Dr. Scott A. Klasky

The University of Southern Mississippi
Department of Computer Science and Statistics
Hattiesburg, MS 39406-5106

Research Summary

Over the past 8 years my research interests have been solving large-scale Partial Differential Equations on supercomputers. My main research interest was in the field of numerical relativity. Since the Einstein equations are a set of many coupled equations with changing length scales in time, this problem is extremely computationally intensive. To handle these problems, we had to learn to use the latest numerical algorithms, as we as use the next generation of computers, capable of Tera-flop speeds and Giga-byte storage. A major goal of numerical relativity is to be able to predict characteristics of the gravitational radiation of astrophysical events, which should be observed with the LIGO/VIRGO gravitational detectors. Computationally, these problems all deal with solving highly Nonlinear Partial Differential Equations. This is where most of my research interest lies.

One of the first large scale codes I designed was used to solve the initial value problem for two black holes, along with Matthew Choptuik and Richard Matzner. I have written numerous solvers for elliptic partial differential equations, with non-trivial boundary conditions. Some of the more popular techniques that were employed include: Successive Over Relaxation methods, Domain Decomposition, and Multigrid techniques. I have also developed several adaptive codes based on the Berger and Oliger algorithm. Since my codes can handle general boundary conditions for elliptic partial differential equations, it should be trivial to model other physical processes.

After solving this problem, I later become the head Cognitive Scientist for the “Binary Black Hole Grand Challenge Alliance” main evolution code. Recently we began publishing results from this code showing we were the first to: evolve a 3D black hole forever, and translate a 3D black hole across the computational domain. This code evolves black hole data using Einstein’s equations. This code is an extremely long F90 code (over 60K lines) which I help write over 50% of this code. It runs on parallel machines and includes the capability of using Adaptive Mesh Refinement (AMR). This technique is used to place computational grid points where the physics dictates they should be. In other words, the amount of physics is proportional to the amount of grid points used in the simulation.

In creating this code, I noticed that we needed better visualization support, so along with a graduate student at NPAC, I designed and implemented, Scivis, to help with the data analysis. Recently Scivis has been getting national recognition, and is starting to be used in many universities and corporations (University of Texas @Austin, NCSA @ University of Illinois, Chung-Hua University, Failure Analysis Associates, Institute of Geography, FSU Jena, Max-Planck-Institut fuer Gravitationsphysik, Cornell University, Drexel University, etc.). This visualization package is written in Java (over 45K lines), and thus, it enables researchers to visualize supercomputing simulations from their PC’s at home, as well as their workstations at work. There is a tremendous amount of work that we still need to do in order to fully implement Scivis for a variety of applications. One of it’s most serious drawbacks is its speed in rendering complex visualizations. Java 3D will take care of this, and we will be including this in the next several months.

Another research topic that I have been working on for the past 6 months is financial modeling. In particular I have developed a Monte Carlo code to model complex financial instruments. In this proprietary code, we use a simulated annealing technique for the Monte Carlo Metropolis algorithm to generate results which might deem to be important by Wall Street. We are in a close collaboration with a company on Maui for this project.

In summary, in order to solve large scale Partial Differential Equations, I have become an expert on: parallel computing, scientific visualization, state-of-the-art numerical algorithm development and collaborative software. I have also become the “group” leader of many of these research activities.

Summary of Research Plans

Over the past 2 years, as a research scientist, I have realized that I get great joy in teaching, and working with students. One of my major reasons for trying to get a faculty position involves my interest in teaching students state of the art computational and visualization techniques. My Research Plans are as follows:

1. One of my immediate goals is to further develop my collaborative scientific visualization code (Scivis) such that it will become “more” usable in many areas of research (such as reservoir modeling, etc.). This will involve with creating more API’s to allow Scivis to better interact with finite element codes, and to allow more types of visualization techniques.

2. Computational steering of complex numerical codes is another area of research which I plan to work in. My plans are to develop a highly modular system consisting of API’s which will allow my Java front end to interact with simulation codes. This Java front end will then interact with visualization servers (such as Scivis) to allow researchers to interactively visualize and steer their computations. There is much work that is needed in this field since I envision my system to be fully collaborative, thus to allow a large number of researchers to control a simulation simultaneously.

3. Development of a extendable numerical toolkit to allow researchers to develop parallel numerical codes easier. This Project will design an extendible workbench using Java with CORBRA and allow researchers to design codes (such as multigrid codes), effectively and efficiently using a visual form of programing. The unique aspects of this research is that researchers will be able to design their numerical code, and design them sequentially using tools in the toolkit, and then further extend these tools, which can be re-used. Each module in the toolkit will be written with parallelization in mind, thus to allow researchers to achieve a modest level of parallelization, with a minimal level of effort.

4. Further effort for development of parallel solvers for complex financial instruments. One of the “hottest” new areas of research is in financial modeling. This field requires fast solvers, which requires efficient algorithms on parallel computers.

Equipment Needs

In order to develop code for scientific visualization I will need: 1 SGI Octane, and 1 Pentium II workstations running Linux, and 1 Pentium II laptops. I believe that the majority of the scientific visualization community utilizes SGI’s, and software that we design should be designed with these system in-mind. Pentium II machines running Linux are probably the best Java machines of today. I would need this system to have Open GL accelerators, such that my scientific visualization system would be able to use this hardware rendering capability of this system. The laptop is a machine which I need in order to have software also developed for the NT platform, as well as serve as a way to produce a LAN between the Linux machine and the NT machine. Both machines are capable of running MPI, and I could design many prototype systems on these 2 machines.

NOTE

I did not include my supervisor, Geoffrey Fox on my research interest because of my current position under him. I am a research scientist under Geoffrey Fox, and I wish him to not know about my job application, for fear of losing my position.