Supercomputing's Role in Data Problems and Its Contribution to Solutions

Dr. Ke-Thia Yao, Dan M. Davis and Dr. Robert F. Lucas
Information Sciences Institute, University of Southern California
Marina del Rey, California

Dr. Philip Amburn
Science Applications International Corporation
Wright-Patterson Air Force Base, Ohio

High Performance Computing (HPC) has proven to be a blessing that unfortunately generates a conundrum, the solution to which is of major importance: Will the resultant glut of data overwhelm the user to the point of destroying the benefit of the advances in computation? The DoD can look forward with certainty to significant steady increases in computer performance and may soon experience an even more expansive revolution in this area. This article sets forth actual experiences with several instantiations of HPC battlefield simulations, discussing the potential value contained in the flood of data that they generated. The impediments to the effective use of that data are identified and alternative methods of reducing these impediments are surveyed. A solution, which is now being implemented and tested, is presented. Further subjects such as data mining and data visualization, both needed to enhance insights, are described and discussed. The article ends with an outline of the future and an over-arching vision for the appropriate response to the threat of data inundation.

I Introduction.

Since before the advent of written history, military commanders have sought ways to prepare their armed forces for upcoming battles. Some of this preparation involved creating plans of attack and some involved assessing the various capabilities of forces. Though today's commanders may use different tools, their goals would be recognized by the commanders of the past. Both would like to observe their troops in something akin to real combat and make judgments based on those observations. In the case of the early commanders, this might involve merely observing their troops (Judges 7:4-7, ~1255 BCE), but as the millennia have passed, the data available to the commander has grown to previously unimaginable proportions. The advent of industrial power has added to the commanders' burdens and our current military leaders have had their job skills extended and morphed into something more akin to logisticians than their ancient counterparts. Modern warfare allows little room for contemplation and adjustment.

To help ameliorate this problem, the US Joint Forces Command (JFCOM) has been designated as the transformation laboratory of the US Armed Forces. One of the research objectives of their Joint Experimentation Directorate (J9) is to assess the capabilities of various systems when deployed in an urban setting, they thus must “field” an urban population in experiments such as Urban Resolve (UR). This is a complex problem, calling into use all of the capable skills of the simulation programmers, system designers, experiment operators, and data analysts. They use Semi-Automated Forces (SAF) simulations, principally Joint SAF (JSAF).

Today, a single IA-32 based PC (*e.g.*, Intel Pentium) running with the Linux operating system can support a few thousand civilian entities. Lashing 20 to 30 of these computers together on a Local Area Network (LAN) using Ethernet can support a population of around 30K (Ceranowicz, 2002). Looking at any major urban center in the world, one would expect to see nearly two orders of magnitude more entities than that. For example, Baghdad has a population of 7.8M people, with vehicles adding at least another million. The LAN solution does not scale to this level. The JFCOM Joint Experimentation on Scalable Parallel Processors (JESPP) Project was initiated to respond to this problem using High Performance Computing (HPC) (Lucas, 2003). Utilizing the readily available capabilities of the HPC Modernization Program (HPCMP), the JESPP team was successful in scaling to over one million entities.

Thence came the data deluge. Like others, JFCOM analysts now faced increasingly large data sets, complex sensor results, convoluted scenarios, and diverse environments. The need to re-run certain actions of interest also heightened the need for collection, processing, management, storage and retrieval of the data. Literally terabytes of information are collected, and even that volume of data assumed that much (80%) of various background activity was not archived and therefore could not be exactly duplicated later for analysis.

In most high performance computing implementations, the data analysis is performed on discrete platforms that are separate from the data generation devices, such as test range instruments or physics sensors. These HPC facilities are often designed specifically to facilitate the manipulation of the data or to interface easily with sensors and other data-producing devices. The JFCOM situation is somewhat different. Here the data is originated on the HPC assets themselves. JFCOM is driven by several calls upon their services. These include:

- analysis of battlefields of the future, ten years or more
- design of systems intended for use in the next year or two
- real-time assistance to the warfighter today

These demands and the resultant operational tempo already requires the use of JFCOM’s clusters for analysis of the data that they generate on their clusters.

This paper discusses how simulation-based experimentation is increasingly overwhelmed by the data generated by increasingly powerful computers, and how the community is rising to this challenge. The remainder of the paper is organized as follows. Section two discusses the current state of affairs. Section three presents our vision of how the next generation of systems will exacerbate the problem. Section four describes the solution we are exploring for managing this data. Section five discusses the analysis and visualization tools we anticipate using. The paper ends with a brief summary.

II The source of the data deluge in Joint Experimentation.

The authors do not intend to burden the reader with technical descriptions of the format of the data produced by JSAF, culture, and the associated sensor federates used in Urban Resolve (see Graebener, 2003 for more details.) That which follows is a higher-level and more germane general description of the data available for collection and the steps necessary to render it useful to the analysts and experiment controllers. This should allow the reader to better assess similarities between their simulations and the simulation under study by the authors and provide a basis for analyzing the relevancy of this work to theirs.

For simplicity’s sake, we break down the JFCOM simulations into four broad areas: 1. Terrain and Environment, 2. Civilian or “culture”, 3. Operational entities, and 4. Sensor platforms

Of these, only the first broad category, Terrain and Environment does not represent a major data task. Consider that as a benchmark, when a global high resolution terrain database is NOT one of the major data issues! This is due to the fact that the terrain is a largely static entity and the environmental variables are often easily duplicated without storing the actual values during the experiment, *e.g*. the day/night interface is easily recovered, while the impact of clouds may be more random and need to be recorded.

On the other hand, the amount of data presented by the clutter of civilians can be huge. Positions of pedestrian entities and vehicle models are reported to the system on the order of once every 10 to 100 milliseconds. Being able to simulate millions of entities, (Barrett, 2004) now presents the problem of what to do with all of this location data (in three dimensions), orientation data (in three axes) and state data (color, type, damaged, dead, …) that can report as often as 100 times in a second. The JESPP team has developed a solution for saving data so generated (Wagenbreth, 2005) locally on the processor that generates it. However, as will be discussed below, the volume of data is so high that the only solution when it comes to collecting the data for analysis today is to simply discard the "culture" data, treating it much as if it were environmental data, albeit more dynamic.

Next are Operational Entities. These are largely armed forces of US, Allied and enemy units. JSAF, like all of the SAFs, presents the possibility of very good Human-In-The-Loop (HITL) intervention. That means that JFCOM military personnel can personally control US and Allied forces and a "Red" team can bring a hostile presence to the experiment with all of the creativity and experience they have garnered, frequently after decades in the service. This brings to the surface a significant difference between JFCOM data and, say, Bank of America "transaction data" or high energy physics "sensor data." The solutions sought for DoD use in both the software and hardware areas must be sufficiently general to support any conceivable data type and load, yet sufficiently specific to generate the validity necessary for JFCOM use. Neither the banker nor the physicist has so little control over their parameters.

Finally, there is the data to be gleaned from the intelligence sensors simulation programs. Currently, in Urban Resolve, the simulations generate nearly a terabyte of data per week. Were all of the data to be collected of a simulation of the resolution and sophistication that is possible, the amount would swell to around 80 terabytes per week. Just physically managing that much data is a daunting task, not to speak of the difficulty in reliably validating the relationship between simulated sensor output and simulated activity on the terrain database and facing proprietary data concerns.

The System

One of the great strengths of the JFCOM experimental design is its distributed and dispersed nature. The experiments themselves are housed and controlled by the JFCOM out of its experimental bay near Suffolk, Virginia. Environments and data are managed remotely out of Fort Belvoir in Northern Virginia. The civilian clutter, or more politely: culture, are laid down and managed by a team a continent away in San Diego, at the SPAWAR Systems Center on Point Loma. The two 128 node, 256 processor Linux clusters that are provided by the HPCMP, are located in Maui (named KOA, the Hawaiian word for "soldier") at the Maui High Performance Computing Center (MHPCC) and at Wright Patterson Air Force Base (named GLENN, after the Marine, Astronaut and Senator) at the Aeronautical Systems Center Major Shared Resource Center (ASC-MSRC) in Ohio. Communications between the sites are provided by the Defense Research and Engineering Network (DREN). Experiments have been unclassified, but work requires encryption. The Linux operating system is generally used and most of the programs are written in C or C++, with a growing smattering of Java.

Figure 1 Notional Multi Path WAN between,

TEC, JFCOM, ASC-MSRC, SPAWAR, and MHPCC

Note that there are geographical dispersion issues, Maui being on the order of five thousand miles from Suffolk, as indicated in the notional diagram in Figure 1. This precludes easily and economically meeting with the entire staff. Further, with a five time zone span, the operational synchronization is difficult, most especially in the summer when the mainland sites go to daylight savings time, while Hawai'i does not, thereby creating a six-hour difference.

The JFCOM experiments typically run for five days a week, ten hours a day. Depending on availability and requirements, one or both of GLENN and KOA were used to simulate up to two hundred thousand non-combatant "culture" entities. Several thousand non-clutter entities were simulated on the other sites. A single node on the large clusters simulated 1000-2000 clutter entities. Data logging was performed in two modes: near real time and after action. Real time data was inserted in an *sqlite* database. A node simulating 1000 clutter items would generate an *sqlite* database of approximately 50 MBytes in an hour. The databases were deleted and reinitialized when they grew to over a gigabyte. If 100 nodes of the cluster were used for clutter simulators, approximately five gigabytes per hour of data were generated. For after action use, compressed binary data was stored in an archive directory and those data are approximately seven times smaller than a corresponding database. Each night, the archived data were transferred, via *rsync*, to a data server, and then expanded and decoded into a single MYSQL database.

Clutter data from the GLENN and KOA clusters was not entered into the JFCOM database server in Virginia, due to size limitations. Data from 100 nodes on GLENN for a ten-day event would have been close to a terabyte. Data from TEC, SPAWAR, J9 and J9 mini-cluster for non-clutter entities were entered into the MYSQL database. A two-week exercise generated about a terabyte of data in the MYSQL database. The nightly data transfer was about 15 gigabytes of compressed data. Network transfer rate to Saber (JFCOM's data server) was approximately ten megabits per second. Three to four hours were required to do the rsync transfer. Decoding and indexing the data into the MYSQL database took 12 hours if everything worked perfectly.

Database queries used in Urban Resolve are generally summary in nature. They count how many events or entities (database rows) meet specified criteria. Complex join operations were rarely, or never, used. Were it not for this constraint on the queries, an efficient distributed design would be much more difficult.

III The expected impact of future technology on the data deluge

The future of large-scale simulations will involve both more sophisticated codes as well as new platforms that can support them. This will not help the data deluge. The treatment of the physical phenomenon such as weapons' effects, communications, sensors, and weather is currently limited to probabilistic table look-ups and is necessarily of very low fidelity. In the modern urban battlefield, these statistical techniques inadequately model collateral damage and fail to accurately or realistically model the behavior of the systems on which U.S. military effectiveness is increasingly dependent. Future HPC capabilities are on the horizon to address these problems. (Bunn, 2006)

New HPC developments include the Defense Advanced Projects Agency's High Productivity Computing Systems (DARPA HPCS). These proposed Petascale systems feature shared, global address spaces, and many hundreds of thousands of processors managed by new programming languages that make these systems accessible. New Petascale Forces Modeling and Simulation (FMS) codes will be able to have shared, dynamic terrain representations replacing today's replicated, static terrain models. A new generation of scaleable behavioral models will be created enabling the sophistication of individual battlefield entities to exceed the limits imposed by individual processors on today's codes. Finally, Petascale HPCS systems will make it possible to finally incorporate physics-based models of phenomena such as weather (restricting mobility, sensors, fighting effectiveness, etc.) with entity-level battlefield models.

A number of on-going efforts and programs can be leveraged to reach this goal of a truly multi-disciplinary, complete analysis capability. The FMS and the Test and Evaluation (T&E) communities will extend their previous achievements in using HPC. The program should exploit the experience of academic, commercial, and governmental experts whose expertise spans computational science, HPC, military forces, human behavior modeling, physics, communications systems, networking, systems testing, and information systems.

Future battlefield simulations and T&E environments must have virtually unlimited scale, sufficient resolution of the urban environment (buildings including the interiors, doors, windows, streets, etc.), and the ability to run faster than real time, all with the concomitant data surfeit. These simulations must support Human (and Hardware)-In-The Loop (HITL) capabilities, as no system is useful to the warfighter if the human or hardware interfaces fail.

IV The Simulation Data Grid for Addressing the Data Deluge

To deal with its data deluge and enhance its ability to support data analysis, the Joint Experimentation community is developing a new data management system called the Simulation Data Grid (SDG). This will be a new distributed data management application that helps the High Performance Computing user deal with very large, geographically dispersed data sets over heterogeneous environments. It is designed along the lines of the "Data Grids" pioneered by the high-energy physics community, but is tailored specifically for the needs of the forces modeling and simulation community.

The data management capabilities, essential to current and future DoD simulations, provided by SDG will include the capability to:

- collect and store high volumes and high rates of data from geographically distributed data sources
- browse high-level summaries and overviews of the stored data
- discover what part of the data has changed since last reviewed
- transport and redistribute the data
- query details of the stored data

These capabilities are applicable to multiple varied environments and domains in which large amounts of data are generated, such as distributed simulation and distributed testing. The types of people who use these capabilities include domain data analysts, system developers and higher-level decision makers.

The initial performance goal of SDG is to be able to support simulation systems generating rates of 100 GB of data per hour, 8 Terabytes for a two-week event, with data distributed across two supercomputers, and multiple sites, *e.g*. JFCOM's JUO experiment sites.

Unless they develop such a data management system, the DoD risks:

- losing valuable data from tests, exercises, and experiments
- being mired in a deluge of inaccessible and unusable data
- producing false analyses and unsupported conclusions
- limiting the performance of the source of the data

Leveraging Grid Computing

SDG is intended to operate in a joint experimentation environment, where the computing software and hardware elements may be quickly assembled on an *ad hoc* basis. The constituent elements may change depending on exigent needs and resource availability. Many T&E and FMS events stand up a local computing organization to solve specific problems. These systems span multiple administrative domains, each with its own security policies, yet each offering a unique combination of computing, networking and storage capabilities.

The goal of Grid Computing is to provide pervasive dependable access to distributed computing resources. The Grid Computing vision, if realized, promises access to computing as easily as people currently access the power grid through their wall sockets. The main focus of SDG is data collection and analysis, but in order for SDG to work effectively in a joint environment, it must also address many of the same issues that face Grid Computing.

Grid computing research focuses on developing an interoperable common infrastructure that provides dependable, consistent access to distributed computing resources. It addresses the problem of coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations (Foster et al., 2001). To emphasize the focus on developing interoperable tools and interfaces that work across platforms and organizations, Foster (2002) proposed a three-point checklist for grid computing:

- coordinates resources that are not subject to centralized control
- uses standard, open, general-purpose protocols and interfaces
- delivers nontrivial qualities of service.

Experience

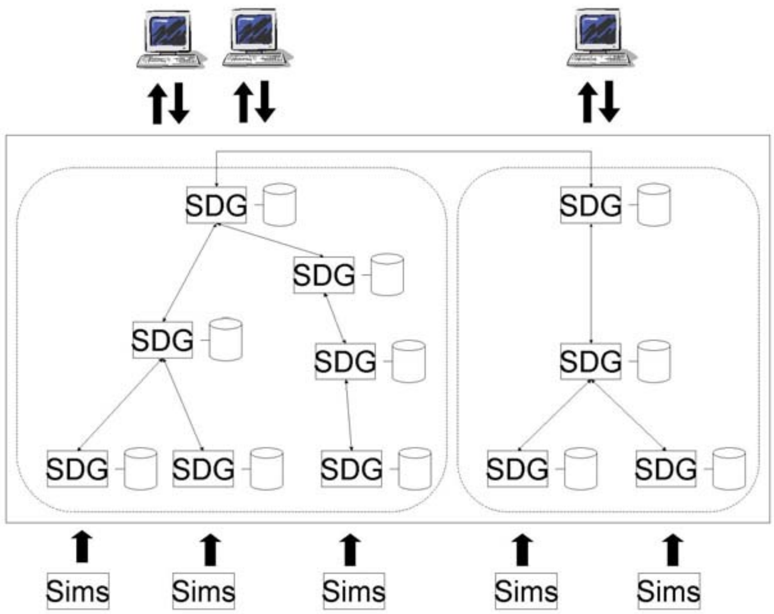

Figure 2 Designer's Conceptual Model.
Round-corner boxes indicate the boundary of local area networks

Figure 2 is a graphical representation of how the SDG functions in the FMS environment. SDG Managers perform all of the data access/query/management tasks. Conceptually, there are three types of SDG Managers: top-level, data source and worker.

Top-level Managers have published addresses. Users connect to the SDG through the top-level managers. To minimize network traffic, typically there is at least one top-level manager for each local area network. Top-level managers know how to connect to each other. Non-top-level managers know how to connect to at least one top-level manager.

Data Source managers store the actual data for SDG. SDG provides an API to allow other applications to insert data into Data Source Managers.

Worker managers perform most of the work within SDG. When given a data processing task, the top-level manager:

- decomposes a task into sub-tasks
- enlists one or more managers: worker, data source or other top-level managers.
- assigns sub-tasks to these managers
- defines a data flow topology linking together the sub-task executions.

Having corralled the data, now a prudent developer would look to how one could make the best use of the logged, stored and ordered information. SDG will ensure that it is sorted and indexed in a way that makes it easily accessible for user analysis, leading to the next section of this paper.

V THE FUTURE OF VISUALIZATION AND ANALYSIS

The circumstance of massive amounts of data and the need for better ways to interpret and understand the data is not unique or new. The National Science Foundation sponsored discussions and a workshop entitled Visualization in Scientific Computing (ViSC) in 1986-87, in which it is reported:

> "Visualization is an emerging technology that enables better computational science. Scientists already generate such torrents of information that they can no longer do much more than gather and warehouse it. Visualizing the results of complex computations and simulations is absolutely essential to ensure the integrity of the analyses, to provoke insights and to communicate them to others." (McCormick 87)

Change just a few words in the previous 1987 quotation and it is a good description of where the community is now with simulations of urban battlefields, while it is clear that the landscape of visualization in scientific computing has dramatically changed since the 1987 NSF report. Today, long-standing scientific disciplines such as computational fluid dynamics, computational chemistry and computational structural mechanics have a substantial collection of physics-based models in routine use on high performance computers. In addition to the physics-based models, these mature HPC disciplines also have a visualization toolbox full of tools tailored to aid research scientists in understanding their data.

How can the HPC community provide needed visualization tools to military analysts that will actually be used? It will require some cultural changes as well as some dedicated software efforts. The cultural changes are aimed at the all-too-common attitudes of "Who needs those extra pictures anyway" and "We've never needed that stuff before". Commonly, people in operational settings employ 2D maps and overlays, and anything beyond that is characterized as unnecessary and, in the past, technically advanced capabilities have been hard to field.

From a software development viewpoint, just as the mature computational fields have tailored visualization tools, the HPC need to expand the visualization and analysis tools available to military analysts. Developing software like this requires a multidisciplinary team. Here's the type of team identified in the NSF ViSC report:

- computational scientists and engineers
- visualization scientists and engineers
- systems support personnel
- artists
- cognitive scientists

A comparably composed defense-focused team would watch and learn from military analysts to see how they currently do their tasks, and use that information to develop visualization and analysis tools that will aid the analysts in their work. During this effort, it will be possible for the team to take advantage of some remarkable advances in two areas: open source software and game technology. There is a substantial collection of visualization tools, both commercial products and open source software. In the open source world, packages such as VTK, ITK and Paraview provide exceptional capabilities in quality, open source software. An emerging technology that needs to be exploited is Serious Games. Here the capabilities of modern gaming industry are used to develop games for serious use.

VI CONCLUSIONS

High Performance Computing has contributed to the flood of data that has overwhelmed the users and threatens to do so in the future. Forthcoming hardware advances will at first exacerbate, but can then ameliorate, the data problem. HPC data capability should keep pace with HPC data generation growth. As Richard Hamming observed many years ago "The purpose of computing is insight, not numbers".

Visualization and analysis tools are indispensable tools in the toolbox. And because of the amount of data being generated, HPC resources will be required during the visualization and analysis tasks. The use of large clusters (hundreds or thousands) of processors to meet the demand for high performance computing is a mature technology. The extension to use multiple clusters for the management of the ensuing data is likewise common, but less mature. To support the DoD we have combined distributed clusters and merging data management techniques. Using this approach, and with innovation in key areas, the FMS and T&E communities should be able to support current and expanding needs of the defense establishment. A key principle is to store data close to its source to minimize network traffic. A second principle is to utilize the computational and storage resources of distributed clusters for analytical functions. These two principles reinforce, rather than interfere with, each other in the design and implementation of the data grid. The queries required and commonly used by DoD analysts will be efficiently supported by this system. Fault tolerance and realistic data archiving are additional benefits of a well-planned implementation. The future will see the maintenance and extension of the system to include more processors, more clusters, larger datasets and more robust queries as required by our customer. The answer to the challenge of the of data is the intelligent use of HPC assets and techniques.

Acknowledgements

Most of the work on which this article was based was directed and funded by the Joint Experimentation Group at the Joint Forces Command and the authors wish to thank Lieutenant General Woods, Anthony Cerri, Jim Blank, and the entire J9 staff. None of this could be accomplished without the help of the JSAF team Dr. Andy Ceranowicz, Mark Torpey, Bill Hellfinstine and many others. This material is based on research sponsored by the Air Force Research Laboratory under agreement number F30602-02-C-0213 and FA3750-05-2-0204. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes, notwithstanding any copyright notation thereon.

Authors

Ke-Thia Yao *is a Research Computer Scientist, Distributed Scalable Systems Division, Information Sciences Institute (ISI), University of Southern California (USC), where he also lectures in the Viterbi School of Engineering. Currently he is working on supporting large-scale military simulations involving millions of simulated entities such as vehicles, systems and personnel. He has developed a suite of monitoring and logging analytical tools to aid users in better understanding the implications of both the computational capabilities and the behavioral insights associated with large-scale simulations. He received a Bachelor of Science degree from the University of California, Berkeley, and Master of Science and PhD degrees from Rutgers University. Email:* ***kyao@isi.edu***

DAN M. DAVIS *is Director, Joint Experimentation on Scalable Parallel Processors (JESPP) Project, Information Sciences Institute, University of Southern California, and has been active for more than a decade in large-scale distributed simulations for the Department of Defense (DoD). While he was assistant director of the Center for Advanced Computing Research at the California Institute of Technology (Caltech), he managed Synthetic Forces Express, a multi-year simulation project. He saw active duty in Vietnam as a cryptologist in the Marine Corps, and he recently retired as a Commander, cryptologic specialty, USNR. He received Bachelor of Arts and Juris Doctors degrees from the University of Colorado in Boulder. Email:* ***ddavis@isi.edu***

ROBERT F. LUCAS *Director of the Computational Sciences Division of ISI at USC, where he manages research in computer architectures, very large-scale simulations, compilers, and other software tools. He has been the principal investigator on the JESPP project since its inception in 2002. Prior to joining ISI he was head of High Performance Computing Research Department for the National Energy Research Scientific Computing Center at LBNL. Earlier, he was deputy director of DARPA's ITO and a member of the research staff of the Institute for Defense Analyses' (IDA's) Center for Computing Sciences. He received his Bachelor of Science, Master of Science and Ph.D. degrees in electrical engineering from Stanford University. Email:* ***rflucas@isi.edu***

PHILIP AMBURN *currently works for Science Applications International Corporation (SAIC) as the Programming Environment and Training On Site for Forces Modeling and Simulation, with an office at Wright-Patterson Air Force Base (AFB), Ohio. His research interests include constructive and virtual simulation, interactive three-dimensional (3-D) graphics and visualization. He retired from the U.S. Air Force in the rank of lieutenant colonel. Dr. Amburn received a Bachelor of Science degree in physics from Kansas State Teachers College; a Master of Science degree in computer science from the Air Force Institute of Technology; and a Ph.D. in computer science from the University of North Carolina, Chapel Hill. Email:* ***philip.amburn@wpafb.af.mil***

References

Barrett, B. & Gottschalk, T. (2004a), Advanced Message Routing for Scalable Distributed Simulations, *Proceedings of the 2004 I/ITSEC Conference*, Orlando Florida

Bunn, J.J. & Gottschalk, T.D., (2006, pending), Incorporating High Energy Physics Data Capabilities into Large-Scale The DoD Simulations, *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference*, Orlando, Florida, Publication Pending

Ceranowicz, A., Torpey, M., Hellfinstine, W., Evans, J. & Hines, J., (2002), Reflections on Building the Joint Experimental Federation, Proceedings of the 2002 I/ITSEC Conference, Orlando, Florida

Davis, D., Baer, G., & Gottschalk, T., (2004), 21st Century Simulation: Exploiting High Performance Parallel Computing and Advanced Data Analysis, *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference*, Orlando, Florida

Davis, D.M. & Lucas, R. F., (2006), Joint Experimentation, Data Management and Analysis Enabled by Trans-Continentally Distributed Linux Clusters, *HPCMP Users Group Conference*, Denver, Colorado

Foster, I. & Kesselman C., (1997), Globus: A Metacomputing Infrastructure Toolkit, *Intl Journal Supercomputer Applications*, 11(2): 115 –128

Graebner, R., Rafuse, G., Miller, R. & Yao, K.-T. (2003) *Proceedings of the 2003 I/ITSEC Conference*, Orlando Florida

Judges 7: 4-7, (~ 1255 BCE), *The Holy Bible*, Revised Standard Version, Philadelphia, Westminster, 1952 (the story of Gideon selecting troops at the river)

Lucas, R., F. & Davis, D. M., (2003) Joint Experimentation on Scalable Parallel Processors. In the *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference*

McCormick, B. A., DeFanti, T. A., & Brown, M., eds., (1987),*Visualization in Scientific Computing, Computer Graphics*, volume 21, no 6

Wagenbreth, G., Yao, K-T., Davis, D., Lucas, R., and Gottschalk, T., (2005), "Enabling 1,000,000-Entity Simulations on Distributed Linux Clusters," *WSC05-The Winter Simulation Conference*, Orlando, Florida