LA-UR- 04-6078

Approved for public release; distribution is unlimited.

Title: An Overview of the Los Alamos Crestone Project: Uses for Astrophysical Problems

Author(s): Robert P. Weaver, Michael L. Gittings, Galen R. Gisler, Robert F. Coker, Kimberly C. New and Robert M. Hueckstaedt

Submitted to: AMOS Technical Conference
Maui, HI
September 13-17, 2004

Los Alamos
NATIONAL LABORATORY

AN OVERVIEW OF THE LOS ALAMOS CRESTONE PROJECT: USES FOR ASTROPHYSICAL PROBLEMS

Robert P. Weaver[1], Michael L. Gittings[1,2], Galen R. Gisler[1], Robert F. Coker[1], Kimberly C. New[1] and Robert M. Hueckstaedt[1]

[1] *Los Alamos National Laboratory*
Los Alamos, NM 87545
[2] *Science Applications International Corporation*
10260 Campus Point Drive
San Diego, CA 92121

ABSTRACT

The Los Alamos Crestone Project is part of the Department of Energy's (DoE) Advanced Simulation and Computing (ASC) program. The main goal of this project is to investigate the use of continuous adaptive mesh refinement (CAMR) techniques for application to problems of interest to the Laboratory. An overview of the astrophysical simulations performed with the SAGE/RAGE codes will be shown here, including asteroid impacts in the deep-ocean, asteroid impacts on the continental shelf (e.g. – Chicxulub – the dinosaur killer), calculations of massive black holes at the galactic center, and calculations of supernova explosions. Examples of these simulations will be shown.

1. OVERVIEW OF THE CRESTONE PROJECT

The Los Alamos Crestone Project is one of the most successful code-development efforts at Los Alamos National Laboratory. There are many codes in the Crestone Project, both unclassified and classified. In this overview I will discuss the unclassified SAGE and the RAGE codes. The SAGE (SAIC adaptive grid Eulerian) code is a one-, two-, and three-dimensional, multi-material, Eulerian, massively parallel hydrodynamics code for use in solving a variety of high-deformation flow problems. The revolutionary feature of the codes in the Crestone Project is the effective use of continuous adaptive mesh refinement (CAMR) techniques for resolving particular features of the simulation. This CAMR technique allows the code to automatically refine or coarsen the grid cells in order to follow physically interesting features of the hydrodynamic flow. The RAGE CAMR code is built from the SAGE code by adding various radiation packages, improved setup utilities, and graphics packages. It is used for problems in which radiation transport of energy is important. The goal of these massively-parallel versions of the SAGE and RAGE codes is to run extremely large problems in a reasonable amount of calendar time. Our target is scalable performance to ~10,000 processors on a 1 billion CAMR computational cell problem that requires hundreds of variables per cell, multiple physics packages (e.g., radiation and hydrodynamics), and implicit matrix solves for each cycle. A general description of the RAGE code has been published in [1], [2], [3], [4] and [5].

Currently, the largest simulations we do are massively-parallel three-dimensional complex hydrodynamics runs, using around 500 million computation cells and running for literally months of calendar time using ~2000 processors. Current Department of Energy (DoE) Advanced Simulation and Computing (ASC) platforms range from several 3-teraOPS supercomputers to one 12-teraOPS machine at Lawrence Livermore National Laboratory, the White machine, and one 20-teraOPS machine installed at Los Alamos, the Q machine. Additionally, we are exploring the use of large-scale capacity computing clusters (e.g- LINUX based operating systems) as part of the ASC program. Each machine is a system comprised of many component parts that must perform in unity for the successful run of these simulations. Key features of any massively parallel system include the processors, the disks, the interconnection between processors, the operating system, libraries for message passing and parallel I/O, and other fundamental units of the system.

The SAGE and RAGE codes are intended for general applications without tuning of algorithms or parameters. We have run a wide variety of physical applications from millimeter-scale laboratory laser experiments, to the multikilometer-scale asteroid impacts into the Pacific Ocean, to parsec-scale galaxy formation. Examples of these simulations will be shown. The goal of our effort is to avoid ad hoc models and attempt to rely on first-principles physics. In addition to the large effort on developing parallel code physics packages, a substantial effort in the

project is devoted to improving the computer science and software quality engineering (SQE) of the Project codes as well as a sizable effort on the verification and validation (V&V) of the resulting codes.
The Crestone Project team currently is led by three main individuals: Mike Gittings is the chief code architect for the project; Bill Archer is the Code Development Project leader and Bob Weaver is the Applications Requirements leader responsible for physics requirements and initial demonstration applications. The FY2003 budget for the Crestone project was ~$7.2M and the FY2004 budget is ~$8.5M. This project is a significant code development effort at Los Alamos and supports approximately 25 FTEs (full-time-equivalent staff). There are approximately two-three dozen users of the Project codes, both inside and outside the Laboratory. Although three-dozen users seems like a small number, for the kinds of specialized high-performance computing physics used in the Crestone Project codes, this number is actually larger than any other project we know.

2. PROBLEM SETUP FOR COMPLEX 3D GEOMETRIES

Although there are mesh generation tools built into the SAGE code, by far the simplest approach to grid generation is by importing 3D solid model representations of the object being simulated. Mainly through the work of Rob Oakes and his co-workers in X-Division of Los Alamos [6], we have the ability to import nearly any 3D solid model geometry into the code and build a multimaterial mesh based on this geometry. The beauty of this whole process is that once the solid model exists, then the code itself parses the geometry and automatically creates the CAMR grid conforming to the input geometry. The end-user needs only to specify the physical size of resolution required for capturing the appropriate physics in each material. The code does the rest! Examples of this setup are shown in Fig. 1 and Fig. 2.

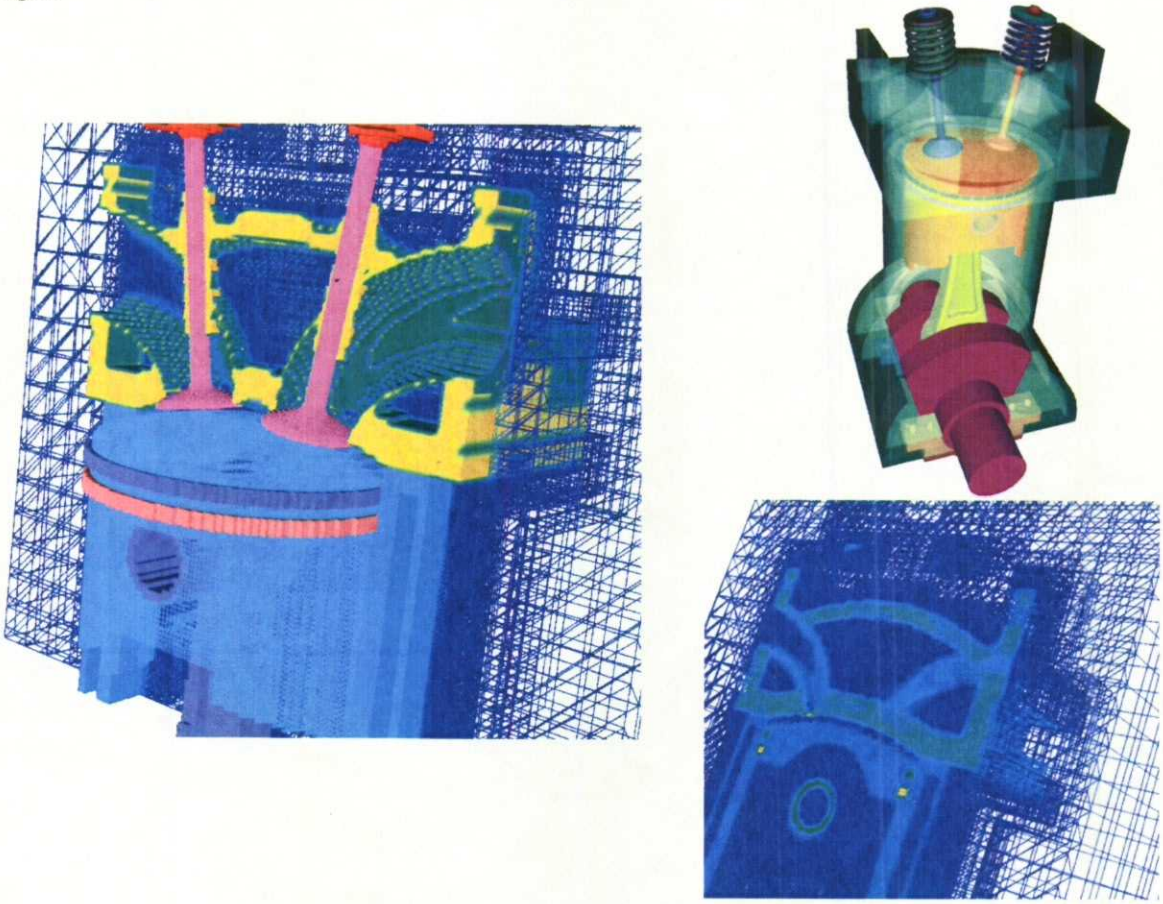

Fig. 1. An example of the use of CAD/CAM 3D solid geometry modeling to generate a continuous adaptive mesh refinement mesh for a portion of a piston engine.

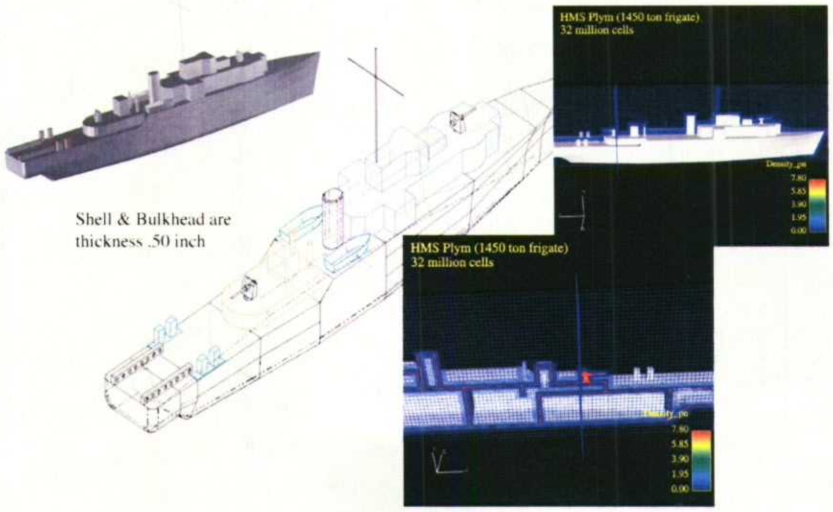

Fig. 2. An example of the use of CAD/CAM 3D solid geometry modeling to generate a continuous adaptive mesh refinement mesh for a portion of a British River-class Frigate.

3. EXAMPLES OF MASSIVELY-PARALLEL SIMULATIONS WITH RAGE

In this section we will show several examples of simulations done with the Crestone project codes. The results will be shown chronologically in order to emphasis the success of the portability and scalability of the software framework. We show an evolution from one of the first generation ASCI machines (the 3 TOps Bluemountain supercomputer), to the 12 TOps White machine at Livermore, and finally to the 20 TOps Q machine at Los Alamos. Our goal for these types of runs is to have a ~1000-fold decrease in wall clock time for 3D runs compared to single processor run-times.

Although some initial 3D runs were performed on the CRAY computers in 1995 and 1996, the real onset of production 3D simulations with the Crestone Project codes started with the delivery of the Bluemountain

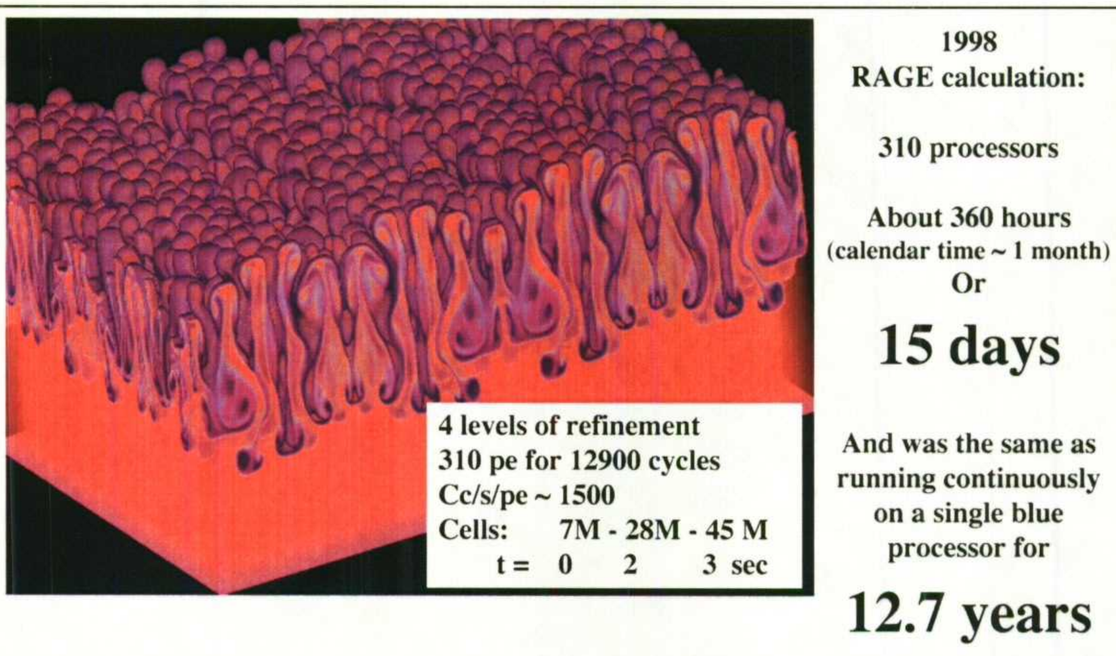

Fig. 3. The first large scale 3D simulation performed with SAGE on the Bluemountain machine: a multimode simulation of Rayleigh-Taylor mixing of two fluids of differing densities in a gravity field.

supercomputer at Los Alamos in 1997. Some of the first runs that we did were related to hydrodynamic mixing of two materials. In 1998, we did a demonstration run on the Bluemountain machine of the 3D Rayleigh-Taylor instability (RTI). A graphic from this run is shown in Fig 3. With only four levels of refinement and 310 processors we were able to finish a simulation in 360 hours of cpu-time or about 15 days of continuous computing. The actual wall-clock time required to complete this run was about one month of calendar time. A comparison we will make as a standard measure of the parallel performance of these runs is the cpu-equivalent time required to run the simulation on a single processor of the same machine. So for this 310 processor run of 360 hours the single processor time would have been ~13 years! The cell-count of this simulation varied with time due to the CAMR because the surface area of the interface between the two mixing fluids grew larger with time. The problem started with about 7 million cells and ended with about 45 million cells, effectively running on only 310 processors.
A dramatic example of the 3D CAMR method is shown in Fig. 4. This Figure shows a 3D simulation of a shock-generated instability (Richtmyer-Meshkov Instability [RMI]) from the passage of a mach 1.2 shock over a perturbed surface of SF_6 in air. The initial perturbation is in the form of a cosine-cosine distribution, and this figure represents a time at which the interface between the air and SF_6 has been shocked from right-to-left and then reshocked from left-to-right in the Figure. Notice the extremely high resolution in the simulation that defines the complex interface

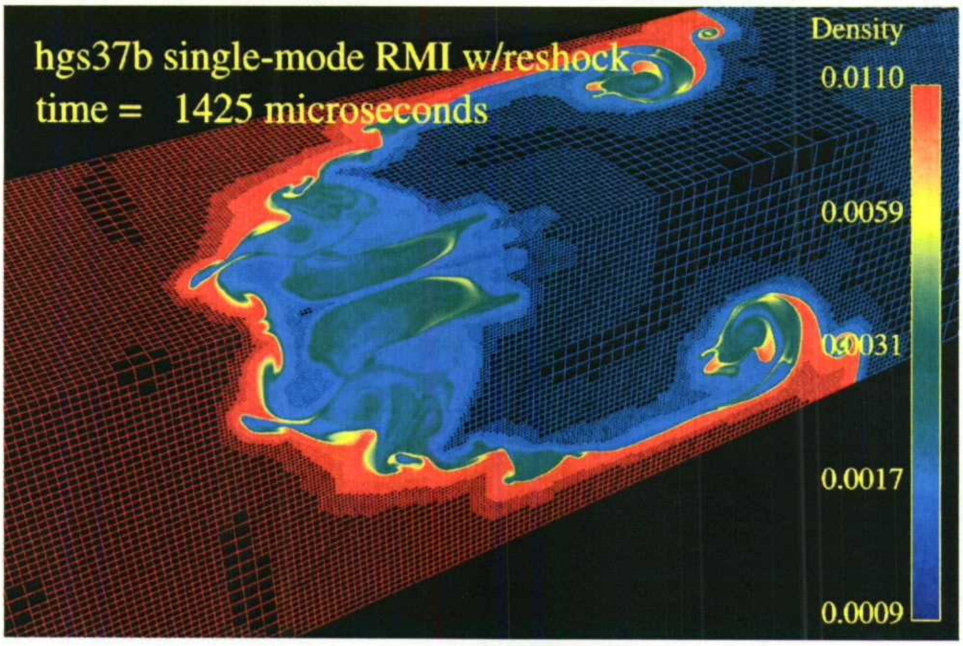

Fig. 4. An example of the 3D CAMR mesh for a simulation of a shock tube experiment involving a perturbed surface of SF_6 gas in air.

between the two materials. These runs were performed in 1999 after the full Bluemountain machine had been delivered to Los Alamos. This full machine of 6144 processors allowed us to perform a parameter study for the same initial conditions on mesh resolution (minimum cell size or maximum number of levels of refinement). In this study we ran the same 3D problem with six, seven, and eight levels of refinement successively. The level six run ran to a problem time of 1.8 ms in 166 cpu hours or 4.8 days of continuous run time. The Level 6, Level 7, Level 8 runs were equivalent to 2 year, 52 year, and 239 year single processor runs (respectively), while they were actually completed in 4.8 days, 34.5 days and 91 days, respectively. So the most refined run, the Level 8 study, was run in 3 months but would have taken over two centuries to complete on a single processor machine assuming, of course that there would be enough memory! These kinds of numbers clearly demonstrate the success of the parallel implementation of the SAGE code and the power of parallel computing. None of these runs would ever have even been started prior to the parallel hardware and parallel software. Within just two years of the start of the ASCI program, the National Laboratory design community was beginning to believe that massively-parallel computing would fundamentally change the scope of simulations that were done.

4. ASTROPHYSICAL APPLICATIONS OF SAGE AND RAGE

In order to apply this massively-parallel radiation-hydrodynamics simulation tool to problems of astrophysical interest, several additional features have been added to the Project codes. The most significant of these features was the addition of a self-gravity solver. This solver allowed the codes to be used for a much broader range of astrophysical problems, including star formation, supernovae implosions and explosions, galactic center research, and cosmology. For more information on this solver, as implemented in the RAGE code, see the work of New, et. al. 2004 [7]. Work has begun on nearly all of these areas of research, and further studies are being developed. Here we will show just a few examples of some initial results from astrophysical simulations. Please see Galen Gisler's paper on "Three-Dimensional Simulations of the Chicxulub Impact Event" at this conference for detailed results on asteroid impacts in both deep water (oceans) and the particular impact that is thought to have killed the dinosaurs: the Chicxulub impact. For completeness here, we will show one figure from Gisler's results – a graphic of the splash after the initial impact of this 10 km stony object. This simulation is shown in Fig. 5.

The other topic for which we have initial results from RAGE is the simulations of massive O-type stars orbiting the Milky Way Galactic Center (GC), where there is a supermassive black hole. This work has been performed by Rob Coker and J. Pittard of the UK, (University of Leeds). The purpose of the RAGE simulation was to look at modeling recent CHANDRA X-ray observations of the GC region. The assumption of this work was that there exists a supermassive blackhole (~3 million solar masses) at the GC. The orbits of the several of the main large stars in the GC region have been measured, and these stars were explicitly put into the simulation. Since these massive orbiting stars have strong stellar winds, the source of the observed X-ray flux could be the interaction of the strong winds. This simulation was a prototype run was very crude zoning in 3D, using about 10 million cells and running for 2 months on 64 Intel cpus. We show movies of the 3D simulation for the meeting, and some sample images are shown here. Figs. 6, 7 and 8.

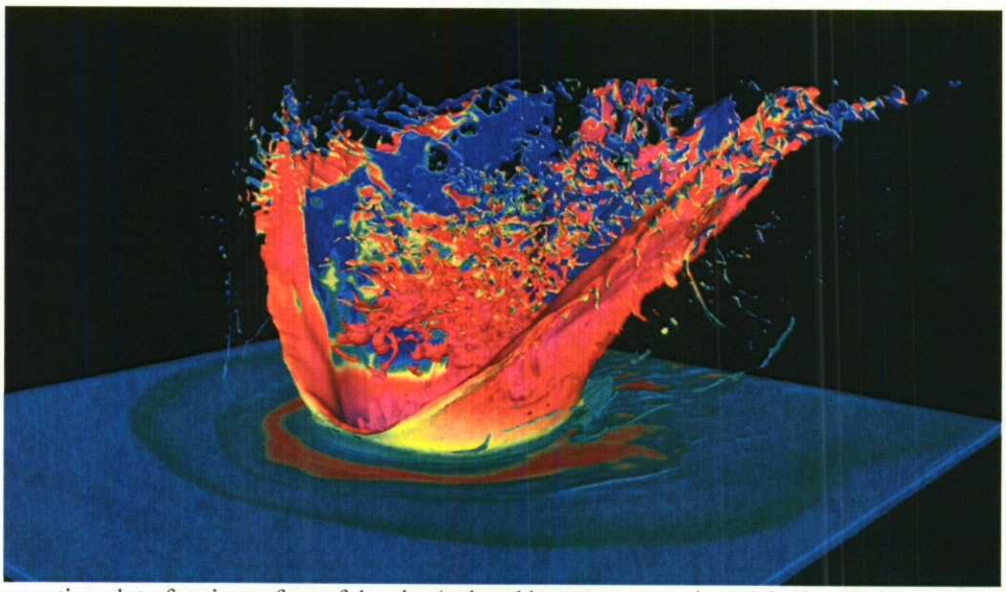

Fig. 5. A perspective plot of an isosurface of density (colored by temperature in eV) in the 45° impact event seconds after the start of the calculation. A hot rooster tail of vaporized water and calcite projects downrange, carrying much of the horizontal component of the projectile's momentum. The debris curtain of proximal ejecta is seen forming around the still expanding crater, and a seismic-thermal pulse propagates outward along the surface of the shallow water.

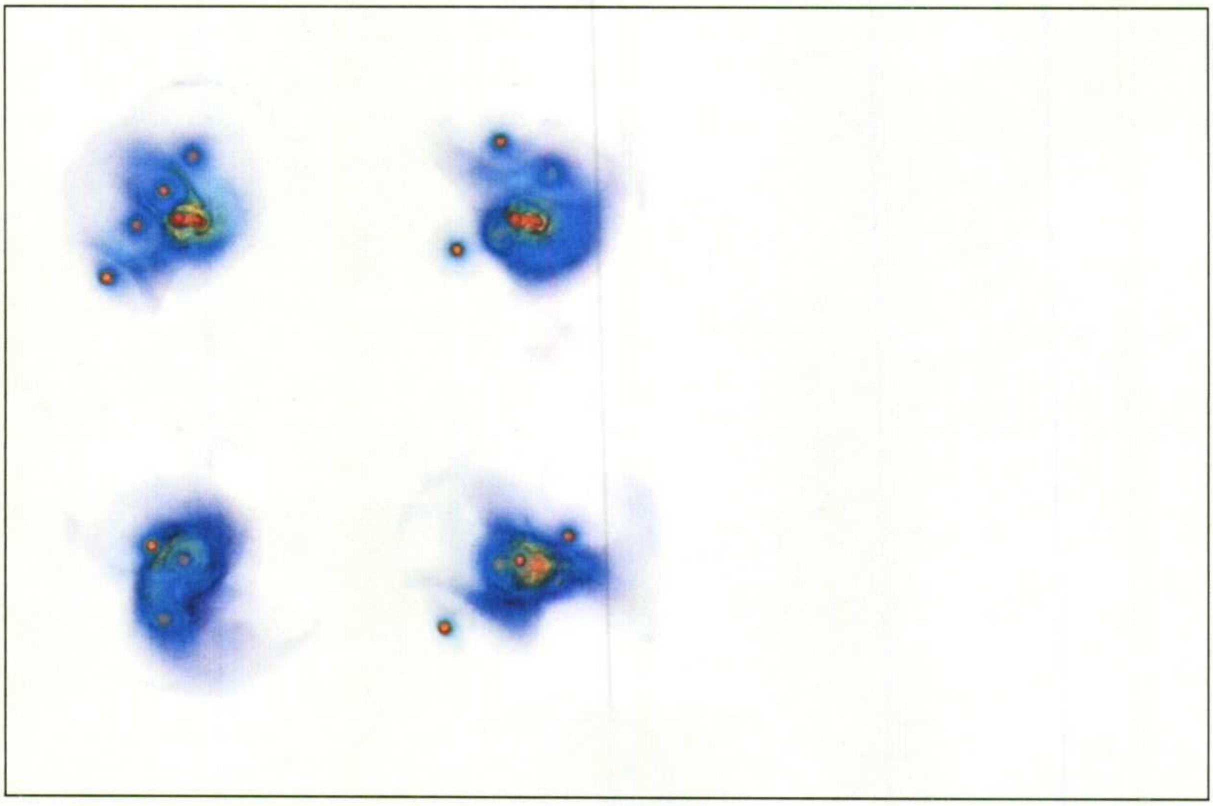

Fig. 6. A sequence of images from a 3D visualization of the density of gas surrounding the GC black hole. The stars are in predetermined orbits around the GC.

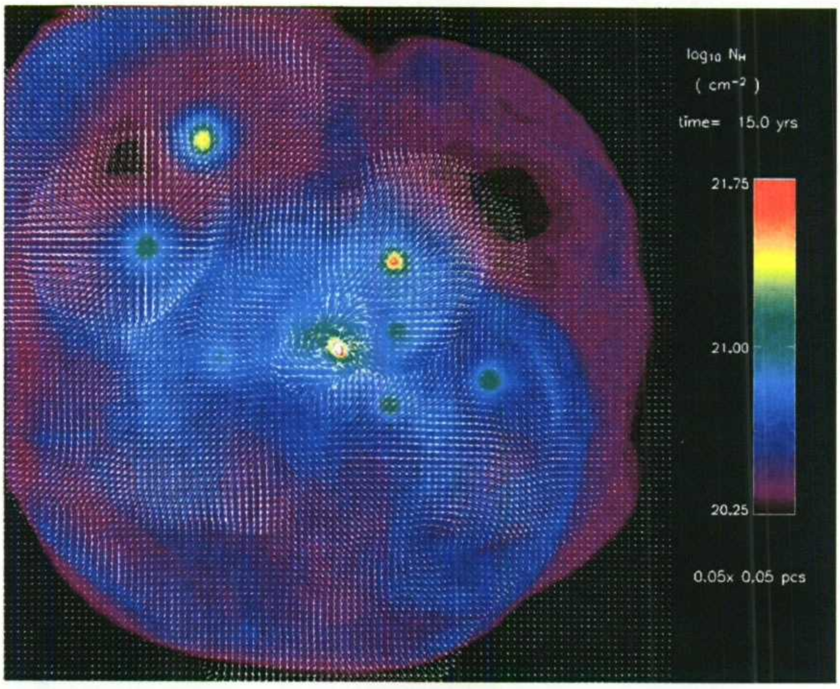

Fig. 7. An image from a 3D visualization of the density of gas surrounding the GC black hole. The stars are in predetermined orbits around the GC. The velocity of the material is shown in this view.

Fig. 8. A true-color synthetic X-ray image (modified to enhance contrast) at 24 years into one simulation. Red is 2-3 keV emission and is likely to be partly absorbed due to the high extinction towards GC. Green is 3-5 keV and blue is 5-10 keV.

5. SUMMARY

We have shown that the SAGE and RAGE codes of the Crestone Project have been very successful products of the DoE's Advanced Simulation and Computing program. It is clear to those performing massively-parallel computations, that the use of thousands of processors in parallel is fundamentally changing the way we think about computer simulations. The Crestone Project codes are fully utilizing each new ASC supercomputer as they become available. The SAGE and RAGE codes are sophisticated Continuous Adaptive Mesh Refinement hydrodynamics codes for large parallel simulations. SAGE and RAGE are becoming useful tools for astrophysical applications. Further research is starting in a wider variety of areas, including cosmological studies with Mike Norman's group at UCSD.

6. REFERENCES

1. "Massively-Parallel Simulations with DOE's ASCI Supercomputers: An Overview of the Los Alamos Crestone Project" R. Weaver and M.L. Gittings 2004, *Conference on Adaptive Mesh Refinement (University of Chicago September 2003)* proceedings to be published by Springer-Verlag 2004.
2. "The SAGE Code Multimaterial Equation of State Methods" M.L. Gittings in *Numerical Methods Symposium*, 28-30 April 1992; "The RAGE Code" R.N. Byrne, T. Betlach, and M.L. Gittings in *Numerical Methods Symposium*, 28-30 April 1992. Copies may be ordered from the Defense Nuclear Agency (currently the Defense Threat Reduction Agency), 56801 Telegraph Road, Alexandria, VA 22310-3398.
3. "2D and 3D simulations of RM instability growth with RAGE: a continuous adaptive mesh refinement code" R. Weaver, M.L. Gittings, R.M. Baltrusaitis, Q. Zhang and S.Sohn, July 1997, proceedings of the *21st International Symposium on Shock Waves (ISSW21)*, Australia, paper 8271.
4. "The parallel implementation of RAGE: a 3D continuous adaptive mesh refinement radiation-hydrodynamics code" R. Weaver, M.L. Gittings, M.L. Clover, July 1999, proceedings of the *22nd International Symposium on Shock Waves (ISSW22)*, London, paper 3560.
5. "The Simulation of Shock-Generated Instabilities" R.M. Baltrusaitis, M.L. Gittings, R. Weaver, R. Benjamin and J. Budzinski 1996, *Physics of Fluids*, 8 (9), p. 2471.
6. "On 3D, Automated, Self-Contained Grid Generation Within the RAGE CAMR Hydrocode" by W.R. Oakes, P.J. Henning, M.L. Gittings, and R.P. Weaver, in proceedings of the *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, p 973, Chateau Whistler Resort, September 2000.

7. "Implementation of a self-gravity solver for RAGE" by K.C.B. New, R.M. Hueckstaedt, R.F. Coker and M.L. Gittings, Monthly Notices of the Royal Astronomical Society, 2004 to be published.