COMPUTATION

2016 ANNUAL REPORT

PRODUCTION TEAM

LLNL Associate Director for Computation
Bruce Hendrickson

Deputy Associate Directors
James Brase, John Grosh, and Terri Quinn

Scientific Editors
Mark Pettit and Brian Gallagher

Art Director
Acen Datuin

Production Editor
Deanna Willis

Writers
Holly Auten, Allan Chen, Rose Hansen, Arnie Heller, Karen Kline, Don McNichols, Caryn Meissner, Ann Parker, and Deanna Willis

Proofreader
Holly Auten

Photographer
Acen Datuin

Print Production
Diana Horne and Monarch Print Copy and Design Solutions

U.S. Government Printing Office: 2016/670-187-52060
LLNL-TR-729102

ACCELERATING SIMULATION SOFTWARE WITH GRAPHICS PROCESSING UNITS

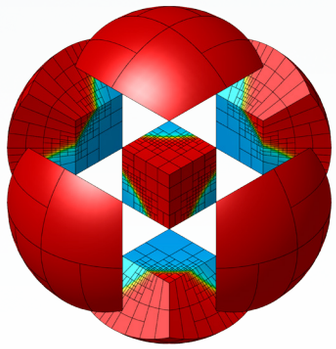
Livermore scientists are redesigning simulation software to leverage the capabilities of next-generation exascale computing.

To address the challenges of transitioning to the next generation of high performance computing (HPC), Livermore is bringing together designers of hardware, software, and applications to rethink and redesign their HPC elements and interactions for the exascale era (i.e., systems capable of a billion billion floating point operations per second or 10^{18} flops). This collaboration process is called co-design and is essential for next-generation supercomputing, where simulation codes will have a billion-way parallelism distributed among many processor cores and accelerator cards.

A key example of the work Livermore's Computation Directorate is doing to prepare for this transition to exascale is its effort to accelerate MFEM (Modular Finite Element Methods), an open-source, scalable software library for high-order finite element methods. Originally funded by Livermore's Laboratory Directed Research and Development program, MFEM plays a foundational role in simulation and modeling. MFEM allows its users to convert real-world physics models into a discrete computational representation based on finite elements (meshes of squares, cubes, triangles, or tetrahedrons with approximate fields defined on them) in a process called discretization. Application codes can use MFEM's finite element representation as the framework to build simulations of physical phenomena. The Department of Energy's Exascale Computing Project recently funded a co-design Center for Efficient Exascale Discretizations (CEED) at Livermore to co-develop algorithms and libraries (e.g., MFEM and Argonne National Laboratory's Nek5000) for the efficient discretization and solution of physical models on the upcoming exascale machines.

HPC-based modeling used to be limited solely by the speed of computation, but now, computer scientists are concerned with moving data around fast enough to be available in time for the calculations, as well as minimizing energy use. One way to achieve high performance on the new types of architectures is to use higher-order numerical algorithms. "Higher-order methods give you results faster, or better results in the same amount of time," says Tzanio Kolev, an applied mathematician who leads CEED and the MFEM project. Computational scientists prefer these methods because they provide the flexibility to adjust their scaling properties depending on the type of hardware, such as by changing the ratio of calculations to memory accesses (flops/byte). "But if you're not careful, you can easily get higher complexity with higher order, leading to prohibitively expensive algorithms," he adds.

The MFEM team is working to reduce the complicated finite element calculations to a series of dense tensor contractions (essentially higher dimension analogues of matrix–matrix multiplication from linear algebra). This new approach, known as partial assembly, was developed initially by Veselin Dobrev, a computational mathematician on the team, and has shown great promise on traditional CPUs (central processing units). "Graphics processing units [GPUs] are a recent development in HPC that are very well suited to handling the dense blocks of numbers that arise in tensor contractions," says Aaron Fisher, a computational scientist working on MFEM. GPUs working in concert with CPUs are an essential element of the next-generation of high performance computers on the road to exascale computing. The Sierra system, the first of these systems slated for Livermore, will arrive in 2018.



The MFEM logo demonstrates the high-order geometry and field representation that the library uses to describe real-world phenomena in a discrete form suitable for processing in high performance computer simulations.

TZANIO KOLEV
kolev1@llnl.gov

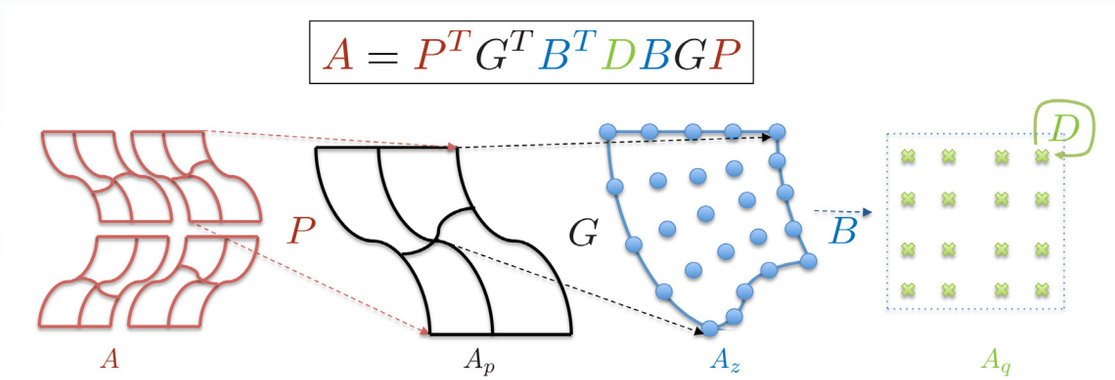
AARON FISHER
fisher47@llnl.gov

VESELIN DOBREV
dobrev1@llnl.gov

The MFEM team is developing a three-pronged approach to prepare the library for Sierra and the ever faster systems that will follow it. One of the efforts is building on a library called OCCA, which allows developers to write a single code that can be executed on both CPUs and GPUs, and supports just-in-time (JIT) compilation. This JIT approach helps the computer's compiler (the software that converts programs into machine code for execution by the computer) build a more efficient code by utilizing information that is only known when the program is running. Collaborators at the University of Tennessee, Knoxville, are working on a second approach building on MAGMA, a library of LAPACK-type dense linear algebra routines designed to run on large HPC systems with both CPUs and GPUs.

The third approach is built on an in-house library called AcroTensor that handles dense tensor contractions in a user-friendly manner. With AcroTensor, the user can write code that looks nearly identical to the mathematical formula for a tensor contraction, making it easier for the programmer to translate a problem-solving approach into software routines. AcroTensor parses the mathematical formula, translates it to efficient code for the GPU, and JIT compiles it. AcroTensor allows various teams at LLNL to prototype quickly the specific contractions needed in their applications, and gets good performance on the GPU. The AcroTensor research is funded by Livermore's Institutional Center of Excellence project.

The work to improve MFEM's speed has many benefits. As computing transitions to the exascale level, developers will need to alter codes to run optimally on the new machines. Accelerated MFEM running on exascale machines will allow developers to build on an efficient discretization foundation so they can focus on modelling challenging physics phenomena such as heart mechanics, high-energy-density interactions, fusion simulations, and subsurface flow. Other projects and applications at Livermore and throughout the HPC world benefit as well, thanks to the availability of new tools for simulation and basic science discovery. "MFEM is a software library that can power a wide variety of large-scale simulations. By improving the library, many applications benefit. Collaborating with scientists at Livermore and the broader scientific community uplifts our work and everybody else's," says Kolev.



MFEM can represent finite element operators, generated from complex partial differential equations, by decomposing them into parallel, mesh, and geometric/physical components. These components are represented as linear operators, the innermost of which are the "tensor contractions" that can be evaluated in a highly efficient manner on high performance parallel supercomputers.