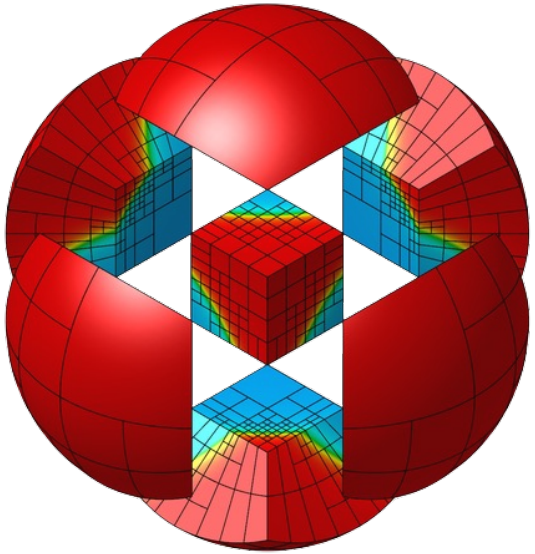
MFEM: Recent Developments



MFEM Workshop 2022

October 25, 2022, Virtual Meeting

Veselin Dobrev and the MFEM team

LLNL-PRES-842999

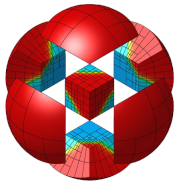
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

CASC

Lawrence Livermore National Laboratory

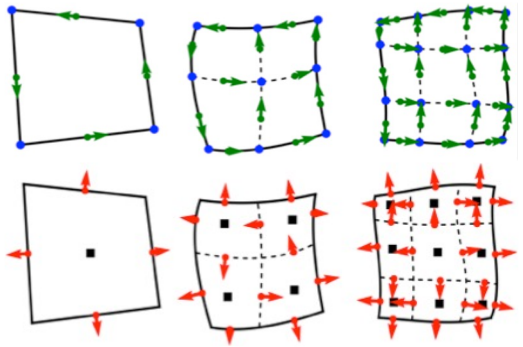
MFEM: Modular Finite Element Methods library

MFEM is open-source C++ library for scalable FE research and fast application prototyping

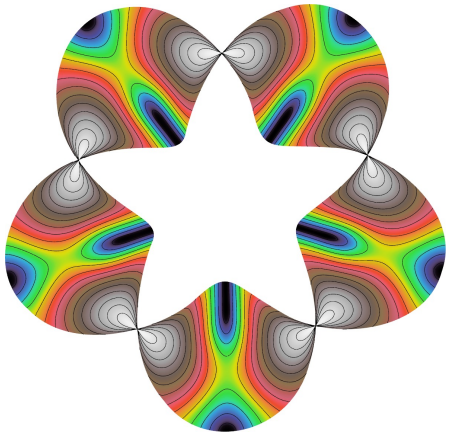


mfem.org

- Triangular, quadrilateral, tetrahedral, prismatic, hexahedral, and pyramidal; volume, surface and topologically periodic meshes
- Arbitrary order curvilinear mesh elements
- Arbitrary order H^1, H(curl), H(div) and L^2 elements
- Local conforming and non-conforming refinement
- NURBS geometries and discretizations
- Bilinear and linear forms for variety of methods (Galerkin, DG, DPG, IGA, …)
- Sparse matrices, smoothers, Krylov solvers, eigensolvers
- Scalable assembly and linear solvers through hypre
- Non-linear operators and non-linear solvers
- Explicit and implicit high-order time integration
- Example codes & Miniapps – simple proxies for high-order discretizations of various physics
- Integration with hypre, PETSc, SUNDIALS, STRUMPACK, and more

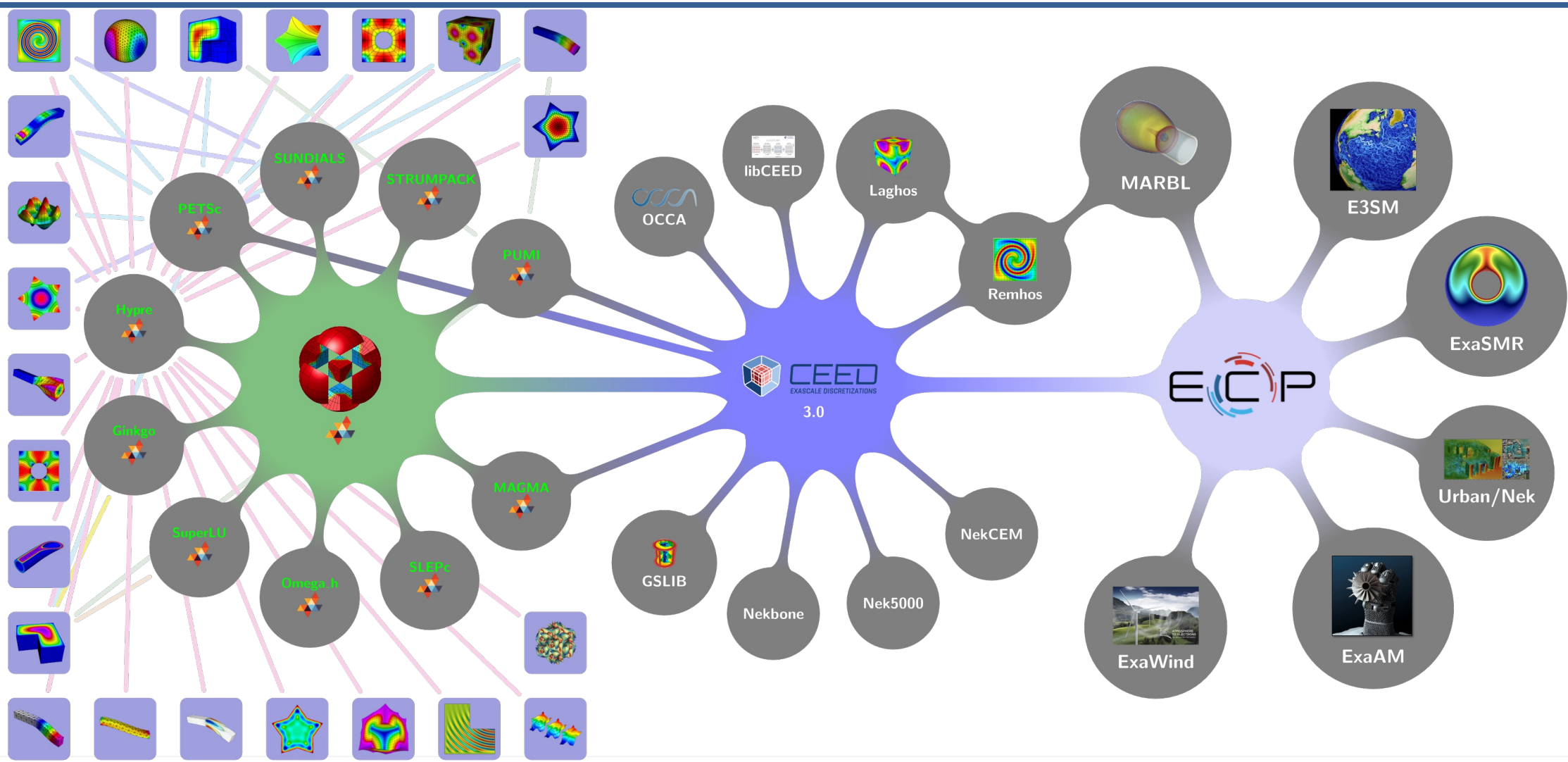


Linear, quadratic and cubic finite element spaces on curved meshes



Maxwell eigenmode on a Mobius strip, computed with LOBPCG+AMS

MFEM connections and integrations



MFEM connections to other projects

- Part of the Extreme-scale Scientific Software Development Kit, xSDK: xsdk.info
- Part of the FASTMath institute: fastmath-scidac.llnl.gov
- Engaged in SciDAC, e.g. RF-SciDAC and TDS-SciDAC

New developments in MFEM

- Sub-mesh extraction and field transfer functionality
- Functionality to reconstruct a parallel mesh on a single rank
- More 3D TMOP metrics, specialized metrics for untangling, worst-case quality
- Support for mixed meshes and pyramids in GSLIB-FindPoints
- Linear form assembly on GPUs
- Support for coefficient evaluation on GPUs
- Improved LOR solvers and fast DG mass inverse (see Will Pazner's talk)

Highlight: sub-mesh extraction

- SubMesh and ParSubMesh inherit from Mesh and ParMesh
- Sub-meshes can be subsets of volume or boundary elements
- Sub-meshes are defined through volume or boundary attributes
- Field transfers (of GridFunctions) can be performed in all cases:
  - Parent mesh to child mesh (sub-mesh)
  - Child mesh to parent mesh
  - Between any two meshes that share a common ancestor (parent, parent of parent, etc)
- Goal is to serve as the basis for multi-domain problems

Highlight: linear form assembly on GPUs

- Support can be enabled with LinearForm::UseFastAssembly(true)
- The following integrators are supported for now:
  - DomainLFIntegrator
  - VectorDomainLFIntegrator
  - DomainLFGradIntegrator
  - VectorDomainLFGradIntegrator
- Efficient coefficient evaluation is essential for performance

Highlight: coefficient evaluation on GPUs

- GPU evaluation is facilitated by the new methods Coefficient::Project, VectorCoefficient::Project, etc.
- These new Project methods can be overridden in derived user classes
- Internally, the evaluation is unified via the class CoefficientVector
- Fast code path for scalar, vector, matrix constant coefficients
- Fast code path for scalar, vector, matrix GridFunction and QuadratureFunction coefficients
- The base class (generic) implementation is on CPU using the legacy Eval methods

New developments in MFEM (cont.)

- New patch recovery-based (Zienkiewicz-Zhu) a posteriori error estimator
- Added more flexibility to the constrained solver classes
- New nonlinear elasticity miniapp, Hooke, using AD (built-in or Enzyme)
- New miniapp for body-fitted volumetric and shape integration with the Algoim library
- New example 33/33p – solution of spectral fractional PDEs
- Support for the Enzyme AD tool
- Support for the ParMoonolith library

Highlight: Hooke miniapp

- Static nonlinear elasticity solver using “matrix-free” algorithms
- Customizable elasticity models; provided are linear and Neo-Hookean models
- Can use AD to evaluate gradients of the model
- Implements a simple built-in AD
- When configured, it can use the (external) Enzyme AD tool

New developments in MFEM (cont.)

- GPU support for mixed meshes using libCEED
- New benchmark for the different assembly levels; inspired by the CEED Bake-Off Problems
- Added Windows 2022 CI testing on GitHub

CASC

Center for Applied
Scientific Computing

Lawrence Livermore
National Laboratory