Terascale High-Fidelity Simulations of Turbulent Combustion with Detailed Chemistry

Sponsored by

U.S. Department of Energy, Office of Science, Grants and Contracts Division
Scientific Discovery through Advanced Computing (SciDAC)

Final PSC Technical Report, October 15, 2004

The Pittsburgh Supercomputing Center (PSC) has created and maintains the web site
http://scidac.psc.edu for this project. The downloadable code distribution resulting from PSC's work is found in the Members Area of this web site.

The PSC work on this project consisted of three research and development thrusts:

1. Integration of new modules developed by the university partners into the production S3D code

The physics developments done by the three university participants were integrated into the latest S3D production code that had been jointly produced by Sandia and PSC. The merged code was successfully verified with the test data sets provided by each of the universities, on two different computing platforms.

As a result, the production version of S3D now has the following new capabilities:

o Discrete Transfer Method (DTM) for modeling radiation (U. Maryland)
o Discrete Ordinate Method (DOM) for modeling radiation (U, Michigan)
o Spray module (U. Wisconsin)

These capabilities and their scientific impact are discussed in the resulting research papers linked from scidac.psc.edu:

- Ramanan Sankaran and Hong G. Im (Dept. of Mechanical Engineering, Univ of Wisconsin), Evatt R. Hawkes and Jacqueline H. Chen (Combustion Research Facility, Sandia National Labs), ``The Effects of Nonuniform Temperature Distribution on the Ignition of a Lean Homogeneous Hydrogen-Air Mixture'', submitted for publication to *30th Intl. Symposium on Combustion, Chicago, July 2004.*

- Y. Wang and C.J. Rutland (Dept. of Mechanical Engineering, Univ of Wisconsin), ``Effects of temperature and equivalence ratio on the ignition of n-heptane fuel droplets in turbulent flow'', submitted for publication to *30th Intl. Symposium on Combustion, Chicago, July 2004.*

- Chun Sang Yoo and Hong G. Im (Dept. of Mechanical Engineering, Univ of Michigan), ``Transient Dynamics of Edge Flames in a Laminar Nonpremixed Hydrogen-Air Counterflow'', submitted for publication to *30th Intl. Symposium on Combustion, Chicago, July 2004.*

- Jacqueline H. Chen, Evatt R. Hawkes, and John C. Hewson (Combustion Research Facility, Sandia National Labs), Ramanan Sankaran, and Hong G. Im (Dept. of Mechanical Engineering, Univ of Michigan), Scott D. Mason (Lockheed Martin Corporation) ``Ignition Front Propagation In a Constant Volume With Temperature Inhomogeneities'', submitted for publication to *30th Intl. Symposium on Combustion, Chicago, July 2004.*.

- Yi Wang and Arnaud Trouvé (Dept. of Fire Protection Engineering, Univ of Maryland),``Artificial Acoustic Stiffness Reduction in Fully Compressible, Direct Numerical Simulation of Combustion'', submitted for publication to *Combustion, Theory and Modelling*.

<u>2. Research on incorporating Adaptive Mesh Refinement (AMR) into the S3D code by using the Grid Adaptive Computational Engine (GrACE) framework.</u>

GrACE is a C++ framework for doing Adaptive Mesh Refinement (AMR). Since higher order AMR still faces unresolved research issues such as regridding and clustering criteria and efficiently scalable parallelization, it was agreed among the partners to focus on the technical issues involved in adapting the production S3D code to use GrACE. To preserve the investment put into the S3D code over several years, it was considered necessary to keep as much of the S3D code in Fortran 90 as possible.

To conduct this research, we created an experimental version of S3D:

- A library of Fortran 90 wrappers for GrACE was developed so that GrACE can be called from within the F90 program. Note that the GrACE developers have long planned to provide Fortran-based GrACE interfaces, which could have eliminated the inter-language issues, but these are unavailable to date.

- The Fortran 90 code was restructured to work with "Ghost Zones". A "grid" module was created to implement the mapping of spatial decomposition onto parallel processors in a manner compatible with the requirements of GrACE (as well as those of CCA componentization, see Thrust 3, below).

- The experimental version can be compiled to function identically with the production version, i.e. to **not** use GrACE, by choosing the appropriate DEF option in the Makefile. This allowed us to validate the experimental S3D code against the production code, using the Ignition test case which has been deemed to verify all the critical aspects of the code. The following two plots compare the test results from the experimental version to those of the original version (labeled "james" in the plots) of the code:
http://scidac.psc.edu/plots/ignition-1d-x-press.png
http://scidac.psc.edu/plots/ignition-2d-dev-james.png

To facilitate research on incorporating higher order AMR into S3D, a simplified version of the experimental code (called SS3D) has been created and included in the final code distribution. SS3D uses a code and data structure very similar to the experimental S3D code, but the treatment of chemical reactions has been eliminated and only Euler equations (mass, momentum, and energy) are being solved. SS3D serves as a starting point to test various AMR implementations (via GrACE or other possible frameworks) in a focused manner.

The key issue that must be considered further is this: pointing from the Fortran interface to certain C++ classes (such as GridHierachy or GridFunctions) of GrACE that are dynamically created or destroyed as simulation progresses is highly complex and prone to book-keeping errors. For non-AMR cases, where no regridding and clustering takes place, GridHierarchy and GridFunctions can be pointed correctly at the beginning of program. This means our Fortran wrappers must be made smarter. So far, we have only been able to put the 'level' information in the grid module and

wrappers, but the 'patch' information, which plays an important role in load-balancing, will require further research.

<u>3. Research on Componentizing the S3D code by means of the Common Component Architecture (CCA) framework.</u>

The version to be componentized must be the experimental version of S3D, which can handle AMR via GrACE as discussed above. This poses some unique challenges for componentization. S3D makes extensive use of F90 pointers, which are structurally quite different from the C/C++ pointers on which CCA is based. Thus, we focused on developing a novel technique to solve this problem.

Our solution is being published under the title "A General Approach to Creating FORTRAN Interface for C++ Application Libraries" (see scidac.psc.edu) and was presented at the 2004 International Conference on High Performance Computing and Applications in Shanghai, China.

A general template for creating CCA components from Fortran routines was developed by creating a few basic components. This template makes it easy to create additional components as necessary. This is described in a "How to" document that is included in the code distribution in the Members area of scidac.psc.edu .

Conclusions

Our project has produced a terascale production version of SNL's S3D code that enables high-fidelity simulations of turbulent combustion with detailed chemistry with powerful new physics capabilities. This work has already resulted in a number of scientific publications that report significant progress in combustion theory and modeling.

We have also produced research versions of S3D which provide a platform for investigating the incorporation of higher-order adaptive mesh refinement (AMR) and the adaptation of S3D to use the GrACE framework and to componentization via CCA. In the process, we have developed a new general approach for interfacing F90 code with C++ libraries.