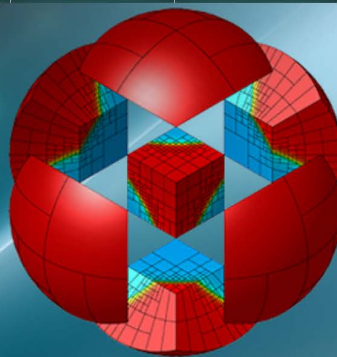


MFEM is an open-source, lightweight, general, and scalable C++ library for high-order 2D and 3D finite element methods



About Computing

Our People

Organization

News & Press

Hackathons

Multimedia Gallery

Contact

[Home](#) > [About](#) > [News & Press](#)

NEWS

High-Order Finite Element Library Provides Scientists with Access to Cutting-Edge Algorithms

Friday, January 6, 2016

A team at the Center for Applied Scientific Computing is continuing to develop MFEM, an open-source, lightweight, general, and scalable C++ library for high-order 2D and 3D finite element methods with conforming or nonconforming adaptive mesh refinement. Conceptually, MFEM can be viewed as a finite element toolbox that provides the building blocks for developing finite element algorithms in a manner similar to that of MATLAB for linear algebra methods.

The finite element method is a powerful discretization technique that uses general unstructured grids to approximate the solutions of many partial differential equations. High-order finite elements in particular are ideally suited to take advantage of the changing computational landscape because their order can be used to adjust the algorithm for different hardware or to fine tune the performance by increasing the FLOPs/bytes ratio.

The [MFEM project](#) performs math research and software development, aiming to provide application scientists with access to cutting-edge algorithms for high-order finite element meshing, discretizations, and linear solvers. MFEM's meshes and solutions can be visualized with its lightweight native visualization tool GLVis, as well as with the VisIt visualization and analysis tool developed at LLNL.

"MFEM enables computational mathematicians to quickly and easily develop and test new research algorithms in very general, fully unstructured, high-order, parallel settings," says project leader [Tzaniu Kolev](#). "At the same time, MFEM enables application scientists who have developed their applications on top of MFEM to focus on the physics without worrying about discretization details, and provides them with access to cutting-edge math algorithms."

MFEM began as a student project in 2000, and the team started developing it seriously in 2010 with Laboratory Directed Research and Development (LDRD) funding. Today, MFEM receives funding from [Advanced Simulation and Computing](#), the Sierra Institutional Center of Excellence, the DOE Office of Science, and Information System Contingency Plan programs.

MFEM is used in a variety of applications, including the [BLAST](#) and Atratus codes at LLNL, where it acts as a discretization engine for high-order Arbitrary Lagrangian-Eulerian (ALE) shock hydrodynamics and high-order radiation-diffusion, respectively. This separation of physics (as in BLAST) and discretization (as in MFEM) is not common, but "has been very beneficial over the years," says Kolev. MFEM has also been applied successfully to a variety of other problems, including linear and non-linear elasticity, reaction-diffusion problems, time-domain electromagnetics, DG advection problems, and Stokes/Darcy flow.

The algorithms in BLAST are now being picked up by Weapons and Complex Integration as part of the Multi-Physics on Advanced Platforms project. MARBL is a new multi-physics application code for simulating high-energy-density physics and focused experiments driven by high-explosive, magnetic, or laser-based energy sources. It is based on combining the high-order finite element ALE capabilities of BLAST with the high-order finite difference Eulerian capabilities of Miranda to increase calculation flexibility. MFEM, as BLAST's foundation and discretization engine, will be a critical component of this code effort. As such, the MFEM team will continue to improve and develop it in the future.

Share



Recent News

- [High-Order Finite Element Library Provides Scientists with Access to Cutting-Edge Algorithms](#)
- [LLNL's de Supinski earns prestigious IEEE Fellowship](#)
- [NeurIPS papers aim to improve understanding and robustness of machine learning algorithms](#)
- [Girls Who Code - 'Big' program goes virtual](#)
- [LLNL, IBM win SC20 'Test of Time' for Blue Gene/L](#)

[more](#)

Computing

[Home](#)
[About](#)
[Research](#)
[News](#)
[History](#)
[Careers](#)
[Contact](#)
[Credits](#)

LLNL Mission Areas

[Bio-Security](#)
[Counterterrorism](#)
[Defense](#)
[Energy](#)
[Intelligence](#)
[Nonproliferation](#)
[Science & Technology](#)
[Weapons & Complex Integration](#)

About LLNL

[What We Do](#)
[How We Do It](#)
[Our Values](#)
[Organization](#)
[Publications](#)
[History](#)

LLNL

[Home](#)
[Missions](#)
[Programs](#)
[Doing Business](#)
[News](#)
[Careers](#)
[Community](#)

Social Media

