intense horizontal rotation is also briefly visible in the rendering of the cloud field (see [5]), which indicates a horizontally oriented cloud roll encircling and ascending around the tornado, a phenomenon that matches field observations of some tornadoes.

Animations of vorticity reveal the presence of dozens of shallow, curved vortices that form along the forward flank gust front (FFGF) and feed into the tornado vortex. The FFGF is visible in the surface buoyancy field in Figure 2 as the sharp boundary between the cool (blue) air produced by the storm and the ambient air ahead of the storm, extending to the right of the tornado. Animations of the vorticity field tell a compelling story where the tornado grows as it assimilates the many cyclonic vortices that form in this region. It is also interesting to note the presence of anticyclonic (blue) vortices that form in this same region and the different behavior observed when these vortices (as opposed to the cyclonic vortices) encounter the tornado. The anticyclonic vortices, rather than being absorbed into the tornado's circulation, tend to maintain their integrity, but are swept around the outer circulation of the tornado, often completing one or more full cycle of rotation around the tornado while concurrently being tilted and lifted upwards by the rising air that is found surrounding the tornado.

V. Summary

HPC resources were utilized to run and visualize a breakthrough simulation involving a long-track EF5 tornado embedded within a supercell. Code was developed to utilize buffered HDF5 output in the CM1 model in order to achieve satisfactory throughput when doing I/O. A plugin was developed in order to interface VisIt to the CM1 output format. Volume rendering was utilized to create images of the cloud and rain field that show features observed in field studies of tornadic thunderstorms. Volume rendering of vorticity reveals a complex, fascinating turbulent flow regime in the vicinity of the tornado as it forms, grows, and churns away for over 90 minutes.

VI. Acknowledgments

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications.

References

[1] Howard B. Bluestein. *Severe Convective Storms And Tornadoes: Observations and Dynamics*. Springer, 2013.

[2] George H. Bryan and J. Michael Fritsch. A Benchmark Simulation for Moist Nonhydrostatic Numerical Models. *Monthly Weather Review*, 130(12):2917–2928, 2002.

[3] Norman, OK National Weather Service Office. The Calumet-El Reno-Piedmont-Guthrie Tornado of May 24, 2011. http://www.srh.noaa.gov/oun/?n=events-20110524-tornado-b2, 2011.

[4] Leigh Orf. An I/O strategy for CM1 on Blue Waters. Technical report, Central Michigan University, November 2013. http://orf5.com/bw/cm1tools-November2013.pdf.

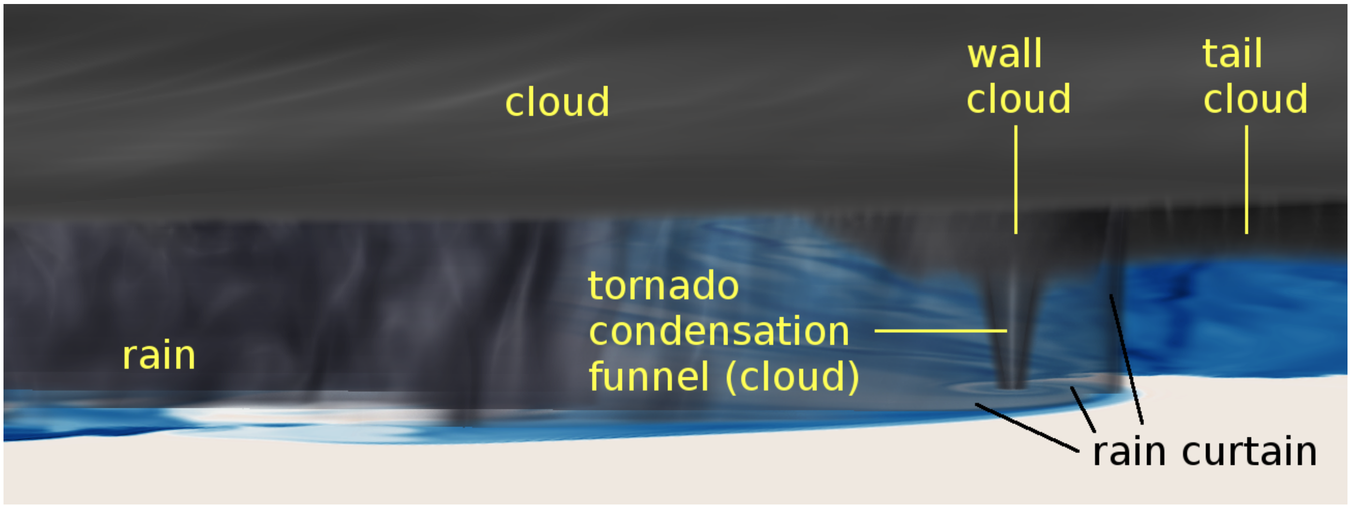

Figure 1: Annotated volume rendering of cloud and rain mixing ratio, and surface buoyancy represented as a 2D pseudocolored field.

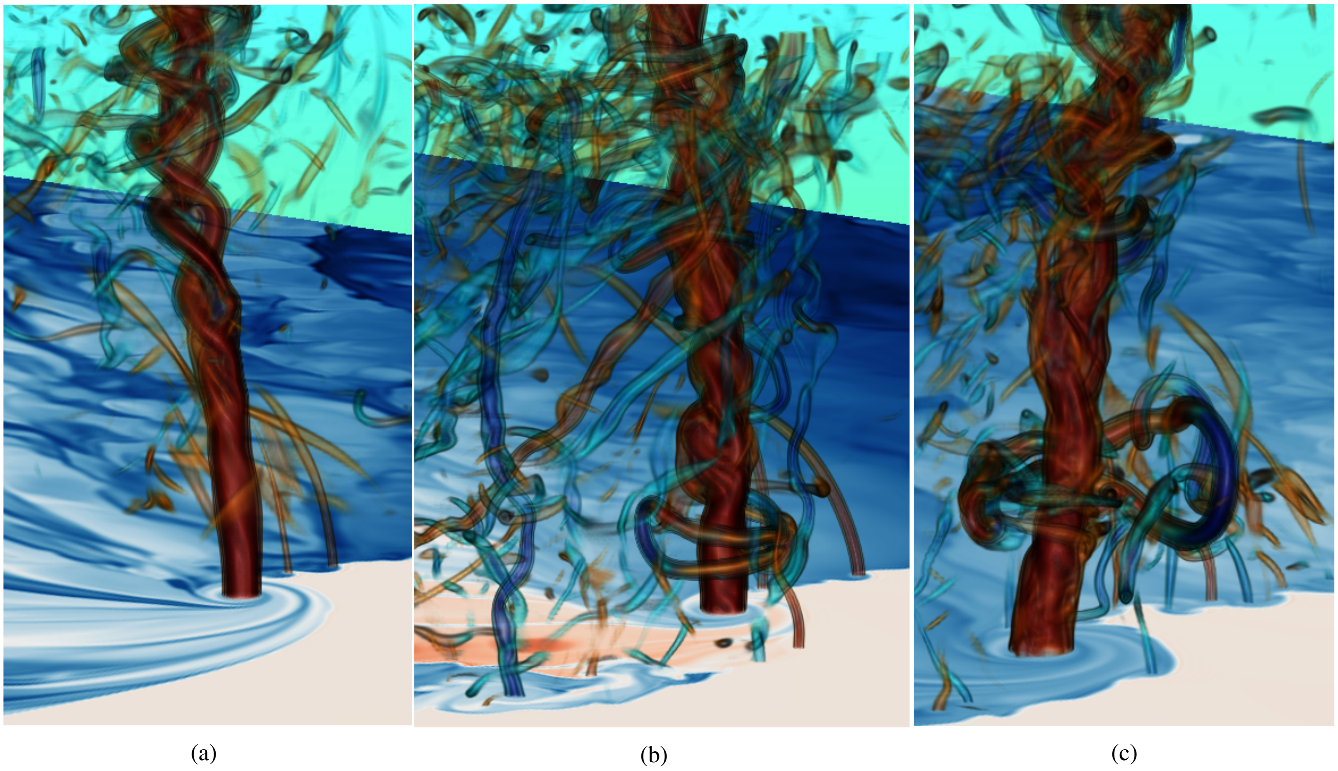

Figure 2: Volume renderings of vorticity magnitude, colored by the vertical component of vorticity, at three different times. As in Figure 1, surface buoyancy is also shown as a 2D pseudocolor surface. Red indicates vorticity with a positive vertical component (such the a cyclonically rotating tornado in (a)–(c)), blue indicates vorticity with a negative component (such as the anticyclonic vertically oriented vortex in (b)). This approach allows for the visibility of purely horizontal vortices while distinguishing cyclonic from anticyclonic vertically oriented vortices.

[5] Leigh G. Orf, R. Wilhelmson, and L. J. Wicker. A Numerical Simulation of a Long-Track EF5 Tornado Embedded Within a Supercell. https://ams.confex.com/ams/94Annual/webprogram/Paper242579.html, 2014. 94th American Meteorological Society Annual Meeting.

[6] VisIt users portal. http://visitusers.org/, 2011.