Performance Results for the Weather Research and Forecast (WRF) Model on AHPCRC HPC Systems

Tony Meys, *Army High Performance Computing Research Center / Network Computing Services, Inc.*

ABSTRACT: *The Army High Performance Computing Research Center (AHPCRC) supports research and development for many applications, including weather models. In recent months, the Weather Research and Forecast (WRF) Model has been implemented and tested on the AHPCRC Cray X1 and on an Atipa Opteron Linux cluster. This paper will discuss recent benchmarking results for configurations of WRF that have used mid-sized and large domains. In addition to performance analysis, this paper will also relate our recent experiences with post-processing and visualizing large, high-resolution weather simulations.*

KEYWORDS: Cray X1E, AMD, WRF, NWP, cluster

1. Introduction

High-resolution forecasts of atmospheric conditions are of great interest to the U.S. Army. Weather conditions in the boundary layer of the atmosphere, that part of the air/land/water system that includes the surface and extends above ground for up to one kilometer, are of particular importance since this is where ground and near surface operations are conducted.

Understanding and forecasting weather parameters in the lower region of the atmosphere poses unique, still unsolved challenges. Complex interactions between terrain and atmospheric flow must be considered. The sourcing, transport, and deposition of moisture, particulates, and chemicals must be included. Atmospheric phenomena that bend and refract electromagnetic and acoustic waves need to be accounted for as well.

In the field of numerical weather prediction (NWP), simulating more detailed surface processes increases the computational requirements of problems. This increase in computations can roughly be categorized as arising from two sources. The first comes from the increased complexity of the problem. As already alluded to, models that consider surface details no longer can hide the reality which gets assumed away or handled by a parameterization in a model with coarsely spaced grid points. The second increase in the number of computations results from finer grid spacing requirements over the area of interest (AOI). For example, a grid mesh with dimensions of 100x100x35 with a 10 km spacing between points in the horizontal may produce a reasonably useful depiction of general temperature, wind, and precipitation patters over an AOI using a very modest computing resource by today's standards. Decrease that spacing down to 1 km and instead use 50 vertical levels and the representation of the atmosphere above the AOI now requires 143 times more memory. With the requirement of a smaller time step, on order of 1430 times more compute cycles will now be needed.

The Army High Performance Computing Research Center (AHPCRC) provides Army and university researchers with the tools to investigate the fringe of high resolution numerical weather prediction. Two of the most recent additions to the AHPCRC computational arsenal are the Cray X1E and an Atipa Linux Opteron Cluster. This paper will compare and contrast these two parallel computing platforms when used to forecast the weather using the Weather Research and Forecasting (WRF) Model.

2. The Cray X1E

The Cray X1E is based on a parallel architecture in which multiple nodes each contain 16 vector capable processors. Each of these processors is referred to as a single streaming processor (SSP) and is the smallest individual processor element that an application can be programmed for on the machine. For many applications a second processor configuration is used. Using special hardware and software, 4 SSPs on an X1E node can be joined together to create what appears to the user as a single, more capable, processor. This processing unit is referred to as a multi-streaming processor (MSP). As its name implies, an MSP provides several streams of instruction execution at a very low level. In most cases, the compiling system on the Cray X1E is capable of identifying or transforming work to inner loops where independent chunks of calculations can be streamed through the multiple pipes of an MSP. Each node of an X1E has 16 SSPs, hence, 4 MSPs.

Figure 1. The AHPCRC Cray X1E.

The fundamental building block of an X1E above the processor level is the Cray X1E Compute Module. In the original Cray X1 design, a node is synonymous with a Compute Module. On the X1E, 8 MSPs share a common module memory. From this hardware two logical nodes are implemented each using 4 MSPs and half of the module memory. The AHPCRC Cray X1E has Compute Modules with 16 Gbytes each, so each node can be used as an 8-way MSP (or 32-way SSP) symmetric multi-processing (SMP) computer with 8 Gbytes of memory.
The AHPCRC Cray X1E contains two cabinets of Compute Modules. Each cabinet can hold up to 16 modules. The AHPCRC Cray X1E, being fully populated, contains 256 MSPs (1024 SSPs) and has 512 Gbtyes of total memory. Communication between modules uses a proprietary interconnect that uses multiple 2D torus topologies. The total on-line disk storage of the system is 50 Tbytes. Components that control I/O for the system are housed in a third, attached cabinet.

The operating system for the X1E is Cray's proprietary version of UNIX, UNICOS/mp. In the AHPCRC configuration, one node of the machine is dedicated to the most familiar OS functions: login sessions, user commands, and other basic system activities that users expect in a shell environment. All other nodes of the system have full UNIX functionality as well, but many of the system requests are in fact handled by that part of the OS that resides on the OS node. UNICOS/mp running across the nodes is coordinated by the system in such a way that a single OS image is presented to the user.

Interactive and batch (PBS) jobs are supported. Cray provides its own proprietary compilers and linkers for code development.

3. The Linux Opteron Cluster

In contrast to the Cray X1E, which has a proprietary hardware and system software design with a variety of HPC-specific design characteristics, the AHPCRC Linux Opteron Cluster is configured using more generally available hardware components and is controlled with a Linux-based operating system. For purposes of this discussion, the Linux Opteron Cluster can be viewed as representative of a growing class of servers that use Intel or AMD processors in groups of 1 to 8 on a node or "blade", connected by at least a 1000baseT network, and often an even faster network specifically designed for clustering.

The AHPCRC Linux Opteron Cluster is comprised of three cabinets that contain a total of 74 compute nodes. Each node supports two 2.2 Ghz AMD Opteron processors for a total of 148 processors. Large memory nodes on the system have 16 Gbytes each. Small memory nodes have 8 Gbytes. There are 54 small memory and 16 large memory nodes. Communication between nodes is performed using a high-speed, low-latency Myrinet network. The available disk storage on the AHPCRC Linux Opteron Cluster is 4.5 Tbytes.

Figure 2. The AHPCRC Linux Opteron Cluster.

The OS for the Linux Opteron Cluster is based on a modified version of RedHat Linux. As such, the user environment for commands and single threaded jobs is virtually identical to running on a Linux workstation. Applications are compiled and linked using Portland Group, Inc. (PGI) tools. In true cluster style, each node on the machine runs its own, complete copy of Linux. For applications targeted to run on multiple nodes, MPICH is used. By linking an MPI application with MPICH libraries built with Myrinet support for the Linux Opteron Cluster, parallel applications can be run.

The Linux Opteron Cluster at AHPCRC is configured such that all users log into a "head" node. To use other nodes on the system, the user must submit a job to a PBS queue.

4. The Weather Research and Forecast (WRF) Model

The Weather Research and Forecast (WRF) Model has been under development for the past several years. Viewed as a follow-on and replacement to its predecessor the Mesoscale Model Version 5 (MM5), WRF is expected to address the needs of both research and operational meteorologists interested in mesoscale forecasting.

Mesoscale modeling typically is done over a limited geographic area. Examples of the size of a typical mesoscale domain would be the Midwest or the Great Lakes regions of the United States. Given the limited extent of a typical WRF domain, boundary information from another model run is required. This boundary information can be thought of as weather conditions that feed into the outer edges of the domain. For all of the model runs discussed in this paper, output results from the National Center for Environmental Prediction (NCEP) North American Meso (NAM) model were used. The NAM was formerly known as the Eta-coordinate model.

Models like WRF are used to study small scale weather features that include frontal thunderstorms (squall lines), mesoscale convective complexes (MCC), coastal land-sea forced winds, and topographically induced weather conditions such as those created by urbanization or the cover of vegetation. To resolve weather events at this scale, WRF is typically run with a gridpoint spacing of from 15 km to 1 km.

WRF is actually comprised of several components necessary to implement a process to run a numerical weather forecast. The WRF Standard Initialization (WRFSI) component is used to ingest gridded analysis, boundary condition, and terrestrial data to create initialization information for the forecast model that has been mapped to the number of levels and AOI for the planned simulation. This data is then given to the Real Data Initialization process that prepares a zero hour initialization file and a periodic (in time) boundary conditions file for input to the WRF Model. See Table A for a listing of many of the features of the solver used in the WRF Model. See Table B for examples of the physics options in the code.

Table A. WRF Model Solver
fully compressible nonhydrostatic equations with hydrostatic option
complete coriolis and curvature terms
two-way nesting with multiple nests and nest levels
one-way nesting
mass-based terrain following coordinate
vertical grid-spacing can vary with height
map-scale factors for conformal projections: polar stereographic, Lambert-conformal, Mercator
Arakawa C-grid staggering
Runge-Kutta 2nd and 3rd order timestep options

Table A: WRF Model Solver (continued)
scalar-conserving flux form for prognostic variables
2nd to 6th order advection options (horizontal and vertical)
time-split small step for acoustic and gravity-wave modes
small step horizontally explicit, vertically implicit
divergence damping option and vertical time off-centering
external-mode filtering option
lateral boundary conditions
idealized cases: periodic, symmetric, and open radiative
real cases: specified with relaxation zone
upper boundary absorbing layer option (diffusion)

(Source: MM5 WRF Users Page, Modeling System Overview, WRF Model, Version 2.)

Table B. WRF Model Physics
microphysics (Kessler / WRF Single Moment (WSM) 3, 5 and 6 class / Lin et al./ Eta Ferrier)
cumulus parameterization (new Kain-Fritsch with shallow convection / Betts-Miller-Janjic, Grell-Devenyi ensemble)
planetary boundary layer (Yonsei University (S. Korea) / Mellor-Yamada-Janjic)
surface layer (similarity theory MM5 / Eta)
slab soil model (5-layer thermal diffusion / Noah land-surface model / RUC LSM)
longwave radiation (RRTM) shortwave radiation (simple MM5 scheme / Goddard)
sub-grid turbulence (constant K diffusion / Smagorinsky / predicted TKE)
land-use categories determine surface properties

(Source: MM5 WRF Users Page, Modeling System Overview, WRF Model, Version 2.)

5. Basic Research Problem Benchmark

With the addition of the Linux Opteron Cluster early in January 2005, and an upgrade of the AHPCRC Cray X1 to an X1E in March 2005, several WRF experiments were assembled to test and evaluate new hardware. One of these tests covers the continental United States with a grid spacing of 5 km. This case, named "conus5", has a horizontal dimension of 680x1000 and has 31 levels in the vertical. The forward timestep of conus5 is 30 seconds. See Figure 3 for an illustration of this domain. For this benchmark, output was written from the model every three forecast hours. The forecast length was set for 12 hours. This experiment is representative of a mid-size research NWP model and is similar in size and compute intensity to many regional operational forecast configurations.

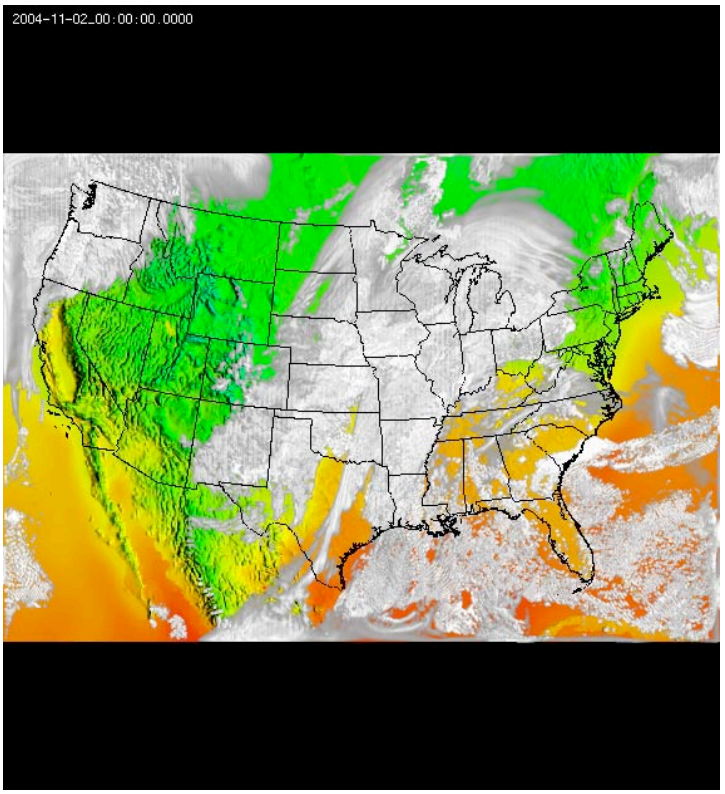

Figure 3. The conus5 domain.

The Cray Program Environment Version 5.3.0.2 was used to build the WRF Model for this test. The results from a series of model runs with differing numbers of MSPs are shown in Table C.

MSPs (SSPs)	X1 (sec)	X1 GF/sec
004 (16)	14454	12
008 (32)	7658	23
016 (64)	4144	42
032 (128)	2466	70

Table C. Cray X1E conus5 results for a 12 hour simulation.

This same case was also run on the Linux Opteron Cluster after building executables using Portland Group (Version 5.2-4) compilers. The results of a series of model runs with differing numbers of processors are shown in Table D.

Opteron procs	Opteron (sec)	Opteron GF/sec
016	17230	10
032	9961	17
064	5109	34
128	2984	58

Table D. Linux Opteron Cluster conus5 results for a 12 hour simulation.

In terms of performance, the Opteron at 2.2 Ghz performs fairly well when compared to a Cray X1 SSP. Scaling for this problem is similar. Generally, the test performed about 20% slower on the Opteron cluster when the unit of comparison used is Opteron processor to SSP. Notably, at 128 Opteron processors almost all (86%) of the cluster is needed to run this one problem. In comparison, only 1/8 of the Cray X1E is used to run the same WRF simulation. The X1 is clearly the more capable machine, but the Linux Opteron Cluster shows that, if grown large enough, it could still run similarly sized problems.

6. Large Benchmark Problem

The next case, named "conus1000x1600", has a horizontal dimension of 1000x1600 and has 31 levels in the vertical. The forward timestep of conus1000x1600 is 15 seconds. Except for increased resolution, the domain was the same as the conus5 test. For this benchmark, output was written from the model only at the end of the simulation. The forecast length was set for 3 hours. This experiment is representative of a large sized research NWP model.

The Cray Program Environment Version 5.3.0.2 was used to build the WRF Model for this test. The result from a series of model runs with differing numbers of MSPs is shown in Table E.

MSPs (SSPs)	X1E (sec)	X1E GF/sec
032 (128)	2316	89
060 (240)	1500	138
128 (512)	905	230
192 (768)	769	270

Table E. Cray X1E conus1000x1600x31 results.

This same case was also run on the Linux Opteron Cluster after building executables using Portland Group (Version 5.2-4) compilers. The result from that model is shown in Table F.

Opteron procs	Opteron (sec)	Opteron GF/sec
128	3300	62

Table F. Opteron conus1000x1600x31 results.

7. A Performance Case Study From a Recent Ensemble Experiment

Although benchmarks provide a very useful metric when evaluating systems, the true test of functionality is when computers are applied to an actual research problem. One such problem that has been performed on the Cray X1E involves an ensemble of forecasts centered on an Army area of interest in the southwestern United States. White Sands Missle Range (WSMR) researcher Robert Dumais is the principal investigator for the project. Working with the author, both the Linux Opteron Cluster and the X1E were used to perform pre-processing of data and execution of the forecast ensemble. Each member of the ensemble is run using a grid that is 512x512 in the horizontal with 45 levels in the vertical. The horizontal spacing between grid points is 2 km. Figure 4 shows the full extent of the domain. Figure 5 zooms into a portion of the domain to reveal some of the detail WRF can capture at a 2 km resolution.

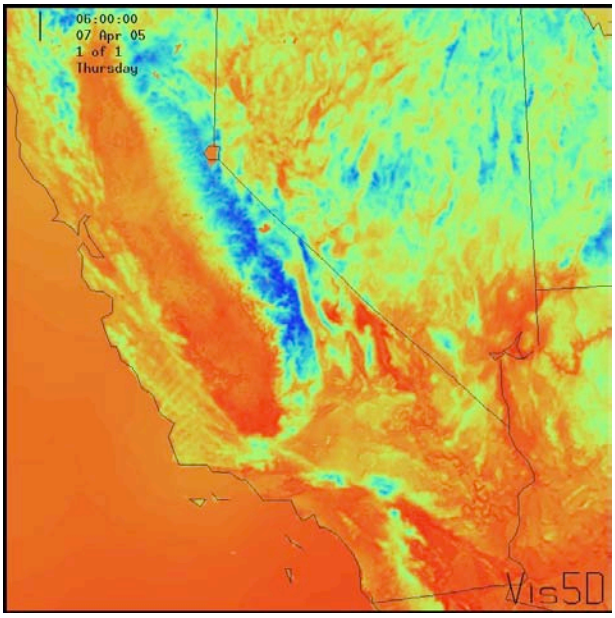

Figure 4. The domain used in the ensemble.

The ensemble is comprised of four members. Each member requires running a complete forecast cycle for the area of interest. The members of the ensemble differ from each other by the physics options used. This paper will not examine the initial scientific results of the ensemble project, but will instead note some of the computational characteristics of the ensemble to give the reader an idea of the challenges involved with a numerical weather prediction ensemble experiment.

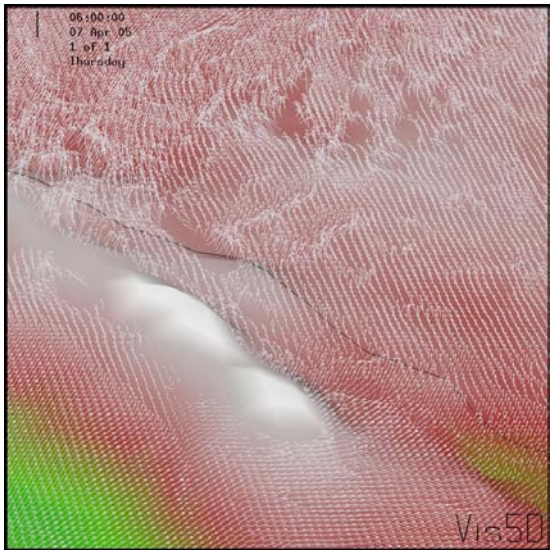

Figure 5. An example of a wind features forecast by one member of the ensemble.

The first step in the configuration of any WRF Model is to define the horizontal and vertical boundaries of the experiment. A graphical user interface (GUI) that comes with WRFSI provides an efficient and straightforward tool to accomplish this. Although many parts of the WRFSI are available on the Cray X1E, the GUI interface is not. The WRFSI GUI design lends itself well to a Linux workstation environment, so it made sense to use several nodes of the cluster system when creating the relatively large terrain files that define the domain. Domain definition is a one time process. Once the domain is defined and generated for each member of the ensemble, it does not need to be done again each time the ensemble is run.

Combining initialization data with the domain definition files is a process that needs to be done individually for each member of an ensemble and repeated anew each time a new ensemble forecast is made. The Linux Opteron Cluster is capable of doing this part of the WRFSI, but since the size of files even just to start the WRF Model can be large, it makes more sense to run this stage of the analysis on the X1E. Note that since the work at this stage involves fairly scalar code that is not very scalable, both the Linux Opteron Cluster and the X1E can do the analysis in a similar window of time. The selection of running this part of WRFSI on the X1E was done largely out of convenience and a wish to avoid file transfers between machines every time the ensemble is run.

As previously noted, when comparing SSPs to Opteron processors, the X1E has a modest advantage. This is only part of the story, however. Although these are research model runs, it is desirable to set the experiment up in a way that allows the ensemble to be run in a time window of about four hours. The idea here was to run experiments that could also be done in near real-time if so desired. The AHPCRC Cray X1E can run the entire ensemble in the time window using about one-half of the available capacity of the machine. The Linux Opteron Cluster would have to be grown by a factor of four to run the same problem and would need to be fully dedicated to running just the ensemble workload.

8. Post-processing

Running the WRF Model is only part of the challenge for researchers working with high-resolution problems. As problem size increases, the input and output files used by WRF become more difficult to manage. Some examples of WRF resolution and required disk space to save hourly output for a 48 hour forecast are shown in Table G. Note that once the problem size reaches 1000x1600x31, output from a single run is nearly a quarter terabyte. Moving this type of file off the machine for analysis is possible, but cumbersome.

Dimensions	Gbytes/hr	Gbytes/48hr
100x80x31	0.013	0.6
512x512x45	0.7	35.4
680x1000x31	1.1	53.2
1000x1600x31	5.0	240.0

Table G. Problem size compared with WRF output data. Hourly output and total output after 48 simulation hours is shown above.

Post-processing this data continues to be an on-going challenge as problem size increases. HPC configurations that provide some nodes with

very large memory, large amounts of disk space, and Linux/Unix environments that can run existing post-processing and visualization tools are best suited for this task, in the author's opinion.

Part of the answer to the large file post-processing problem is to filter output data down to only those fields of interest. Another data management solution is to do at least part of the post-processing of results on the HPC machine. Tools such as WRF2Vis5D and WRF2GrADS have been used in the AHPCRC environment to select out fields for analysis and then create smaller files that can be transferred to a workstation and viewed with Vis5D and GrADS.

NCS has also developed a client/server-based visualization tool that can read WRF files on the X1E and then create time series animations. The tool, named Vivendi, includes a server that runs on the Cray X1E and a client control and viewer application that runs on a workstation. The client is used to send commands to the server to direct rendering. Communication is done via a secure socket connection. Since images are rendered on the server side, the client itself is a fairly lightweight application. Sending only image data to the client also greatly reduces bandwidth requirements to view model output.

9. Conclusions

Both the Cray X1E and the Linux Opteron Cluter have been demonstrated as useful tools to pre-process, run, and post-process the WRF Model. The AHPCRC Linux Opteron Cluster at NCS shows its greatest strength when used for preparing and analyzing data associated with the WRF model. It is capable of doing modest WRF simulations, but the Cray X1E at the AHPCRC/NCS has better per processor performance and much greater capability to do more demanding high resolution and ensemble simulations.

Both systems demonstrated good scaling properties for a modest number of processors when running WRF. A larger Opteron-type cluster would be needed to make comparisons beyond that which has been noted in this paper.

It should be noted that the capability which is just becoming available in an Opteron cluster has been available and in use using the X1 for over two years. As more users have been added to the Linux Opteron Cluster it has also become apparent that the OS support on the X1 is notably more mature for handling multiple users.

Looking towards the future, the still unresolved and most daunting issue for weather simulations using HPC is, in the author's opinion, dealing with the large amounts of data generated by modern mesoscale (and finer scale) atmospheric science problems. Selection of architecture and machine types will, of course, continue to be a topic of interest so long as a diversity of HPC platforms exists. That discussion may, however, be of less central importance in the future as common problems shared by all types of HPC machines set limits on atmospheric research. Dealing with gigabytes, and soon terabytes of data per experiment seems to be that new, common limitation for all systems.

Acknowledgments

The research reported in this document/ presentation was performed in connection with contract/instrument DAAD19-03-D-0001 with the U.S. Army Research Laboratory. The views and conclusions contained in this document/presentation are those of the authors and should not be interpreted as presenting the official policies or positions, either expressed or implied, of the U.S. Army Research Laboratory or the U.S. Government unless so designated by other authorized documents. Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

About the Author

Tony Meys is a User Support Specialist, Atmospheric Science, with Network Computing Services, Inc. (NCS). Mr. Meys has worked with atmospheric and other environmental models on Cray Research and other HPC machines since 1989. He can be contacted at Network Computing Services, Inc., 1200 Washington Avenue South, Minneapolis, MN 55415 USA. E-mail contact is tmeys@ahpcrc.org.