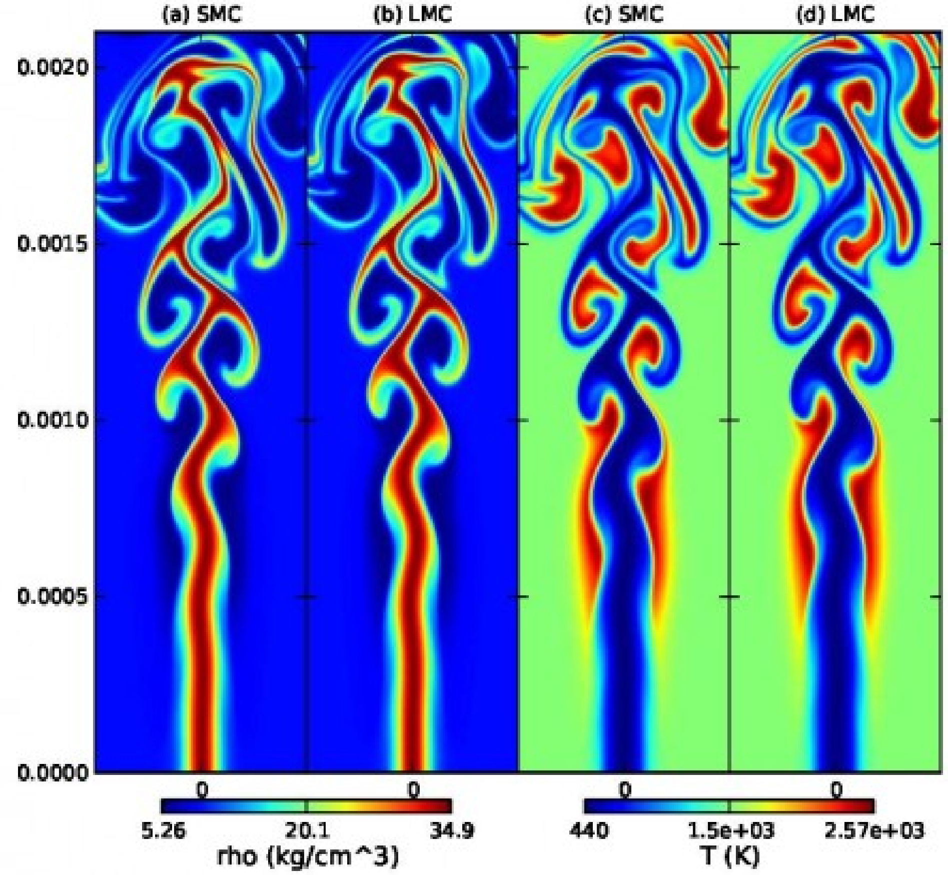
PHYS.ORG

Optimized algorithms help methane flame simulations run 6x faster on NERSC supercomputer

December 19 2014, by Kathy Kincade



This simulation illustrates the total mass density (left) and temperature (right) of a dimethyl ether jet fuel simulation. It is a snapshot of the solution that corresponds to a physical time of 0.00006 seconds. Credit: Matthew Emmett, Weiqun Zhang

Turbulent combustion simulations, which provide input to the design of more fuel-efficient combustion systems, have gotten their own efficiency boost, thanks to researchers from the Computational Research Division (CRD) at Lawrence Berkeley National Laboratory (Berkeley Lab).

Matthew Emmett, Weiqun Zhang and John Bell developed new algorithmic features that streamline turbulent flame simulations, which play an important role in designing more efficient combustion systems. They tested the enhanced code on the Hopper supercomputer at Berkeley Lab's National Energy Research Scientific Computing Center (NERSC) and achieved a dramatic decrease in simulation times.

Their findings appeared in the June 2014 Combustion Theory and Modelling, and some of their simulations were featured on the journal cover.

More than 80 percent of energy consumed in the U.S. occurs via the burning of fossil fuels in transportation systems, heat and stationary power generation systems. This is why research into new fuels and more efficient engine technologies offers enormous potential for savings as well as for pollutant reduction. Shortening the design cycle of new fuels optimally tailored to work with new fuel-efficient engines requires fundamental advances in combustion science.

This is where applied mathematics and supercomputers come in. Modeling and simulation have become integral parts of the combustion design process. A good simulation can inform experimental design to assure the return of high-quality experimental data and enhance the analysis of physical phenomena.

In the supercomputing world, a key component of combustion research is direct numerical simulation (DNS). The combustion community has experimented with various DNS approaches for more than 25 years, the study authors noted. But the advent of manycore computer architectures is prompting computational scientists to update DNS codes to ensure that they can run efficiently on the next generation systems.

"The thing that we as applied mathematicians want to get across is that the systems you are running these codes on are changing, and the code you are running now is not going to necessarily be efficient on future machines," said Emmett, a postdoc in the CRD. "And to get it to be efficient isn't just a matter of getting somebody to optimize the code—you need new algorithms."

Time-stepping Strategies

One of the standard computational tools used in DNS studies of combustion is the numerical integration of the reacting, compressible Navier-Stokes equations. Although these types of DNS codes have been successful, they suffer from some inherent weaknesses, according to Emmett. For one thing, the chemical equations function at a different time scale than the flow equations, which impacts how efficiently the code will run. In addition, manycore architectures make it more expensive to access memory, particularly when trying to communicate across the network to access data stored somewhere else.

"What we are seeing is that floating point operations, or FLOPS, are becoming relatively cheap while network operations and memory accesses are becoming more expensive," he said. "So if you rewrite your algorithm to do more floating point operations, which are cheap, and do less communication, which is expensive, then you should get a faster code that is more efficient because you're doing the expensive part less frequently. And this is new. Ten years ago it was the other way around. Now you want to save on memory access and increase your FLOPS."

To address these issues, Emmett and Zhang developed a hybrid OpenMP/MPI parallel DNS code called SMC designed to simulate turbulent combustion on next-generation architectures. In particular, it can more robustly handle cases where the time scales of the advection, diffusion and chemical processes are considerably different, Emmett explained.

Two new algorithmic features of the code are a narrow stencil finite-difference algorithm and a multi-rate time-stepping strategy. Implementing a narrow stencil rather than the traditional wide stencil increases the number of floating point operations but reduces the amount of communication between cores, while the multi-rate integrator enables the chemistry to be advanced on a different time scale than the fluid dynamics.

"The big improvement here is the multi-rate integration and the way we separate the advection-diffusion processes and the chemistry process," Emmett said. "In traditional codes these things are locked together and you have to march them forward at the same time, which has consequences for how efficient your code can be. But we separate them to make the code run more efficiently. And even though the two processes have been separated, overall a very tight coupling is maintained between them. This is one of the improvements over previous DNS codes."

For example, a methane flame simulation run with the multi-rate integrator ran six times faster on Hopper compared to the same code with the multi-rate integrator turned off. And in a series of dimethyl ether (DME) jet simulations designed to demonstrate how the SMC code could be used to study complex new fuels, the multi-rate integrator was able to operate with a larger time-step than single-rate integrators and thus obtained an accurate solution in less time.

"The DME jet is an example of a difficult problem that is essentially infeasible, in the context of DNS codes, without using advanced algorithms such as the multi-rate integration scheme in SMC," Emmett said. "Some of the chemical reactions in a DME flame happen very quickly and restrict the size of the numerical time-step that some codes can take. With a code that doesn't have a multi-rate integrator, trying to run the DME jet simulation would take a very, very long time."

Enabling the Next Generation

The new algorithms in SMC are also designed to scale up to the manycore architectures that will power the next generation of supercomputers, such as Cori, a Cray XC scheduled to go online at NERSC in 2016. In fact, this research was done in conjunction with the Center for Exascale Simulation of Combustion in Turbulence, a collaborative effort comprising six national labs (Berkeley, Sandia, Oak Ridge, Lawrence Livermore, Los Alamos and NREL) and five university partners (The University of Texas at Austin, Stanford, Georgia Institute of Technology, The University of Utah and Rutgers). Members of CRD have also been chosen to participate in the new NERSC Exascale Science Applications Program to further optimize their codes for Cori.

In the meantime, they continue to work closely with combustion scientists. "We like to have real applications where we are teaming up with domain scientists who are driving the applications and then match the algorithms and codes we are writing to do real science," Emmett said. "Evolving computer architectures constrain the design of simulation software; the role of mathematics is to develop efficient algorithms that respect those constraints."

More information: "High-order algorithms for compressible reacting flow with complex chemistry." *Combustion Theory and Modelling* Volume 18, Issue 3, 2014 DOI: 10.1080/13647830.2014.919410

Provided by Lawrence Berkeley National Laboratory

Citation: Optimized algorithms help methane flame simulations run 6x faster on NERSC supercomputer (2014, December 19) retrieved 27 April 2024 from https://phys.org/news/2014-12-optimized-algorithms-methane-flame-simulations.html