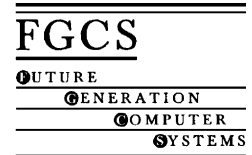




ELSEVIER

Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Future Generation Computer Systems 19 (2003) 849–859



www.elsevier.com/locate/future

Distributed, on-demand, data-intensive and collaborative simulation analysis

Arthurine Breckenridge^{a,*}, Lyndon Pierson^a, Sergiu Sanielevici^b, Joel Welling^b,
Rainer Keller^c, Uwe Woessner^c, Juergen Schulze^c

^a Sandia National Laboratories, Albuquerque, NM, USA

^b Pittsburgh Supercomputing Center, Pittsburgh, PA, USA

^c High Performance Computing Center, Stuttgart, Germany

Abstract

Distributed, on-demand, data-intensive, and collaborative simulation analysis tools are being developed by an international team to solve real problems such as bioinformatics applications. The project consists of three distinct focuses: compute, visualize, and collaborate. Each component utilizes software and hardware that performs across the International Grid. Computers in North America, Asia, and Europe are working on a common simulation programs. The results are visualized in a multi-way 3D visualization collaboration session where additional compute requests can be submitted in real-time. Navigation controls and data replication issues are addressed and solved with a scalable solution. Published by Elsevier B.V.

Keywords: Compute; Visualize; Collaborate; Data-intensive; Simulation analysis; Bioinformatics applications

1. Introduction

The international collaboration being presented in this paper represents a snap shot in time of on-going work. Since our team has proudly participated in all three International Grid conferences, the tremendous advantage of participation can be shown in the project chronology. Initially, the G7 Economic Summit defined a task force to enhance international networking and electronic collaborations. In 1996, the demonstration of our G7 team was presented only locally; no external-to-conference-exhibit network functionality was available. The various international high-speed

links were lacking due to physical gap in the wiring. In 1997, partially due to work on StarTap by Tom DeFanti¹ et al., the first complete transatlantic Asynchronous Transfer Mode (ATM) network high-speed (i.e., 1997 metrics, 90Mbps) connection was established for the Supercomputing (SC'97) conference. In our demonstration, HLRS, SNL, and PSC² were networked. Next, a general call of participation was issued to have international projects demonstrated in a common booth, the first International Grid, at

¹ Electronic Visualization Laboratory, University of Illinois at Chicago, <http://www.startap.net>.

² Sandia National Laboratories (SNL), US; Pittsburgh Supercomputing Center (PSC), US; Manchester Computing Centre (MCC), England; National Center for High-Performance Computing (NCHC), Taiwan; High Performance Computing Center (HLRS), Germany; Tsukuba Advanced Computing Center (TACC), Japan; Japan Atomic Energy Research Institute (JAERI), Japan.

* Corresponding author.

E-mail addresses: arbreck@sandia.gov (A. Breckenridge), lgpiers@sandia.gov (L. Pierson), sergiu@psc.edu (S. Sanielevici), welling@psc.edu (J. Welling), keller@hlrs.de (R. Keller), woessner@hlrs.de (U. Woessner), schulze@hlrs.de (J. Schulze).

SC'98. Our team provided a two-way Trans-Atlantic immersive collaborative session including force feedback based on a dataset distributed and computed at two supercomputer centers. Next, at the Second International Grid 2000, our team provided three-way Trans-Pacific immersive collaborative session including force feedback based on a dataset distributed and computed at seven supercomputer centers (see footnote 2) using the high-speed network (i.e., 2000 metrics, 90 Mbps still). Now, our accomplishments will be documented using the Third International Grid high-speed network (i.e., 2002 metrics, 10 Gbps).

2. Related work

Our project consists of three distinct focuses: compute, visualize and collaborate. The challenge is how to provide ubiquitous functionality by distributing, on-demand, data-intensive applications and then, provide an environment for collaboration and analysis of the information. Numerous teams are working to provide innovations. Related work in each area will be mentioned at the beginning of each subsection. Also, our tools can potentially be applied to any content area. Our team has demonstrated some of our previous Igrid work in volume, finite element, surface, and VRML rendering applied to architectural design.³ At Igrid 2002, our demonstration application was bioinformatics, specifically intron/exon splice sites [7] graphically shown after computing secondary structure with a parallel version of RNAfold modeling and simulation code.

3. Focus area: compute

3.1. Related work

With contemporary computers, the ab initio computations of the folding of long primary sequences into their corresponding tertiary (3D) structures are prohibitive with the current status of which problems are tractable and/or NP complete given in Fig. 1. The degrees of freedom per base are just too high. Finding this geometry is essential for understanding

the functionality. Since research suggests that folding from single-stranded chain to tertiary structure is a hierarchic process [2], the tertiary structure may be derived through the computation of the planar secondary structure. Still, the calculation of the secondary structure of RNA is computationally expensive. To speedup the calculation, high-performance parallel computers were used with the MPI-parallel version of the program RNAfold. This program was derived from the ViennaRNA package, developed at the University of Vienna [4,9] and improved in the frame of this project to be both efficiently parallel and grid distributable. For the minimum free energy of the sequence, all entries of two triangle matrices F^B and F^M of size $(n \times n)$, with n being the length of the sequence, have to be computed. The overall complexity of the algorithm is in $O(n^3)$. The memory requirement for matrices in minimum free energy calculation module is in $O(n^2)$. The work is equally divided among all processors along the diagonal of the matrices. In each of the n time steps i , each of the m processors computes $(n/m - i)$ values. The memory requirement regarding the matrices within the minimum free energy calculation module is

$$M = \left(\frac{4d^2 + 2d(u_{\max} + 2)}{2N} \right) \text{ size of (int) bytes,}$$

additional memory is consumed by several linear arrays.

3.2. Parallelism, communication speedup

Our research was applied to the RNAfold code with the minor enhancements in input and output format to allow alignment to a single coordinate space for comparisons in a spatial visualization program.

The serial version of the RNAfold program had a severe limitation on the number of bases that could be computed using the resources of one machine such as memory. Also, the initial parallel version had not been kept up to date with the serial versions since high communicational requirements as shown in Fig. 2 also severely limited the number of bases. After completing a diagonal, each processor previously did one send and one receive in the order of $O(n)$.

First, the serial version of the code with its improved energy parameters was ported to the parallel version. Second, our primary work included major

³ <http://www.cs.sandia.gov/ilab> and <http://www.hhrs.de>.

	DNA	RNA	Proteins
Primary			
In Silico Status	Tractable	Tractable	Tractable
	4 Nucleic Acids (ACGT)	4 Nucleic Acids (ACGU)	20 Amino Acids
Secondary			
In Silico Status	Tractable	Tractable	NP Complete
Course Gram View	Yes; Double Helix	Yes; Free Energy Model of Wrapped Single strand	Partial; Lattice Model; Chain of Amino Acids linked by Hydrogen Bonds
Stable Structure	Yes	Yes	No; often partially folded molten globule state
Evolution	Conserved	Conserved	Unknown
Important folding intermediate	Yes; nucleic acids have a simple specific interaction with pairing partner	Yes; nucleic acids have a simple specific interaction with pairing partner	No; amino acids have many-point interactions with side chains
Tertiary			
In Silico Status	NP Complete	NP Complete	NP Complete
	Supercoiled and wrapped with histones	Partial; in vitro shows same base pairs in secondary structure as tertiary structure.	Additional attractions present between alpha helices and pleated sheets
Quad			
In Silico Status	NP Complete	NP Complete	NP Complete
	Forms complexes with Proteins	Forms complexes with Proteins	Forms complexes with Proteins

Fig. 1. Status of simulations.

enhancements to remove the limitation on the number of bases by re-programming the code using strong grid or distributed programming techniques. The pre-grid code either sends a row or receives a column from his right neighbor, or sends a row or receives a column

from his left neighbor. Also, rows are completed on process 0, which then distributes the row to the corresponding storing process. To examine the communication pattern, the first computations were done with VAMPIR [8] on the Cray T3E installed at the High

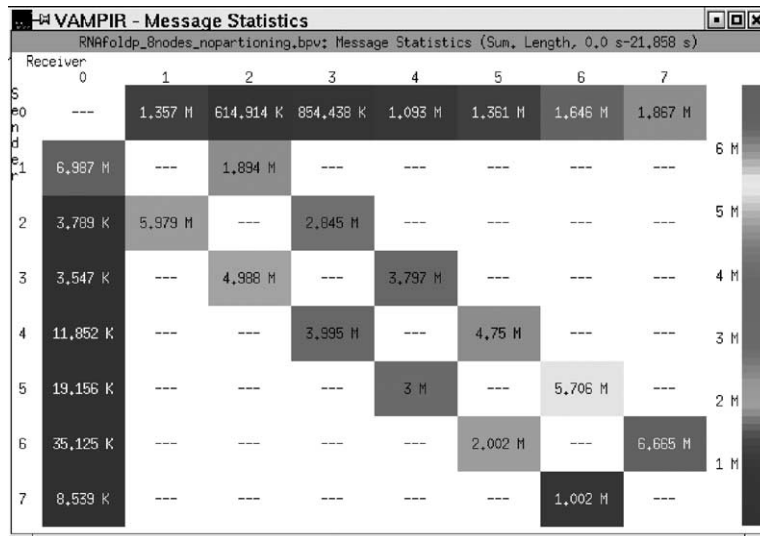


Fig. 2. Message statistics: data sent between processes (eight processes, 1000 bases).

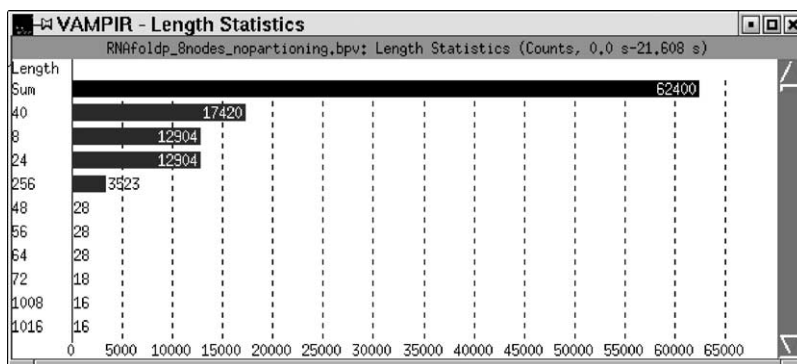


Fig. 3. Message statistics: number of messages sent, sorted by length (eight processors, 1000 bases).

Performance Computing Center in Stuttgart. Fig. 2 shows the amount of communication between each of the eight processors calculating a 1000 bases test case. The main communication pattern is between processors along the diagonal, sending the rows/columns needed by the adjacent processor in a non-blocking way, i.e. communication may be hidden by the computation of the diagonal. The numbers of messages were exorbitantly high, while the amount of data being sent is small; by far the most messages sent were 40 bytes long as seen in Fig. 3.

At the end of the computation, process 0 reassembles the folding of minimum energy with values of matrix F^B and F^M . Sending a request of three integers to the process storing this particular value and sending back one integer was done several times. Furthermore, these inefficient communications were often requested twice, some were asked for even more often. The maximum we observed in the 1000 base pair test case were 700 times.

Our grid-enabled code improves the communication pattern to increase the efficiency of the communication:

- Communication overhead was reduced by coalescing six messages into one message.
- Sending completed columns from process 0 to the corresponding process storing the column were merged.
- Two “received” operations unnecessarily used the MPI “any-source” semantic.
- Caching of data that is sent over the network several times.

Benefits of caching of the values were gained even on HPC machines with fast internal network. Examination of the request pattern for matrices F^B and F^M , showed that the accesses follow a strict pattern. Matrix F^M access pattern may be seen in Fig. 4. It shows a linear access with temporal property and every possible direction (top to bottom and vice versa and left to right and vice versa).

An algorithm was implemented for predicting a particular access pattern, prefetching a fixed number of values and caching them. The access pattern of F^B was different. The requested values did not follow a strict regional property; still they were mostly within a square-sized boundary, as shown in Fig. 5. For this matrix, a fixed-sized square of values were requested and cached.

3.3. Distributed computing

The enhanced MPI-version of the program was run on multiple grid-enabled platforms distributed worldwide. In addition to the Stuttgart equipment (a Cray T3E and an Origin 2000), the Terascale Computing System (TCS) at PSC was used for our calculations. For example, the RNAfold calculation of 78,067 bases of the human MSH6 gene ran in approximately 44 min on 512 processors of TCS. With its 3000 Alpha EV68 processors, the TCS offers an overall performance of six TF peak performance. Not only does it offer the computational power, but it also fits the memory requirement of our code. For further tests, the TCS will be also integrated into International Computational Grid at SC2002 using PACX-MPI [3].

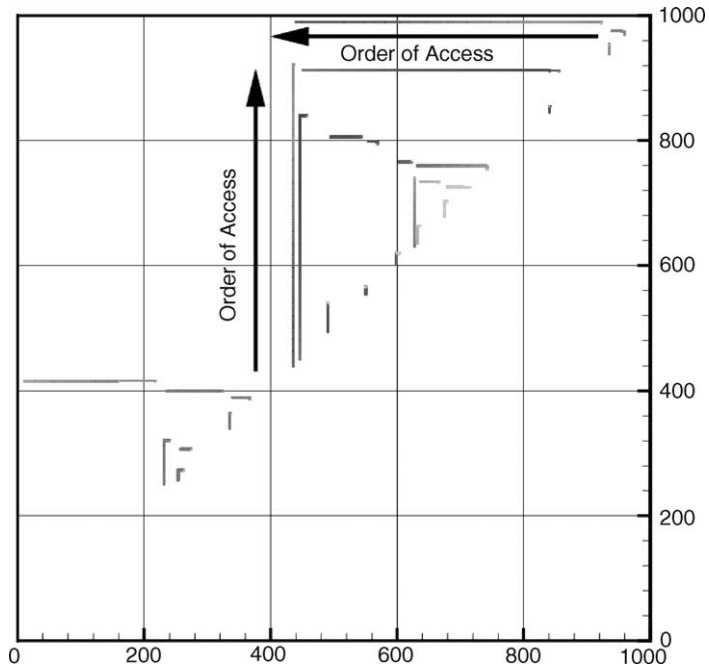


Fig. 4. Access pattern of matrix F^M .

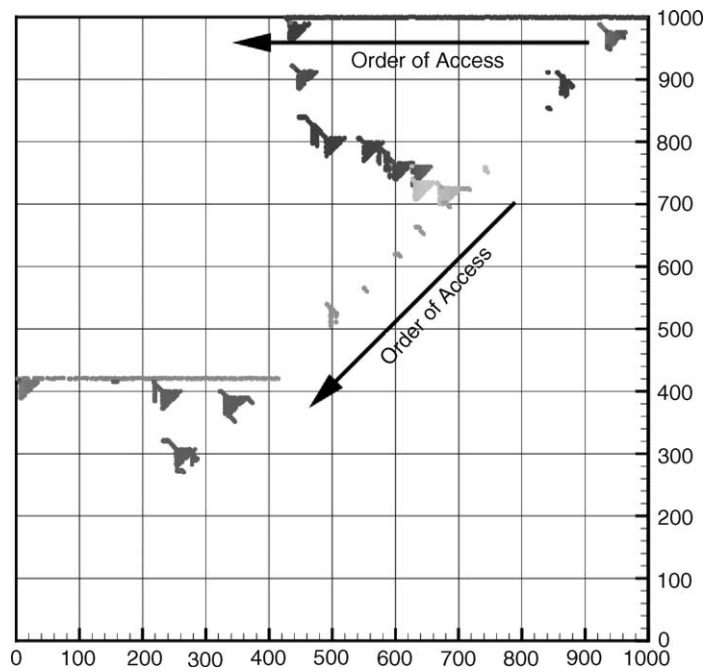


Fig. 5. Access pattern of matrix F^B .

4. Focus area: visualize

4.1. Related work

The current RNAfold basically had two types of outputs: (a) postscript file and/or (b) a circular visual representation, limited to 3000 bases [9]. Both representations could be viewed with traditional 2D image tools such as the Adobe™ suite or Acrobat readers. Our team created an extended output file format that contains the following information:

- Comments for locus, description and accession.
- Original primary sequences.
- Calculated secondary sequence in paragraph/dot syntax (i.e., ((.)).).
- Coordinates in X,Y plane.
- Pairs of nucleotides.

4.2. Graphical user interfaces (GUIs)

To visualize the folded secondary structure, the project used a common plug-in architecture into two separate visualization 3D OpenGL frameworks. The contribution from Sandia was a desktop application with a haptics interface where the sense of touch was mainly used for six degrees of freedom navigation. The plug-in was added to the e-Touch™ Graphical/Haptic User Interface [1]. The contribution from HLRS was an immersive virtual reality application with head tracking used in the CAVE and/or high-resolution power walls. The VR application was developed as a plug-in for COVER [6] that is the VR component of the collaborative visualization and simulation framework COVISE [5]. Both graphical environments allow easy extension through a flexible plug-in system and utilized a common set of visualization functionality (Fig. 6).

4.3. Common plug-in functionality

The customer can interact with the visualization through a 3D menu system and through direct interaction (i.e., 3D widgets, touch activated) with the abstract nucleotide representation of the objects. New components can be dynamically loaded to the system during runtime. The user can select multiple solution files and place them side by side or one over the other

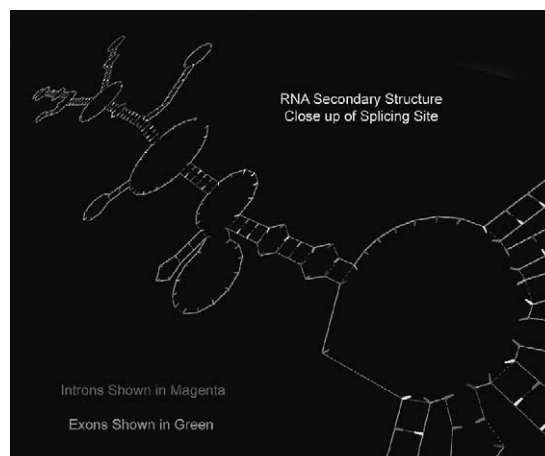


Fig. 6. Sample screen from e-Touch™ GHUI.

to analysis them visually. One can select from a variety of different display styles to differentiate between bond types or exon and intron regions. Each dataset maintains all the original primary sequence information with menus to show one base, groups of bases, or all bases if desired. Advanced visualization features for level of detail and area of enhanced functionality are applied. Optimization of the data representation ranges from two triangles for each nucleotide to 400 triangles for complete XML/X3D representation of a base.

4.4. Unique plug-in functionality/computational steering

An additional client/server plug-in was developed to allow the analysis of folded pre-mRNA or mRNA. The RNAfold simulation program acting as the client, connects to the graphics application, in this demonstration, COVER, acting as the server. The 3D GUI allows the customer to select between pre-computed results or couple to a running simulation of parallel RNAfold and receive real-time computed results. Whenever the RNAfold simulation code is started without a specific dataset to compute and an environmental variable, it connects to a visualization machine and waits for a simulation task to be scheduled for it. The customer in the VR environment can select from a list of available simulation hosts and assign a specific RNA sequence to these computers for simulation. As soon as the

computation finishes, the result is transferred to the visualization machine and is available for display. The result is, also, stored on the visualization machine's disk so that it can later be viewed offline. Multiple simulations can connect to a visualization machine dynamically and simultaneously, which allows the customer to submit a whole series of jobs. Also, the GUI is available for the analysts to choose all parameters of RNAfold, such as temperature, lonely pairs, or disallowing G–U bonding, etc. During runtime, the simulation reports its progress to the visualization machine with a percentage complete widget, so the analysis knows when one can expect the result of a certain simulation. The GUI thus has two grid functions: easy use of submitting batch jobs to the compute nodes and computational steering of smaller sequences in real-time.

4.5. *Unique plug-in functionality/automatic feature detection*

An additional hardware plug-in was developed to allow automatic feature detection of folded pre-mRNA or mRNA. The haptics plug-in was developed to show a visual “Start Here” arrow. The arrow has a magnetic field to draw the analyst to the beginning of a sequence in a 3D environment. Thus, the human's unique skill to detect sequences and motifs is used. The arrow widget allows for easy manipulation of one sequence for alignment with another.

5. Focus area: collaborate

5.1. *Related work*

In addition to computing and visualizing the RNA structure, the primary goal was to gain insight into the information. The project concentrated on linking biologists from around the world to perform analysis, and then addressed the extended functionality needed in a collaborative environment.

5.2. *Tele-presence issues*

Each International Grid partner shared a base set of functionality for collaboration. For audio and video

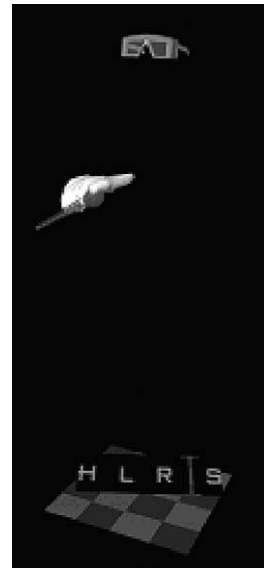


Fig. 7. Avatar.

conferencing, the AccessGrid infrastructure⁴ was used.

In the COVER and the e-Touch™ environments, audio conferencing was used extensively to communicate with other analysts. Traditional video conferencing was not activated in either environment, instead in the COVER environment, a graphical avatar representation was used which consisted of stereo glasses, a hand with a tracking pointer, and a grid on the floor (Fig. 7).

5.3. *Navigation control issues*

Support for collaborative work means that multiple people at different sites can join a visualization session either from a desktop, or from a virtual environment like a CAVE or a power wall. The results of an online simulation are transferred automatically to all participants in a collaborative session. The same applies for stored datasets. If they are not available at one of the participating sites, they are transferred automatically and stored for later use. In COVER, different collaboration modes that offer different strengths of synchronization are implemented.

⁴ <http://www.accessgrid.org>.

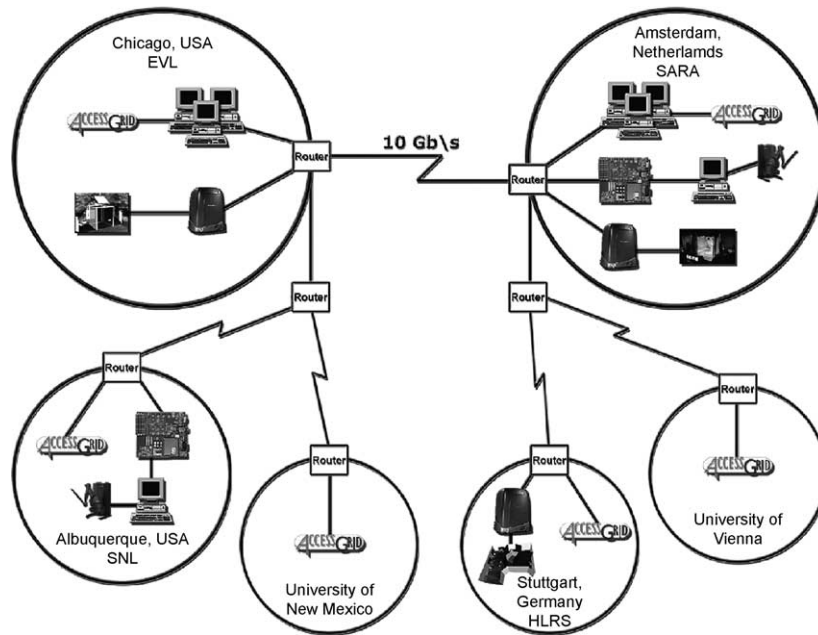


Fig. 8. Collaboration environment at International Grid 2002.

Master/slave: In this mode, all virtual worlds are tightly coupled and only one of the participants, the master, is allowed to interact with both the user interface and the objects in the scene. This mode is especially well suited for presentations.

Tight coupling: This mode also synchronizes all features of the application and the user interface, but it allows all collaborators to interact. This mode works well for joint work on a small dataset, like a smaller gene sequence or if all participants want to have an overview of a large dataset.

Loose coupling: The state of the virtual world stays synchronized but different users can have different views of the dataset. All partners are allowed to navigate independently of each other and not all features of the application are synchronized. The different users are displayed as avatars.

5.4. Collaborative analysis scenario

At International Grid 2002, our demonstration coupled three CAVEs in Amsterdam (SARA), Chicago (EVL), and Stuttgart (HLRS). In this collaborative session, analysts connected online to Supercomput-

ers at PSC, SARA and HLRS. The amount of data transferred for synchronization of the three CAVEs was about 40 Kbps, additional bandwidth was used for audio conferencing (~250 Kbps) and optionally video conferencing (~400 Kbps). Network bandwidth is not a central issue for navigation controls of an application, while latency definitely has to be taken into account. To deal with this, we implemented our navigation and object manipulation tasks in such a way that one partner never has to wait for acknowledgements from all partners to start an interaction (Fig. 8).

5.5. Data replication issues

Another major issue addressed by this project in building a collaborative environment is access to the data. Our team's robust multi-connection virtual environment has for several years fully replicated the data at each site. This severely limits administration of such an environment due to the level of hardware and hardware support needed at each site. This issue has been identified and an initial solution has been prototyped to only send a

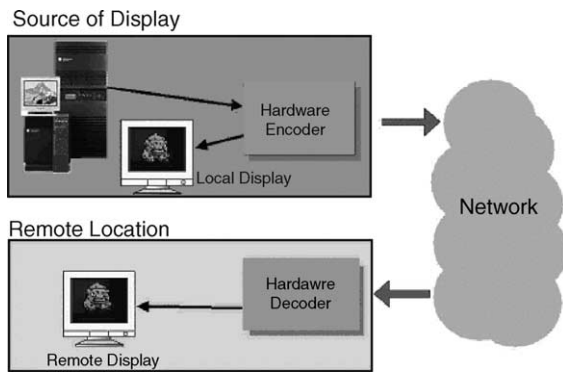


Fig. 9. Interactive remote visualization hardware operational concept.

completely rendered graphical scene via multicast to all participants. Sandia-developed hardware was demonstrated to implement a low latency digital video compression system for remote interactive visualization (Fig. 9).

This system receives high-resolution DVI (or RGB) video data, compresses and formats the data for transmission over an IP/GBE network, and decompresses and displays the data on a DVI (or RGB) monitor at the distant end. The firmware currently supports the $1280 \times 1024 \times 60$ Hz format, yet the hardware can support formats up to $1920 \times 1024 \times 30$ Hz. The equipment captures video data directly from any video graphics accelerator and transports this data in a compressed form over a Gigabit Ethernet network using standard Internet protocols. The equipment supports low latency, interactive visualization by transporting keyboard and mouse inputs from the remote end back to the originating computer. Eventually, the participants will have a simple display, interaction device, and the DVI/GBE decoder at each site.

The interactive remote visualization hardware system is designed to allow the remote visualization user to see a computer image as if sitting at the computer console. For a great many applications, one must have confidence that the image is a true representation of the data and not an artifact of the compression. The compression algorithm implemented delivers a faithful pixel-by-pixel rendering of the original image without compression artifacts. Building and maintaining hardware resources such as ultra-computers at some locations can frequently be impractical. For example,

a viable solution may be to perform data computation and rendering at an existing supercomputing site, and to then transfer the rendered images to the remote location. However, problems arise when this is attempted because it is difficult to provide transfers of a quality resolution and with an acceptable display frame rate. The extent of user interactivity, and consequently of usability, of these transfers is dependent not only on the frame rate, but also on the delay with which the images are presented. The problem lies in developing an apparatus to handle the transfer of this information at a speed acceptable for human interaction and retain images of a high quality. Equipment currently available delivers either low latency, low-resolution images such as video conferencing systems; or high latency, high-resolution images such as used by entertainment systems. The Sandia-developed hardware allows for the transfer of such information while maintaining an acceptable frame rate and resolution, as well as interaction with the visual data from a remote location.

Sandia's early experiments in this area used commercial hardware with frame updates at 30 fps, but at the relatively low-resolution of 640×480 pixels in interlaced NTSC format. At this stage, the equipment transferred compressed video over a network at 8 Mbps. An upgraded implementation utilized four independent video feeds, which raised the resolution to 1280×960 pixels. However, this arrangement proved to be expensive and difficult to setup. The spatial reconstruction and color matching of the four quadrants of the image required careful and frequent adjustment. Switching from NTSC format to the European PAL standard provided better color matching. This early hardware was developed to improve resolution and interactivity, and to eliminate the use of four separate feeds of the lower 640×480 resolution in the previous commercial hardware implementation. The conversion to analog RGB from the previous PAL format was made at this stage in the development to reduce scan conversion losses incurred primarily from interlaced sampling. The early prototype evolved into the hardware demonstrated at Igrid 2002, with several changes to the hardware. The current hardware interfaces to both Gigabit Ethernet and ATM networks, and incorporates both RGB and DVI video interfaces. The current hardware accepts keyboard and mouse input for transport to the remote location.

5.6. Remote visualization scenario

Using the interactive remote visualization hardware, RNA folding visualization generated in Chicago was manipulated from Amsterdam and visa versa. The equipment achieved compression ratios of complex changes of approximately 20:1. The 2.5 Gbps (1280 × 1024 × 60 Hz) of screen data directly from the GeForce4 graphic card's DVI interface was compressed into a 2/10 Gbps portion of an IP data stream carried over the Gigabit Ethernet Interface. Of the 98 ms network round trip delay between Chicago and Amsterdam, only 30 ms were added by the compression.

6. Conclusion

The overall demonstration was very successful. The compute, visualize, and collaborate components all functioned as a unit. In establishing the connections, audio services with remote partners were very important. The compute component needed to modify the base application to be efficient for grid computing. The visualize and collaborate components have established a valid environment to allow the biologist to fully examine RNA secondary structures in the COVER and e-TouchTM environments. Key issues in scaling and making this research readily available have been identified.

Acknowledgements

Special acknowledgements are in order for the networking and system administration teams at SARA that provided local support at the conference and the remote networking and system administration teams at HLRS, PSC, and Sandia. Special mention is in order for Alan Verlo, EVL, which provide local and remote support.

This work done by HLRS has been funded by the DAMIEN project (IST-2000-25406) and by the collaborative research centers (SFB) 374 and 382 of the German Research Council (DFG).

The computations were performed, in part, on the National Science Foundation Terascale Computing System at the Pittsburgh Supercomputing Center. The Pittsburgh Supercomputing Center is a joint effort

of Carnegie Mellon University and the University of Pittsburgh together with the Westinghouse Electric Company. It was established in 1986 and is supported by several Federal agencies, the Commonwealth of Pennsylvania and private industry.

Special acknowledgements are in order for the Sandia hardware design team consists of Perry J. Robertson, Karl Gass, Lyndon Pierson, Ron Olsberg, John Eldridge, Tom Tarman, Tom Pratt, Ed Witzke, John Burns, and Larry Puckett.

The work at Sandia National Laboratories has been funded by the Department of Energy's Advanced Simulation Computing Program (ASCI) and Mathematics, Information, and Computer Science (MICS). Sandia is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

References

- [1] T. Anderson, FLIGHT: a 3D human-computer interface and application development environment, in: Proceedings of the Second PHANTOM Users Group Workshop, Cambridge, MA, 1997.
- [2] R.L. Baldwin, G.D. Rose, Is protein folding hierarchic? Local structure and peptide folding, January 1999.
- [3] E. Gabriel, M. Resch, T. Beisel, R. Keller, Distributed computing in a heterogeneous computing, in: Lecture Notes in Computer Science, Springer, Berlin, 1998, pp. 180–188.
- [4] I.L. Hofacker, W. Fontana, L.S. Bonhoeffer, M. Tacker, P. Schuster, Vienna RNA Package. <http://www.tbi.univie.ac.at/~ivo/RNA/>.
- [5] D. Rantzau, U. Lang, R. Rühle, Collaborative and interactive visualization in a distributed high performance software environment, in: Proceedings of the International Workshop on High Performance Computing for Graphics and Visualization, Swansea, Wales, 1996.
- [6] D. Rantzau, K. Frank, U. Lang, D. Rainer, U. Wössner, COVISE in the CUBE: an environment for analyzing large and complex simulation data, in: Proceedings of the Second Workshop on Immersive Projection Technology (IPTW'98), Ames, IA, 1998.
- [7] M. Spingola, L. Grate, D. Haussler, M. Ares, Genome-wide bioinformatic and molecular analysis of introns in *Saccharomyces cerevisiae*, RNA 5 (1999) 221–234.
- [8] Pallas GmbH, VAMPIR—visualization and analysis of MPI resources. <http://www.pallas.com/e/products/vampir>.
- [9] M. Zuker, The use of dynamic programming algorithms in RNA secondary structure prediction, in: Waterman, M.S. (Ed.), Mathematical Methods for DNA Sequences, CRC Press, Boca Raton, FL, 1989, Chapter 7, pp. 159–184.



Arthurine Breckenridge is a Principal Member of Technical Staff in the Visualization Department at Sandia National Laboratories in Albuquerque, New Mexico. Arthurine has been involved in many aspects of networking and visualization during her 16 years at Sandia. Her work has won recognition with the Interop Award and leading researcher in haptics technology. She holds BS and MS in computer science from Oklahoma State University.



Rainer Keller is now a Research Scientist at High Performance Computing Center, Stuttgart (HLRS). He received his diploma degree in computer science at the University of Stuttgart in 2001. Since then, he is working on his PhD for the working group “Parallel and Distributed Systems” on the metacomputing library PACX-MPI. His research interests include MPI, Parallel-IO, networking and bioinformatics in general.



Lyndon Pierson is a Senior Scientist in the Advanced Networking Integration Department at Sandia National Laboratories in Albuquerque, New Mexico. Lyndon has been involved in many aspects of computer and communication security during his 27 years in inter-site secure communications at Sandia. His work in high-speed encryption has won recognition as a R&D 100 Award in 1991 and again in 1996. His current interests include the scaling of wide area communication technologies to achieve Supra-Gigabit/second secure network performance. He holds a BSEE from New Mexico State University and an MSEE from Stanford.



Uwe Woessner received his Diploma degree in mechanical engineering from the University of Stuttgart in 1999. From 1993 to 1999, he was working in several European projects as well as the G7 GWAAT Project together with Sandia National Labs. Since 1996 he is working in the Collaborative Research Center Rapid Prototyping established at the University of Stuttgart in the field of VR based virtual prototyping. He is now Head of the Hybrid Prototyping Working group which focuses on integration of physical and virtual prototypes as well as collaborative virtual environments



Sergiu Sanielevici is the Assistant Director of Scientific Applications and User Support at the Pittsburgh Supercomputing Center. He is responsible for high-performance, scalable, parallel code development.



Juergen Schulze received his Master's degrees in computer science from the University of Massachusetts at Dartmouth (1998) and from the University of Stuttgart (1999). He has been a PhD student in the visualization group at the High Performance Computing Center in Stuttgart since 1999. His work is focused on volume rendering in virtual environments. It involves the development of rendering routines on graphics hardware and remote rendering using parallel computers.



Joel Welling is a senior scientific software developer in biomedical and scientific visualization at the Pittsburgh Supercomputing Center. He is specifically interested in distributed volume-rendering applications.