

# Performance Analysis of Parallel Visualization Applications and Scientific Applications on an Optical Grid\*

Xingfu Wu and Valerie Taylor

*Department of Computer Science, Texas A&M University, College Station, TX 77843*

*E-mail: {wuxf, taylor}@cs.tamu.edu*

## Abstract

One major challenge for grid environments is how to efficiently utilize geographically distributed resources given large communication latency introduced by wide area networks interconnecting different sites. In this paper, we use optical networks to connect four clusters from three different sites: Texas A&M University, University of Illinois at Chicago, and University of California at San Diego to form an optical Grid as a testbed, and execute parallel scientific applications and visualization applications to analyze the performance of these applications on the optical Grid. The ideal applications for the optical Grid are embarrassingly parallel.

## 1. Introduction

Distributed systems, such as the Distributed TeraGrid Facility [TRG], the Open Science Grid [OSG], and the European Data Grid [EDG] are available and provide vast compute and data resources. Large scale parallel scientific applications and visualization applications often require computational and/or data grids to obtain the larger compute and/or storage resources necessary for execution. For these applications, the major challenge for grid environments is how to efficiently utilize the geographically distributed resources given the large communication latency introduced by wide area networks interconnecting the different sites.

The OptIPuter, so named for its use of **Optical** networking, **Internet Protocol**, **computer** storage, processing and visualization technologies, is an infrastructure that tightly couples computational resources over parallel optical networks using the IP communication mechanism [BS06, SC03, OPT]. The OptIPuter project provides us a distributed cyber-infrastructure for grid computing across the geographically distributed resources. In this paper, we use optical networks to connect four clusters from three different sites: Texas A&M University, University of Illinois at Chicago, and University of California at San Diego to form an optical Grid as a testbed, and run parallel scientific applications and visualization applications to analyze the performance of these applications on the optical Grid.

## 2. An Optical Grid and Its Performance

---

\* This work is supported by US NSF (ITR grant ANI-0225642) OptIPuter Project.

In this section, we describe the configurations of our testbed, an optical grid, and discuss its performance.

## 2.1 OptIPuter Project, National LambdaRail, and LEARN

The US NSF-funded OptIPuter project [OPT] exploits a new world in which the central architectural element is optical networking, not computers - creating "supernetworks". The goal of this new architecture is to enable scientists who are generating terabytes and petabytes of data to interactively visualize, analyze, and correlate their data from multiple storage sites connected to optical networks. Lead institutions in OptIPuter project are University of California at San Diego (UCSD) and University of Illinois at Chicago (UIC). Texas A&M University (TAMU) is one of funded academic partners in the project.



**Figure 1. US National LambdaRail and OptIPuter CAVEWave [BS06]**

The OptIPuter project has its own optical network called CAVEWave, a switched infrastructure, which includes all sites covered by a star shown in Figure 1. The OptIPuter CAVEWave helped launch US National LambdaRail [NLR].

National LambdaRail (NLR) [NLR] is advancing the research, clinical, and educational goals of members and other institutions by establishing and maintaining a unique nationwide network infrastructure that is owned and controlled by the U.S. research community. Ownership of the underlying optical infrastructure ensures the research community unprecedented control and flexibility in meeting the requirements of the most advanced network applications and providing the resources demanded by cutting-edge network research.

The Lonestar Education And Research Network (LEARN) [LEA] is a cooperative effort of 33 institutions of higher education in Texas, USA to provide high-speed connectivity between their institutions as well as to research networks across the US in support of higher education's research, teaching, health care, and public service missions. Texas A&M University (College Station) is connected to National LambdaRail (NLR) Houston site via LEARN shown in Figure 2.

LEARN Topology for State of Texas

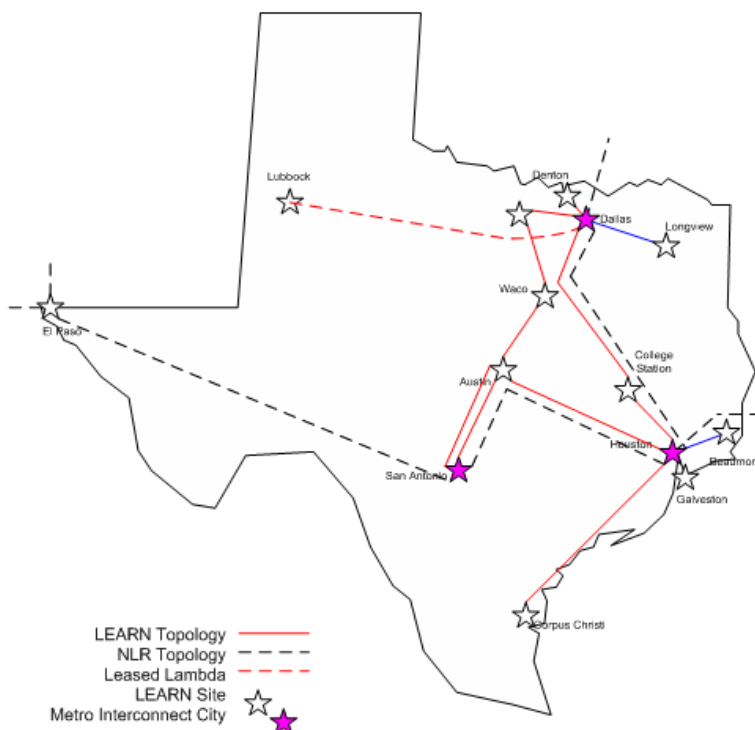


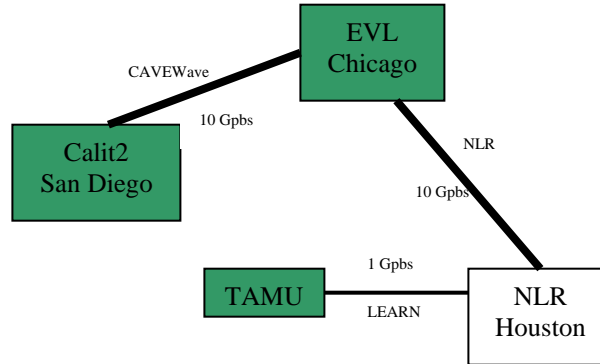
Figure 2. Diagram of LEARN [LEA]

## 2.2 An Optical Grid Testbed across TAMU, UIC EVL and UCSD Calit2

In this section, we discuss an Optical Grid testbed across our OptIPuter partners: TAMU, UIC EVL (Electronic Visualization Laboratory, <http://www.evl.uic.edu>) and UCSD Calit2 (California Institute for Telecommunications and Information Technology, <http://www.calit2.net>). We utilize one Linux graphics cluster Opt (with 5 nodes) at TAMU, two Linux graphics clusters Yorda (with 20 nodes) and Nico (with 10 nodes) at EVL, and one Linux graphics cluster Vellum (with available 8 of 32 nodes) at Calit2 shown in Table 1 to form an optical Grid (or called virtual supercomputer) with optical networks connected among four Linux graphics clusters from TAMU, EVL and Calit2. These Linux graphics clusters have the similar configurations such as dual-core Opteron CPUs, nVidia Quadro FX 3000 GPU, a Gigabit switch, SuSe Linux, and OpenGL, and support scalable tiled display systems [WT05]. These clusters are configured with private optical network IP addresses and gateways like 67.58.\*.\* so that our OptIPuter partners can share the dedicated resources via optical network connections. The gateway for our cluster at TAMU is located in Chicago (EVL), and is directly and dedicatedly connected to the clusters at EVL.

Table 1. Available nodes used for our experiments across TAMU, EVL and Calit2

Cluster @site	Opt @ TAMU	Yorda @EVL	Nico @EVL	Vellum @Calit2
#nodes	5	20	10	8 (32)
CPUs/node	2	2	2	2



**Figure 3. Optical network connections across TAMU, EVL, and Calit2**

The cluster Opt at TAMU has Layer-2 connection to LEARN Houston site via a dedicated single mode fiber (1Gbps) shown in Figure 3. There is a 10Gbps connection between Houston and Chicago (UIC EVL) via NLR. There is a 10Gbps connection between UIC EVL and UCSD Calit2 via CAVEWave. So the communication from TAMU to Calit2 is not direct via NLR. It requires going through LEARN and NLR, then routing a CAVEWave switch located in Chicago to Calit2 via CAVEWave.

**Table 2. Network Latency across TAMU, EVL and Calit2**

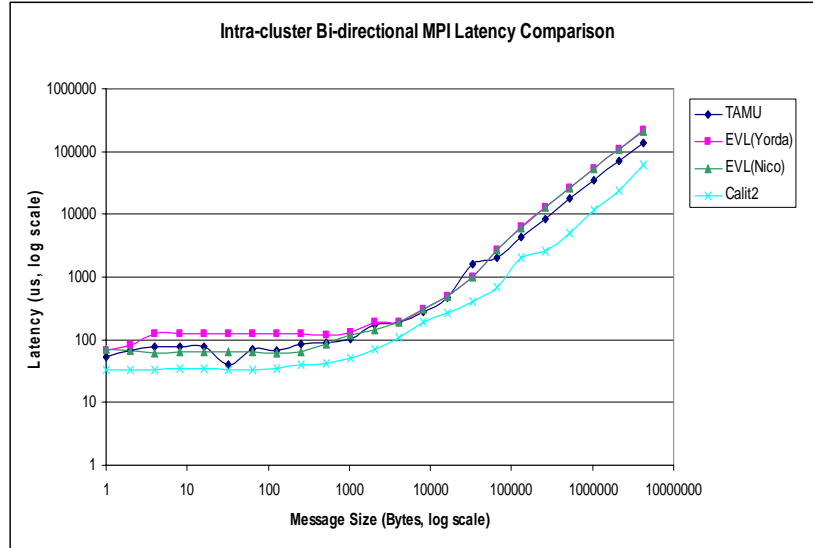
Configuration	Latency (ms, rtt)
TAMU and EVL	27.51
EVL and Calit2	78.33
Calit2 and TAMU	105.44

Table 2 presents the average network latency for round-trip using simple unix command *ping* between TAMU and EVL, between EVL and Calit2, and between Calit2 and TAMU. The latency between TAMU and Calit2 is almost the sum of the latency between TAMU and EVL and the latency between EVL and Calit2.

### 2.3 MPI Execution Environment and Network Performance on the Optical Grid

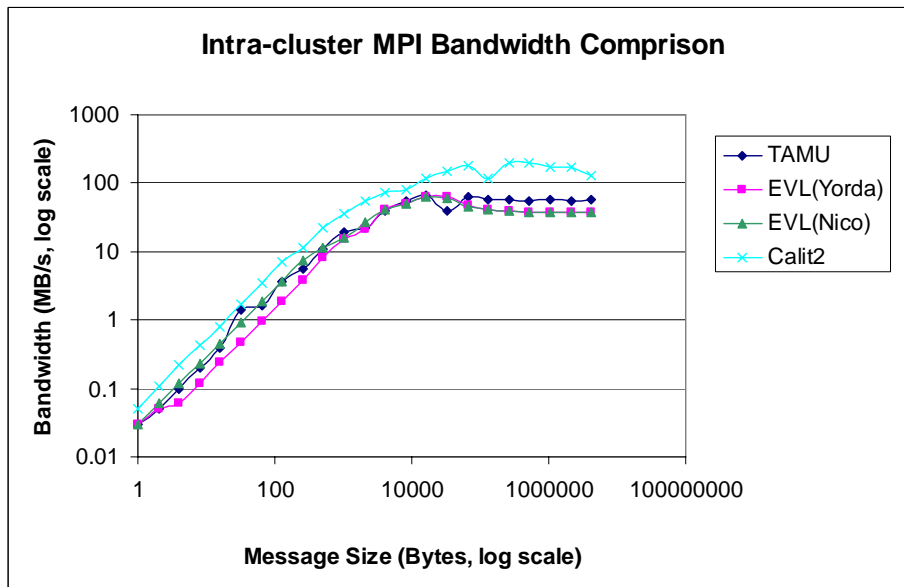
In this section, we discuss how to setup MPI execution environment and measure optical network performance, especially MPI latency and bandwidth between TAMU and EVL, between TAMU and Calit2, and between EVL and Calit2, or on four Linux graphics clusters, Opt at TAMU, Yorda at EVL, Nico at EVL and Vellum at Calit2 because all parallel visualization applications and scientific applications used in our experiments are MPI applications.

Under our account, we install and setup MPICH 1.2.7p1 package [MPI], and setup *ssh* to access any node of the four clusters without password using *ssh-keygen* on each cluster at three different sites: TAMU, EVL and Calit2. After this, a MPI execution environment on the four clusters is ready to run MPI applications. Under our account, the MPI execution environment just looks like a virtual supercomputer which consists of the four geographically distributed clusters shown in Table 1.



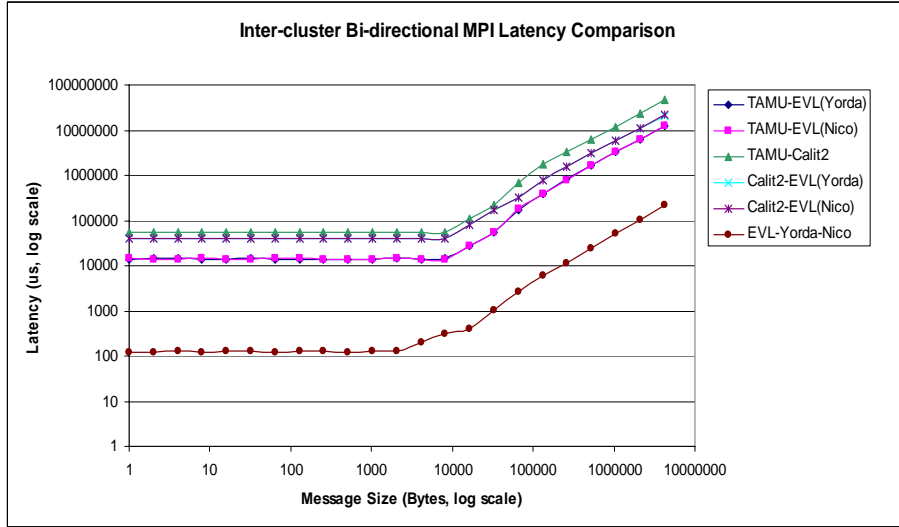
**Figure 4. Intra-cluster Bi-directional MPI Latency for four clusters**

Figure 4 shows the intra-cluster bi-directional MPI latency (unit: microseconds (us)) for the four clusters across TAMU, EVL and Calit2 with the increase of message sizes from 1 byte to 4MB using Intel’s MPI benchmarks [IMB]. The cluster Vellum at Calit2 has the best intra-cluster MPI latency. The cluster Opt at TAMU and Yorda and Nico at EVL have the similar MPI latency with the increase of message sizes.



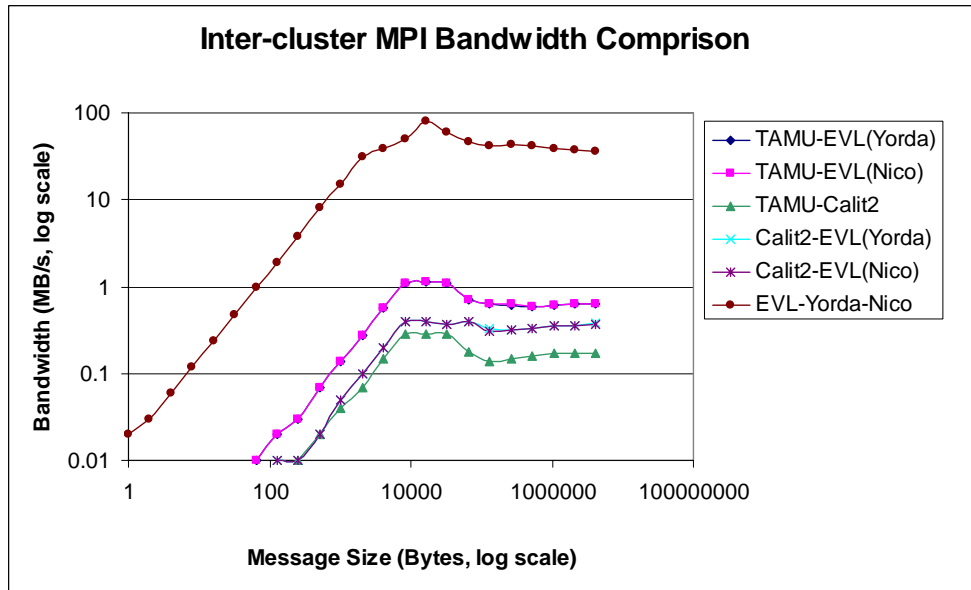
**Figure 5. Intra-cluster bi-directional MPI bandwidth comparison**

Figure 5 presents bi-directional intra-cluster MPI bandwidth comparison across the four clusters from TAMU, EVL, and Calit2. Also, the cluster Vellum at Calit2 has the best intra-cluster MPI bandwidth. The cluster Opt at TAMU and Yorda and Nico at EVL have the similar MPI bandwidth with the increase of message sizes.



**Figure 6. Inter-cluster Bi-directional MPI Latency across TAMU, EVL and Calit2**

Figure 6 shows the inter-cluster bi-directional MPI latency for the four clusters across TAMU, EVL and Calit2, where MPI latency for TAMU-EVL(Yorda) means the MPI latency between the cluster Opt at TAMU and the cluster Yorda at EVL. Comparing to the intra-cluster MPI latencies shown in Figure 4, these MPI latencies are very high (at milliseconds level) except the latency for EVL-Yorda-Nico. The latency for TAMU-Calit2 is almost the sum of the latency for TAMU-EVL(Yorda/Nico) and the latency for Calit2-EVL(Yorda/Nico). This trend is similar to that shown in Table 2.



**Figure 7. Inter-cluster bi-directional MPI bandwidth comparison**

Figure 7 presents bi-directional inter-cluster MPI bandwidth comparison across the four clusters from TAMU, EVL, and Calit2. High inter-cluster latency results in the low inter-cluster bandwidth. Except the EVL-Yorda-Nico, the inter-cluster MPI bandwidth is less than 2MB/s because of MPI's TCP connections for message passing between processors.

### 3. Parallel Visualization Applications

In this section, we present two parallel visualization applications to utilize the Optical Grid. The parallel applications are 3D parallel volume rendering application Vol-A-Tile [WT05, SV04] and 2D high resolution imagery application JuxtaView [KV04].

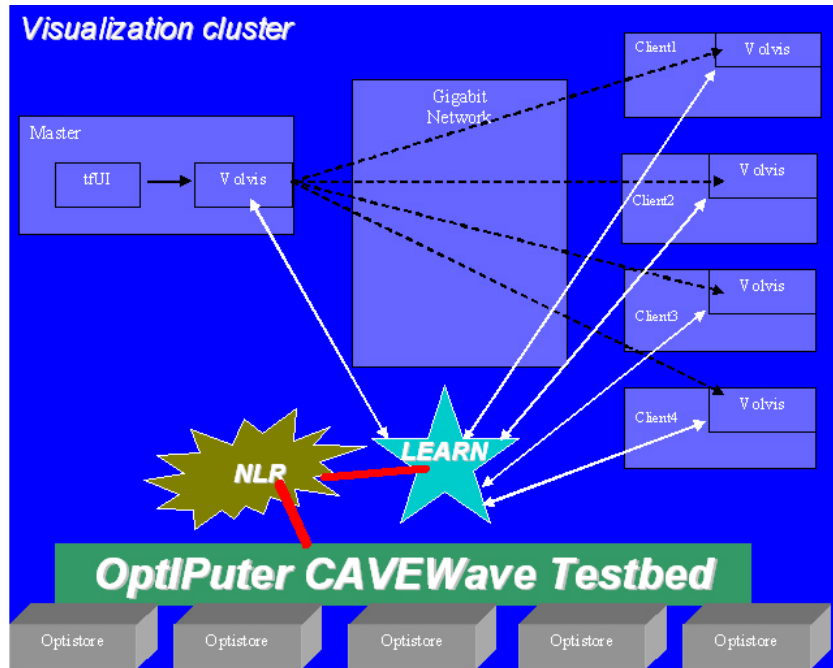
#### 3.1. 3D Visualization Application Vol-A-Tile

Vol-A-Tile is developed by our OptIPuer partner EVL [SV04, WT05]. It is a volume visualization tool for large-scale, time-series scientific datasets rendered on high-resolution scalable displays. These large-scale datasets can be dynamically processed and retrieved from remote data stores over optical networks using OptiStore --- a system that filters raw volumetric data and produces a sequence of visual objects such as iso-surfaces or voxels. Vol-A-Tile consists of three major components:

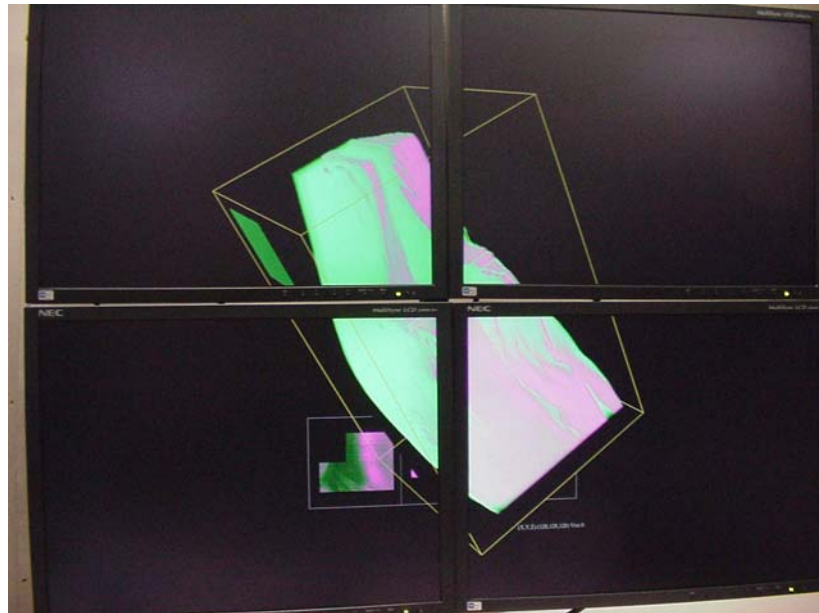
- Data server Optistore,
- Main graphics program Volvis,
- Transfer function Editor (tfUI).

Optistore is the data server, which stores objects as 3D volumes or geometry. The Optistore is designed to assist visualization dataset handling, including data management, processing, representation and transport. Volvis handles the rendering, scalable displaying and user interaction. All the nodes in Vol-A-Tile (master and clients) shown in Figure 8 have a dedicated link to the Optistore to retrieve the datasets. The master handles user interaction and any transfer function updates from the transfer function editor and use MPI to broadcast it to the clients in order to maintain a consistent graph on each client. The broadcast messages are very small, only indicating the operations to be performed on the scene, and are independent of the scene size or the number of tiles. This ensures the scalability of Vol-A-Tile for larger scene sizes on larger tiled displays.

The master node does not calculate any client's scene view. The clients are responsible for processing commands from the master and rendering to their respective view frustums. tfUI is the user interface for transfer function selection. The color and opacity can be selected using the classification widgets. These widgets can be overlaid and then rasterized to a 2D texture which is sent to Volvis. The 2D Histogram for the dataset is retrieved from Volvis and displayed to guide the user in the selection.



**Figure 8. Execution configuration for the viz application Vol-A-Tile**



**Figure 9. Visualization of 3D geologic volumes**

Figure 9 presents a snapshot of the visualization of 3D geologic volumes on 2x2 tiled displays. In [WT05], we did similar experiments on our cluster locally. Here, we transfer needed data (24MB) from EVL to TAMU in less than 10 seconds. Although there is some delay in starting up an initial display because of remote data transfer, the display is still in real-time rate for rotating/zooming the 3D geologic image after the initial display.

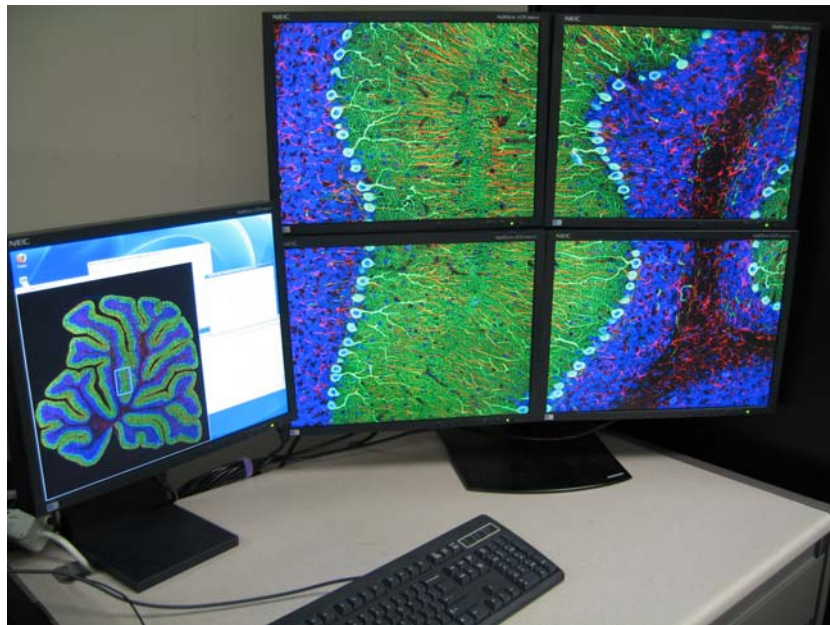
### 3.2. 2D Visualization Application JuxtaView



JuxtaView is a cluster-based application for viewing and interacting with ultra-high-resolution 2D montages such as images from confocal or electron microscopes or satellite and aerial photographs on scalable tiled displays, and is developed by our OptIPuter partner EVL [KV04]. The version of JuxtaView we used is to use MPI for message passing.

The design of JuxtaView is similar to that for Vol-A-Tile using MPI client-server model. The server node (a single standalone screen shown in Figure 10) is in charge of user interaction such as panning/zooming an image and uses MPI to broadcast it to the clients (2x2 tiled display) in order to maintain a consistent image on each client. The broadcast messages are very small, merely indicating the operations to be performed on the image, and are independent of the image size. The clients are responsible for calculating the pixels required for displaying on each tile.

Figure 10 presents the display of a 2D cross-section image of a rat cerebellum which is provided by National Center for Microscopy and Imaging Research (NCMIR) at the University of California, San Diego. Its resolution is 18701x17360 pixels (about 310 Megapixels). The file with RGBA format is about 1238MB. It takes 490 seconds to transfer the file from EVL to TAMU using the Linux command *scp* without password with the average transfer rate 2.53MB/s because the LambdaRAM [KV04] for remote data access is not available for us. So we transfer the needed data from EVL to each node on our cluster at TAMU when needed for display. This causes some delay for starting up an initial display. The display, however, is still in real-time rate for panning/zooming the 2D image of a rat cerebellum after the initial display. When the LambdaRAM is available for us to use, it can help reduce the delay.



**Figure 10. 2D cross-section image of a rat cerebellum**

In summary, in this section, we use two MPI visualization applications, 3D Vol-A-Tile and 2D JuxtaView to utilize the optical Grid. The two visualization applications are designed to be embarrassingly parallel. Huge scientific data (terabytes even petabytes) generated by data-intensive scientific applications such as satellite remote sensing, multi-scale correlated microscopy experiments

and so on, is distributed geographically among data centers. In order to view and analyze visualizations of large distributed heterogeneous data in real-time fashion, it requires ultra-high-speed networks such as optical networks to enable local and distributed groups of researchers to work with one another. Our OptIPuter partners also used the CAVEWave for real-time video streaming for Access Grid, high-definition video-teleconferences, and remote 3D simulation [BS06].

#### 4. Parallel Scientific Applications

In this section, we execute two parallel applications across TAMU, EVL and Calit2, and analyze their performance. The applications are NAS parallel benchmark IS and parallel Mandelbrot Set application (because visualization clusters from EVL and Calit2 do not have any Fortran compiler, we just run MPI programs in C or C++ for current experiments.).

##### 4.1. NAS Parallel Benchmark IS

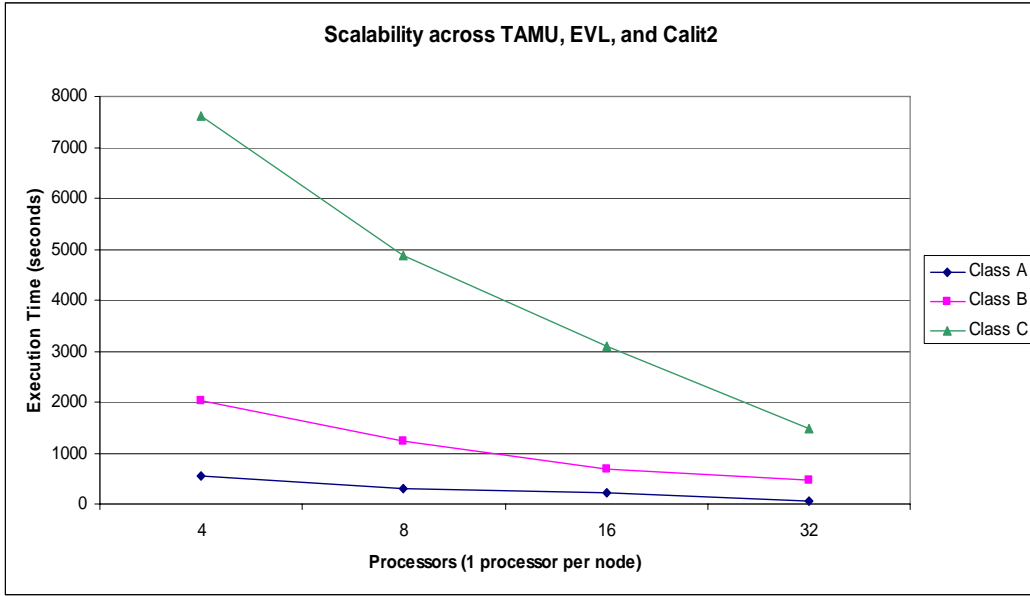
NAS parallel benchmark IS is the only benchmark written in C in NAS parallel benchmarks suite [BB94], and is a large integer sort benchmark which sorts  $N$  keys in parallel. It performs a sorting operation that is important in “particle method” scientific codes. The problem size (the total number of keys) we used is Class A ( $2^{23}$ ), Class B ( $2^{25}$ ), and Class C ( $2^{27}$ ). The benchmark requires that the number of processors is power of 2.

The benchmark IS tests both integer computation speed and communication performance. The communication rate for the benchmark is more than 90% on 4 or more processors on each cluster at TAMU, EVL and Calit2. So the benchmark IS is communication-intensive on the Optical Grid.

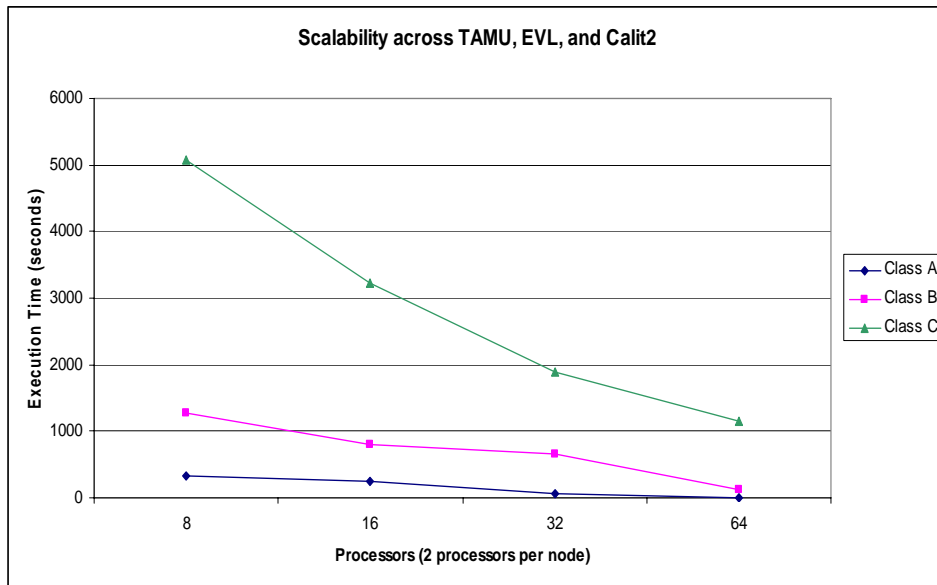
**Table 3. Configurations of CPUs used**

Configuration	Opt @ TAMU	Yorda @ EVL	Nico @ EVL	Vellum @ Calit2
4CPUs/1PPN	1 node	1 node	1 node	1 node
8CPUs/2PPN	1 node	1 node	1 node	1 node
8CPUs/1PPN	2 nodes	2 nodes	2 nodes	2 nodes
16CPUs/2PPN	2 nodes	2 nodes	2 nodes	2 nodes
16CPUs/1PPN	4 nodes	4 nodes	4 nodes	4 nodes
32CPUs/2PPN	4 nodes	4 nodes	4 nodes	4 nodes
32CPUs/1PPN	5 nodes	16 nodes	6 nodes	5 nodes
64CPUs/2PPN	5 nodes	16 nodes	6 nodes	5 nodes

The configurations of CPUs used for our experiments are shown in Table 3. Note that **PPN** stands for **processors per node**. We utilize four clusters from TAMU, EVL and Calit2 on the Optical Grid, and try to use equal number of nodes from each cluster as many as possible to study scalability of the Optical Grid. For example, 16CPUs/1PPN shown in Table 3 means using 4 nodes with 1 processor per node from each cluster; 32CPUs/2PPN means using 4 nodes with 2 processors per node from each cluster.



**Figure 11. Execution Time (seconds) for IS across TAMU, EVL and Calit2 using one processor per node**



**Figure 12. Execution Time (seconds) for IS across TAMU, EVL and Calit2 using two processors per node**

Figures 11 and 12 present the performance of NAS parallel benchmark IS executed across TAMU, EVL and Calit2 with the increase of problem size and the number of processors. Although the large communication latency is introduced by wide area optical networks interconnecting the different sites/clusters, the communication-intensive benchmark IS still scale well with increasing number of processors per cluster, especially for the cases of 32CPUs/1PPN and 64CPUs/2PPN because 16 nodes on the cluster Yorda used result in the lower communication overhead. Of course, the execution time for the benchmark became much larger than that on the same number of processors on one cluster because of much larger inter-cluster communication latency.

**Table 4. Performance for Class A**

Processors	8	16	32
1PPN	301.35	223.44	53.26
2PPN	338.65	249.75	62.38
% difference	12.38	11.77	17.12

**Table 5. Performance for Class B**

Processors	8	16	32
1PPN	1225.14	690.13	456.71
2PPN	1283.91	809.61	663.99
% difference	4.80	17.31	45.39

**Table 6. Performance for Class C**

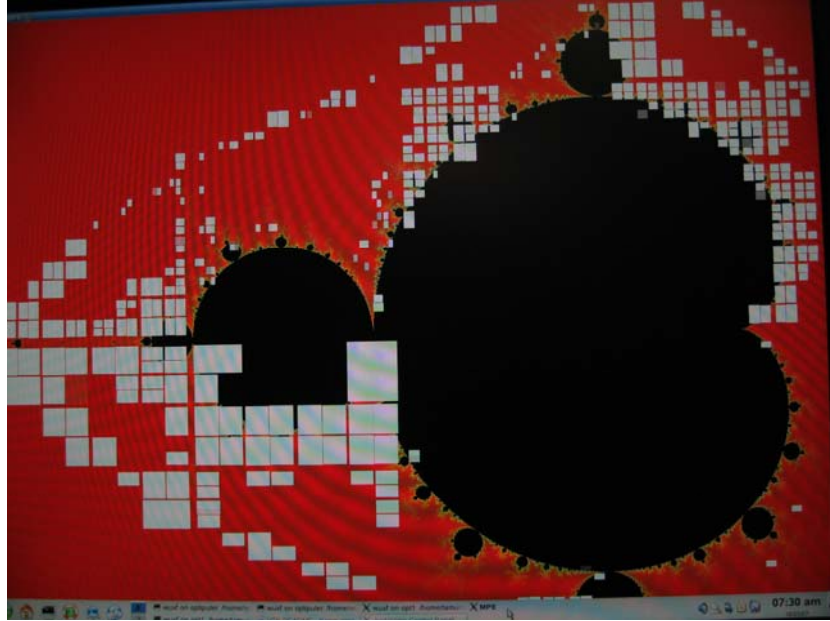
Processors	8	16	32
1PPN	4884.85	3084.7	1475.37
2PPN	5065.83	3229.49	1892.42
% difference	3.70	4.69	28.27

Comparing Figure 11 with Figure 12, the benchmark IS scales better using 1PPN than using 2PPN because of shared caches and memory for the case of using 2PPN. Further, Tables 4, 5 and 6 indicate that the performance for using 1PPN is much better than that for using 2PPN for different problem sizes and number of processors. The performance difference between using 1PPN and 2PPN is up to 17.12% for Class A, up to 45.39% for Class B, and up to 28.27% for Class C. For most cases, the performance difference increases with the increase of number of processors per cluster because of more inter-cluster communication involved.

## 4.2 Parallel Mandelbrot Set

Fractal geometry now plays a central role in the realistic rendering and modeling of natural phenomena. The Mandelbrot set discovered by B.B. Mandelbrot [MB04] is a rather peculiar fractal in that it combines aspects of self-similarity with the properties of infinite change. A Mandelbrot set is generated as follows. Consider a simple equation having the form:  $z = z^2 + C$ , where  $z$  is a variable,  $C$  is a constant and both are complex numbers. For a given value of  $C$ , we choose a particular starting value for  $z$ , and then iterate the equation. It transpires that, depending on the starting value, the sequence of values of  $z$  which are computed will usually either converge to a bounded value or diverge to infinity.

In [WU99], a parallel algorithm for generating a Mandelbrot set was implemented in PVM, and found that the parallel Mandelbrot set is data parallelism with no communication required for exchanging boundary data. This is embarrassingly parallel, and is a good example to utilize the Optical Grid. We find one similar parallel implementation of a Mandelbrot set in the MPICH package [MPI], and run the parallel Mandelbrot set program with the number of iterations of 10,000 on 32 processors as an experiment on the Optical Grid. The distribution of 32 processors is 32 processors/1PPN shown in Table 3.



**Figure 13. Snapshot for generating a Mandelbrot set in parallel using 32 processors across TAMU, EVL and Calit2**



**Figure 14. Snapshot for zooming a small portion of the Mandelbrot set on 32 processors across TAMU, EVL and Calit2**

Figure 13 presents generating a Mandelbrot set in parallel using 32 processors across TAMU, EVL and Calit2. An initial display for the parallel Mandelbrot set has some delay because it takes some time (around 10 seconds) for MPI to build TCP connections among the four clusters across three sites: TAMU, EVL and Calit2. After the initial display, we can zoom any small portion of the Mandelbrot set in real-time rate on 32 processors across TAMU, EVL and Calit2. We also demonstrated the example at the research booth of Gulf Coast Academic Supercomputing Center in SC2007 at Reno, Nevada, USA.

In summary, in this section, we use one communication-intensive application, NAS parallel benchmark IS, and one embarrassingly parallel application, parallel Mandelbrot set, to test our optical Grid. We find that the benchmark IS scaled well on the optical Grid with increasing number of processors and problem sizes. But because of large inter-cluster communication latency for four geographically distributed clusters across TAMU, EVL and Calit2, the ideal applications for the optical Grid are embarrassingly parallel.

## 5. Conclusions

In this paper, we presented an optical Grid testbed, and executed parallel visualization applications and scientific applications across TAMU, EVL and Calit2. The ideal applications for the optical Grid are embarrassingly parallel.

In Section 2, we found that the inter-cluster MPI bandwidth is less than 2MB/s because of MPI's TCP connections for message passing between processors. For future work, we will port MPI to use the new transport protocols and optical signaling, control and management software, which is developed by our OptIPuter partners [BS06] and enables to dynamically manage multiple lambdas for multiple parallel communication paths, to achieve much high communication bandwidth and low latency for large scale scientific applications. We will also use real scientific applications such as the US DOE SciDAC flagship applications Gyrokinetic Toroidal code (GTC) [LE02] and MIMD Lattice Computation (MILC) [MIL] to conduct our experiments on the Optical Grid for further performance analysis.

## Acknowledgements

The authors would like to acknowledge the UIC EVL for the use of visualization clusters Yorda and Nico, and the UCSD Calit2 for the use of the visualization cluster Vellum. We would also like to thank Alan Verlo, Maxine Brown, Jason Leigh, Luc Renambot, and Venkat Vishwanath from UIC EVL, Qian Liu from UCSD Calit2, and Nolan Flowers from TAMU for their help in optical network connections and some visualization applications.

## References

- [BB94] D. Bailey, E. Barszcz, et al., *The NAS Parallel Benchmarks*, Tech. Report RNR-94-007, March 1994. See also <http://www.nas.nasa.gov/Software/NPB/>.
- [BS06] Maxine Brown, Larry Smarr, Tom DeFanti, Jason Leigh, Mark Ellisman, and Philp Papadopoulos, The OptIPuter: A National and Global-Scale Cyberinfrastructure for Enabling LambdaGrid Computing, *TeraGrid'06 conference*, 2006.
- [EDG] The European Data Grid, <http://eu-datagrid.web.cern.ch/eu-datagrid/>.
- [KV04] N. Krishnaprasad, V. Vishwanath, S. Venkataraman, A. Rao, L. Renambot, J. Leigh, A. Johnson, and B. Davis, JuxtView – a Tool for Interactive Visualization of Large Imagery on Scalable Tiled Displays, *IEEE Cluster 2004*.
- [IMB] Intel MPI Benchmarks (Version 2.3), <http://www.intel.com/cd/software/products/asmo-na/eng/cluster/mpi/219848.htm>.
- [LEA] LEARN: Lonestar Education and Research Network, <http://www.tx-learn.net>.

- [LE02] Z. Lin, S. Ethier, T. Hahm, and W. Tang, Size Scaling of Turbulent Transport in Magnetically Confined Plasmas, *Phys. Rev. Lett.* 88, 2002.
- [MB04] Benoit B. Mandelbrot, *Fractals and Chaos: The Mandelbrot Set and Beyond*, Springer, 2004.
- [MIL] MIMD Lattice Computation (MILC) collaboration code,  
<http://www.physics.utah.edu/~detar/milc>.
- [MPI]MPICH 1.2.7p1, <http://www-unix.mcs.anl.gov/mpi/mpich1/>.
- [NLR] National Lambda Rail, <http://www.nlr.net>.
- [OPT] OptIPuter Project, <http://www.optiputer.net>.
- [OSG]Open Science Grid, <http://www.opensciencegrid.org/>
- [SV04] N. Schwarz, S. Venkataraman, L. Renambot, N. Krishnaprasad, V. Vishwanath, J. Leigh, A. Johnson, G. Kent, and A. Nayak, Vol-a-Tile – a Tool for Interactive Exploration of Large Volumetric Data on Scalable Tiled Displays, *IEEE Visualization 2004 (Poster)*.
- [SC03] Larry Smarr, Andrew Chien, Tom DeFanti, Jason Leigh, and Philip Papadopoulos, The OptIPuter, *Communication of the ACM*, Vol. 46, No. 11, Nov. 2003
- [TRG] TeraGrid, <http://www.teragrid.org>
- [WT05] Xingfu Wu, Valerie Taylor, Jason Leigh, and Luc Renambot, Performance Analysis of a 3D Parallel Volume Rendering Application on Scalable Tiled Displays, *International Conference on Computer Graphics, Imaging and Vision (CGIV05)*, Beijing, China, 26-29 July 2005.
- [WU99] Xingfu Wu, *Performance Evaluation, Prediction, and Visualization of Parallel Systems*, Kluwer Academic Publishers, ISBN 0-7923-8462-8, Boston, USA, 1999.