The State of MFEM

MFEM Community Workshop
October 20, 2021

Tzanio Kolev
LLNL

LLNL-PRES-828128

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

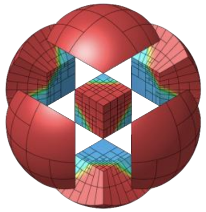
mfem

Lawrence Livermore National Laboratory

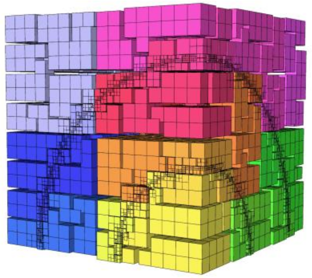
MFEM

Cutting-edge algorithms for powerful applications on HPC architectures

- **Flexible discretizations on unstructured grids**
  - Triangular, quadrilateral, tetrahedral and hexahedral meshes.
  - Local conforming and non-conforming AMR, mesh optimization.
  - Bilinear/linear forms for variety of methods: Galerkin, DG, DPG, …
- **High-order and scalable**
  - Arbitrary-order H1, H(curl), H(div)- and L2 elements.
  - Arbitrary order curvilinear meshes.
  - MPI scalable to millions of cores and GPU-accelerated.
  - Enables application development from laptops to exascale machines.
- **Built-in solvers and visualization**
  - Integrated with: HYPRE, SUNDIALS, PETSc, SLEPc, SUPERLU, …
  - AMG preconditioners for full de Rham complex, geometric MG
  - Support for GPU solvers from: HYPRE, PETSc, AmgX
  - Accurate and flexible visualization with VisIt, ParaView and GLVis
- **Open source**
  - Available on GitHub under BSD license. 75+ example codes and miniapps.
  - Part of FASTMath, ECP/CEED, xSDK, OpenHPC, E4S, …

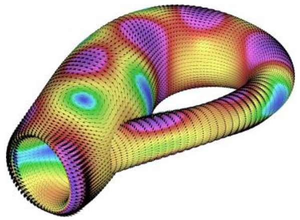


High-order curved elements

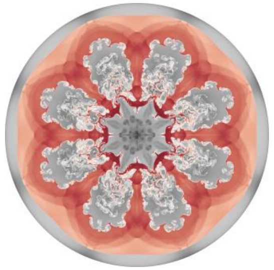


Parallel non-conforming AMR

ECP ASC CEED EXASCALE DISCRETIZATIONS FASTMATH xSDK



Surface meshes



Compressible flow ALE simulations

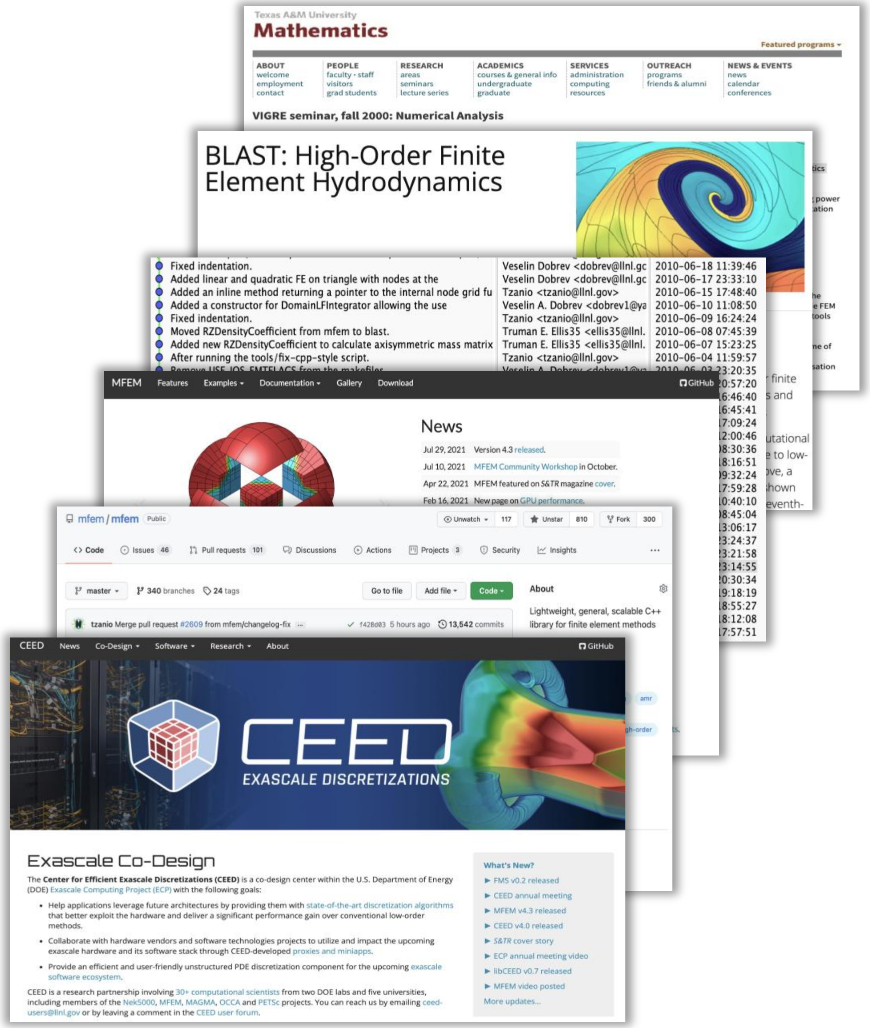
2020 R&D 100 FINALIST

Heart modeling

A Brief History

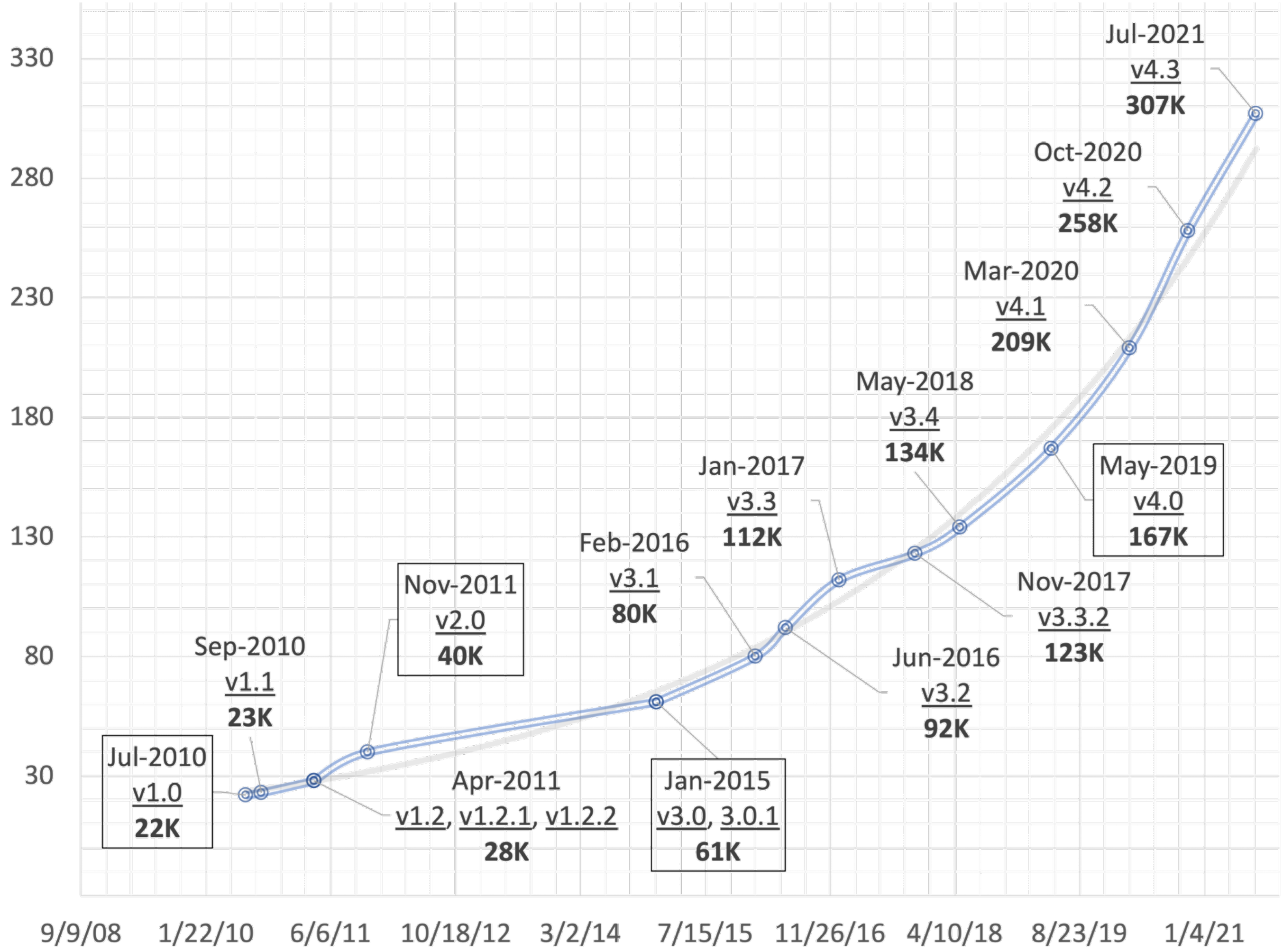
We’ve been doing this for a long time

- **2000 – “VIGRE seminar: Numerical Analysis,” Texas A&M University**
  - Research code: AggieFEM/aFEM
  - Some of the original contributors: **@v-dobrev**, **@tzanio**, **@stomov**
  - Used in summer internships at LLNL
- **2010 – BLAST project at LLNL**
  - Motivated high-order, non-conforming AMR and parallel scalability developments
  - MFEM repository starts in May 2010
  - Some of the original contributors: **@v-dobrev**, **@tzanio**, **@rieben1**, **@trumanellis**
  - Project website **mfem.org** goes live in August 2015
- **2017 – Development moved to GitHub**
  - First GitHub commits in February 2017
  - Team expands to include many new developers at LLNL and externally
- **2017 – CEED project in the ECP**
  - Motivated partial assembly, GPU, and exascale computing developments



The Source Code Has Grown Significantly

SLOC in MFEM releases over the last 11 years



mfem-4.3.tgz	v4.3	Jul 2021	2.8M	307K	
mfem-4.2.tgz	v4.2	Oct 2020	2.4M	258K	
mfem-4.1.tgz	v4.1	Mar 2020	7.9M	209K	
mfem-4.0.tgz	v4.0	May 2019	5.2M	167K	*GPU support*
mfem-3.4.tgz	v3.4	May 2018	4.4M	134K	
mfem-3.3.2.tgz	v3.3.2	Nov 2017	4.2M	123K	*mesh optimization*
mfem-3.3.tgz	v3.3	Jan 2017	4.0M	112K	
mfem-3.2.tgz	v3.2	Jun 2016	3.3M	92K	*dynamic AMR, HPC miniapps*
mfem-3.1.tgz	v3.1	Feb 2016	2.9M	80K	*fem ↔ linear system interface*
mfem-3.0.1.tgz	v3.0.1	Jan 2015	1.1M	61K	
mfem-3.0.tgz	v3.0	Jan 2015	1.1M	61K	*non-conforming AMR*
mfem-2.0.tgz	v2.0	Nov 2011	308K	40K	*arbitrary order spaces, NURBS*
mfem-v1.2.2.tgz	v1.2.2	Apr 2011	240K	28K	
mfem-v1.2.1.tgz	v1.2.1	Apr 2011	240K	28K	
mfem-v1.2.tgz	v1.2	Apr 2011	240K	28K	*MPI parallelism based on hypre*
mfem-v1.1.tgz	v1.1	Sep 2010	166K	23K	
mfem-v1.0.tgz	v1.0	Jul 2010	160K	22K	*initial release*

The Community Has Grown Significantly

GitHub, downloads, and workshop stats

GitHub

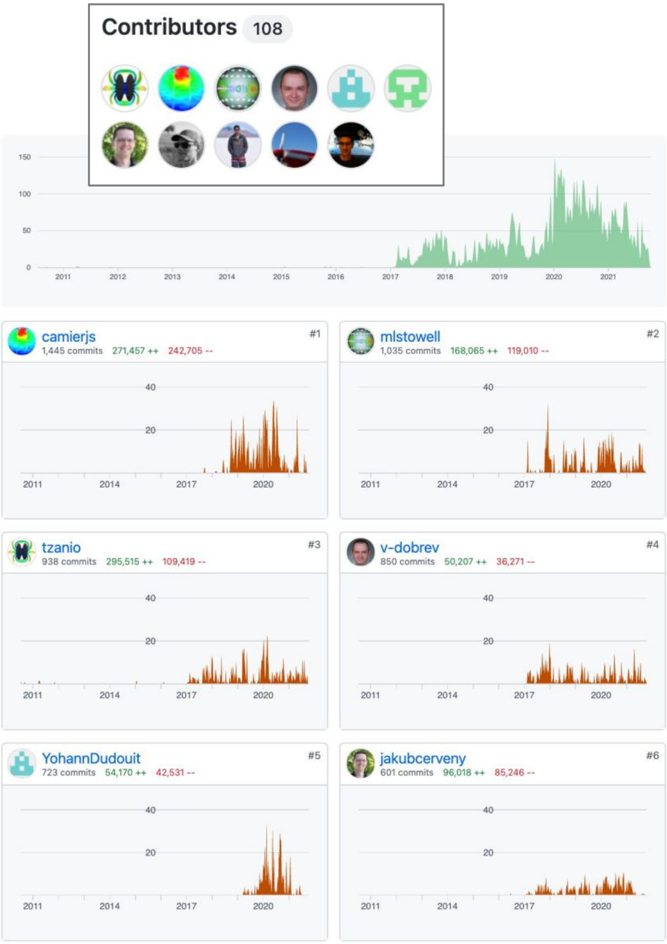
- **108** contributors
- **100** commits / week
- **456** people in the mfem organization – *join to contribute* + *receive announcements*
- **100** visitors / day
- **810** stars – *thank you!*

Downloads

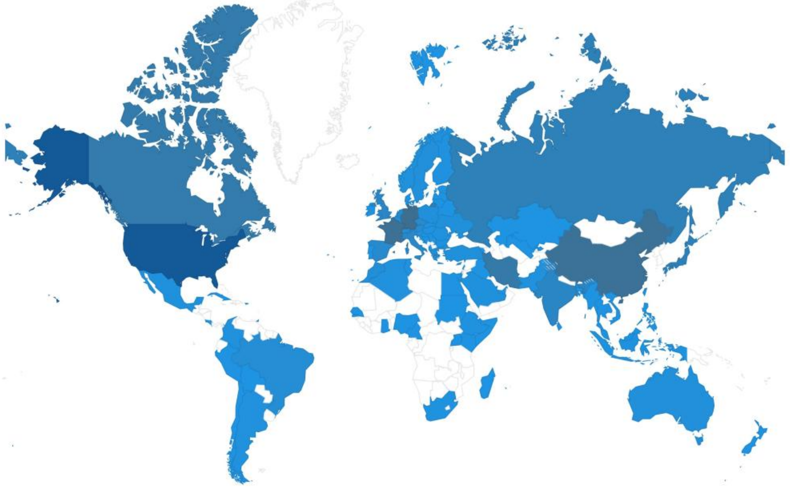
- **35** downloads + clones / day · **12K** / year
- **102** countries total

2021 Community Workshop

- **238** researchers
- **120** organizations
- **28** countries



Top contributors as of Oct 2021



MFEM has been downloaded from 102 countries

mfem.org — **MFEM Community Workshop** — *October 2021*

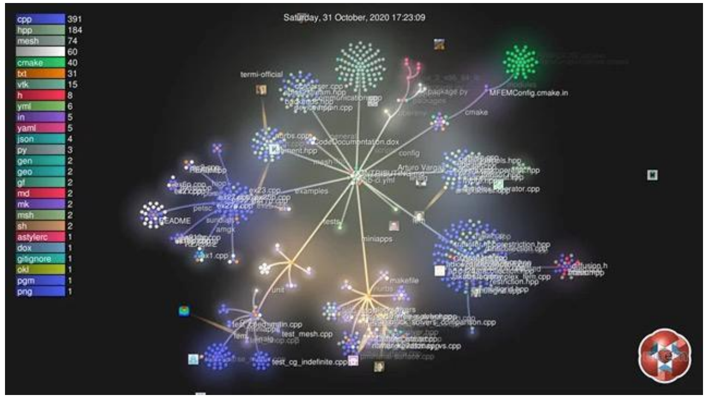
001	Aaron Fisher	*Lawrence Livermore National Laboratory*	fisher47@llnl.gov
002	Abdelrahman Elmeliegy	*North Carolina State University*	aaelmeli@ncsu.edu
003	Abhilash Reddy Malipeddi	*University of Michigan*	abhilash@gwu.edu
004	Abhishek Verma	*Applied Materials Inc.*	AbhishekKumar_Verma@amat.com
005	Aditya Joshi	*Rensselaer Polytechnic Institute*	joshia5@rpi.edu
006	Adolfo Rodriguez	*OpenSim Technology LLC*	adolfo@opensim.technology
007	Adriano Cortes	*Federal University of Rio de Janeiro*	adriano@nacad.ufrj.br
008	Adrien Lefieux	*Covanos*	adrien.lefieux@gmail.com
009	Aidan Hamilton	*University of Delaware*	aidan@udel.edu
010	Ajay Rawat		ajay.rawat83@gmail.com
011	Alberto Gascón	*University of Granada*	albergascon@gmail.com
012	Alejandro Campos	*Lawrence Livermore National Laboratory*	campos33@llnl.gov
013	Alejandro Muñoz Manterola	*University of Granada*	alevale71@gmail.com

2021 Community workshop had 238 registrations

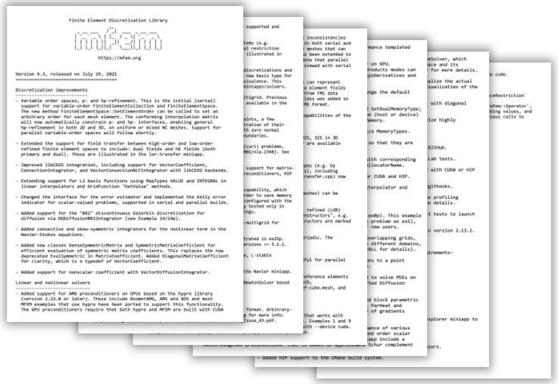
Latest Releases Was a Team Effort

Version 4.3 stats

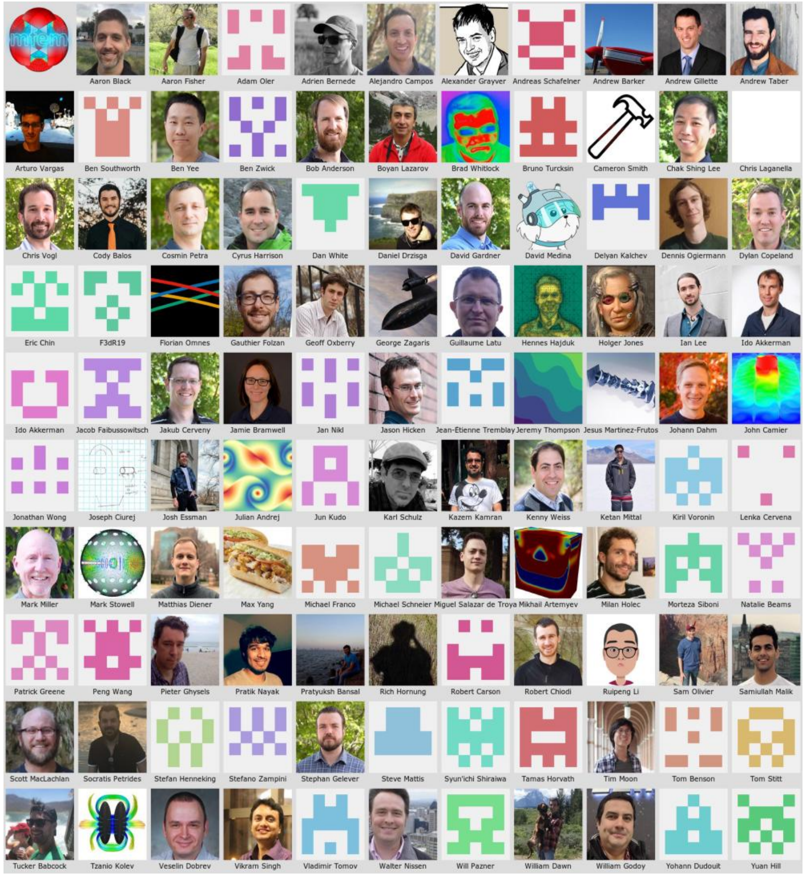
- Released **July 29, 2021**
- **9** months in development
- **81** contributors
- **218** PRs merged
- **265** issues closed
- **48862** new lines of code
- **3806** number of commits
- **Many new features:**
  - GPU solvers from hypre + PETSc
  - LOR discretizations, hp-refinement
  - GPU-powered mesh optimization
  - FMS, Caliper, Ginkgo, VTK support
  - 11 new examples + miniapps



The making of mfem-4.3
youtu.be/3Fc1nxQJUVw



The mfem-4.3 CHANGELOG has 60+ entries



MFEM contributors on GitHub

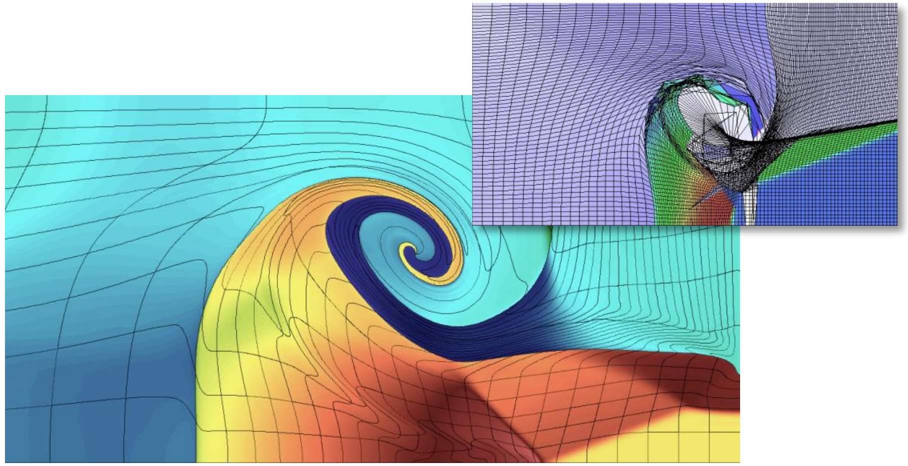
High-Order Methods for Large-Scale Multi-Physics

- **Large-scale parallel multi-physics simulations**
  - Radiation diffusion
  - Electromagnetic diffusion
  - Compressible hydrodynamics
- **Finite elements naturally connect different physics**

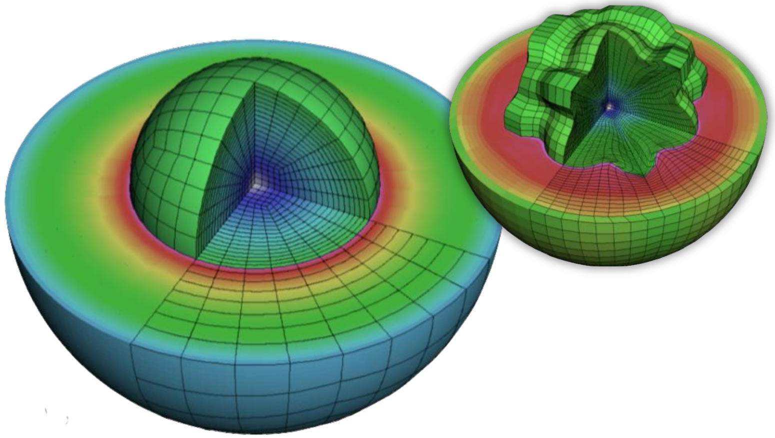
$$H(grad) \xrightarrow{\nabla} H(curl) \xrightarrow{\nabla\times} H(div) \xrightarrow{\nabla\cdot} L_2$$

$H(grad)$	$H(curl)$	$H(div)$	L_2
"nodes"	"edges"	"faces"	"elems"
High-order kinematics	High-order MHD	High-order rad. diffusion	High-order thermodynamics

- **High-order finite elements on high-order meshes**
  - Increased accuracy for smooth problems
  - Sub-element modeling for problems with shocks
  - HPC utilization, FLOPs/bytes increase with the order
- **Need new (interesting!) R&D for full benefits**
  - Meshing, discretizations, solvers, AMR, UQ, visualization, …



Robustness: 8th order Lagrangian shock triple-point (BLAST) vs. classical low-order method (SGH)

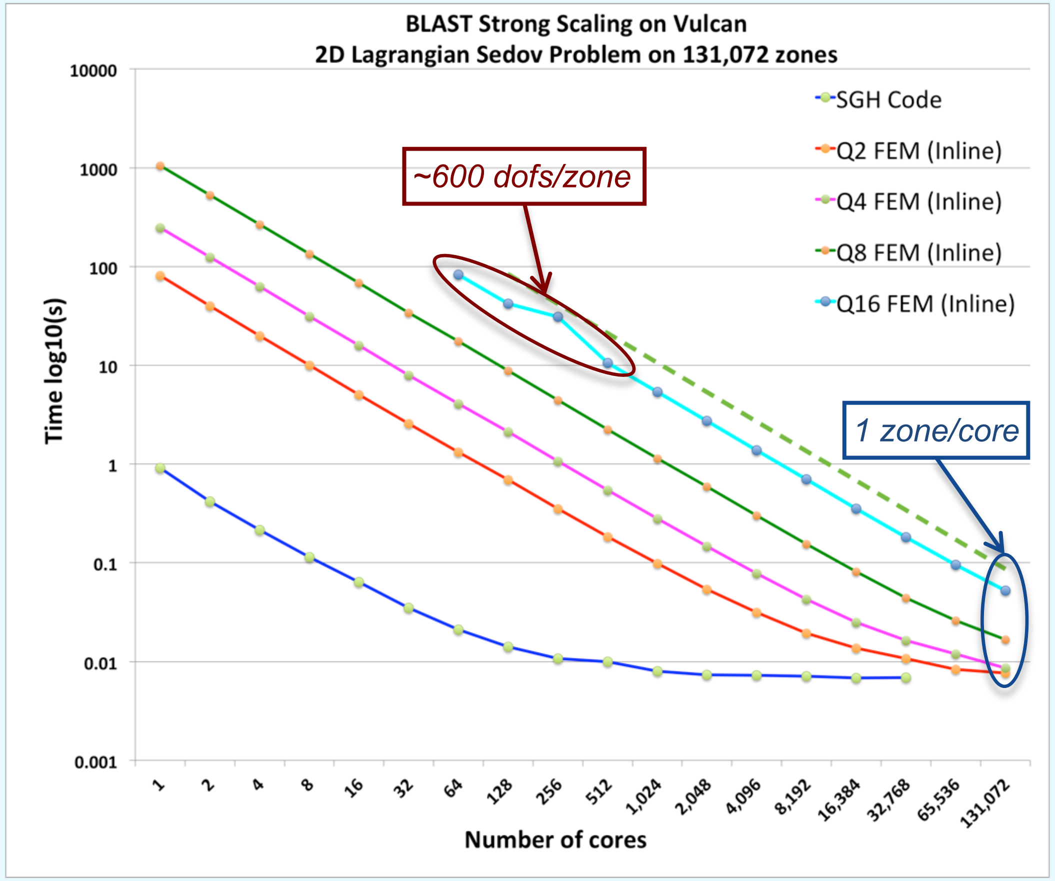


Symmetry: 2nd order Lagrangian ICF-like implosion (BLAST) vs. classical low-order method (SGH)

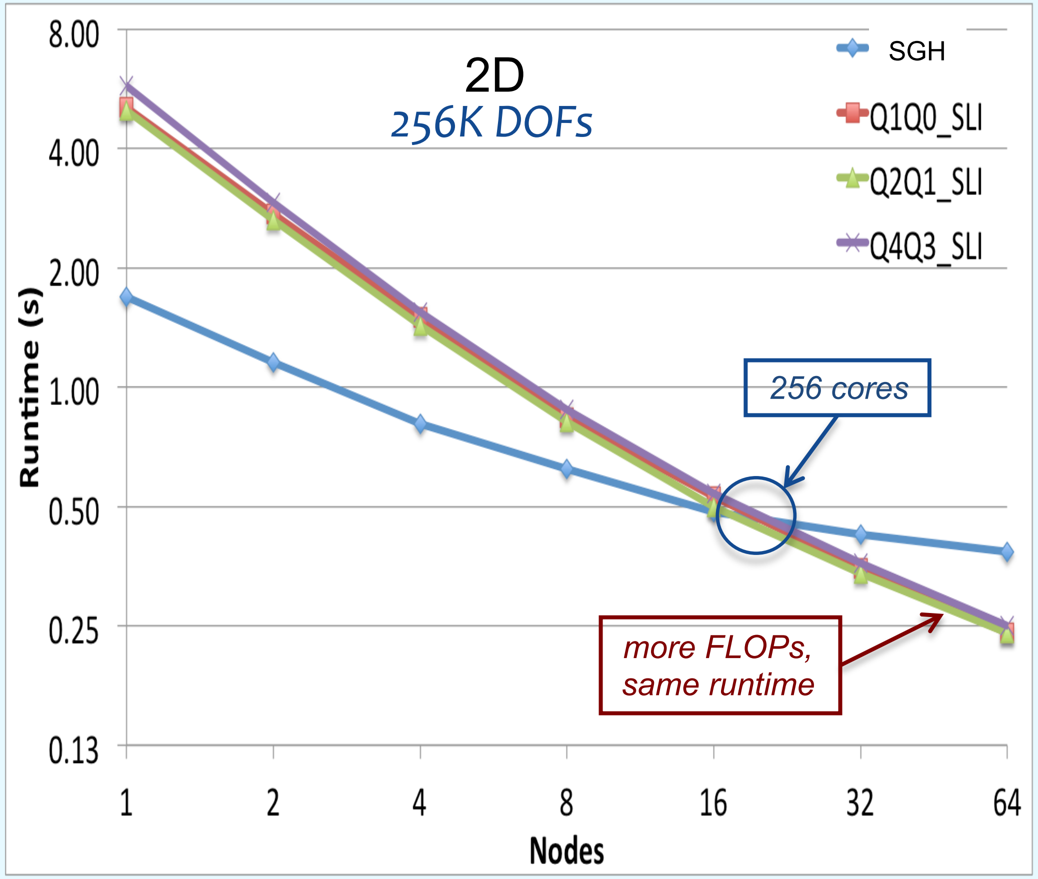
High-Order Methods for Large-Scale Multi-Physics

Parallel scalability

Strong scaling, p-refinement, PA



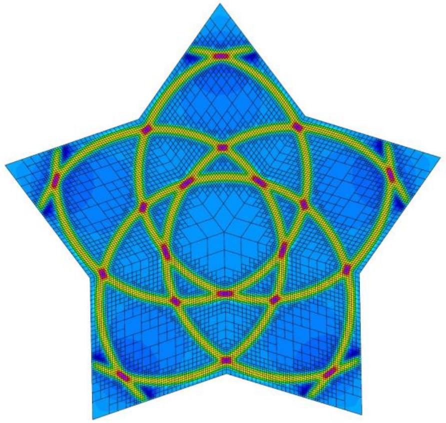
Strong scaling, fixed #dofs, PA+SLI



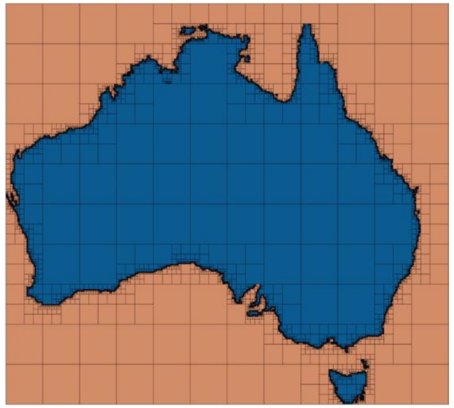
Adaptive Mesh Refinement

MFEM’s unstructured AMR infrastructure

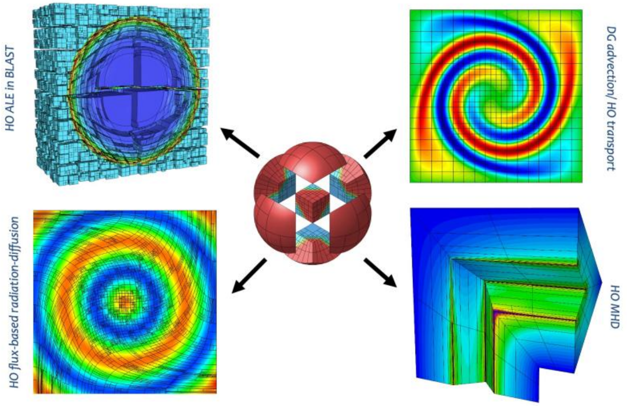
- **AMR on library level**
  - Conforming local refinement on simplex meshes
  - Non-conforming refinement for quad/hex meshes
  - Initial hp-refinement
- **General approach**
  - Any high-order finite element space, H1, H(curl), H(div), on any high-order curved mesh
  - 2D and 3D · hexes, prisms, tets
  - Arbitrary order hanging nodes
  - Anisotropic refinement
  - Derefinement
  - Serial and parallel, including parallel load balancing
  - Independent of the physics
  - Easy to incorporate in applications



Example 15



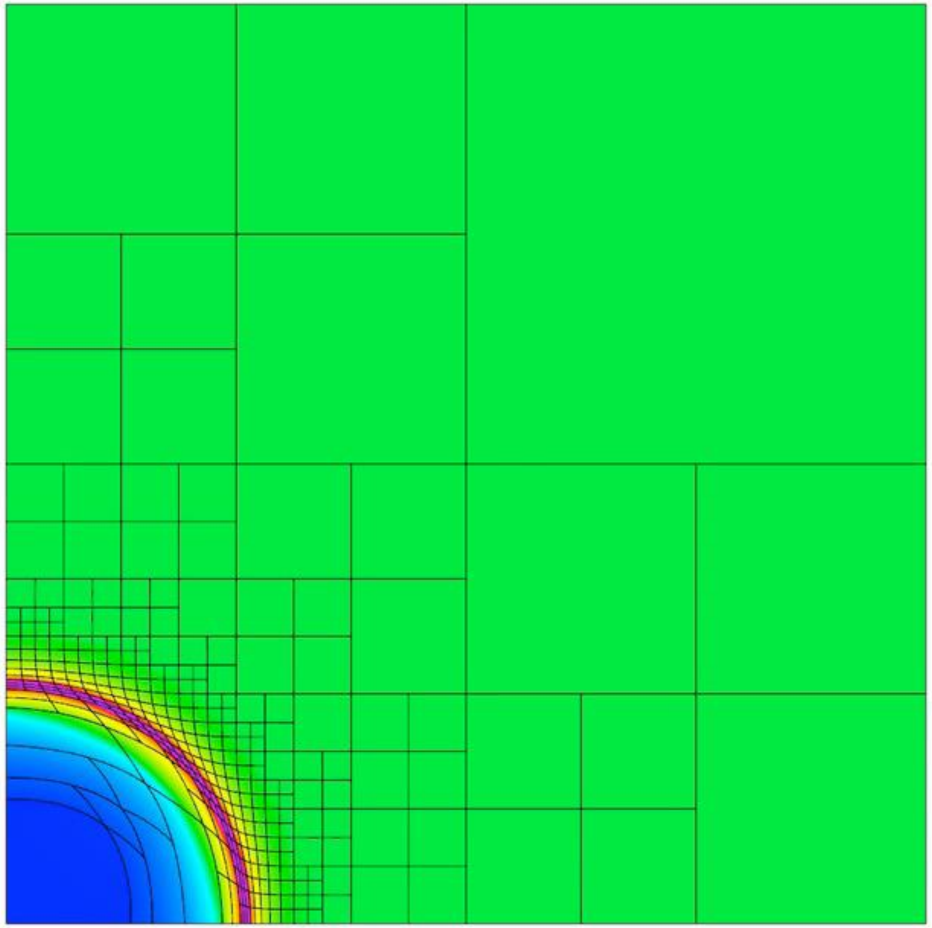
Shaper miniapp



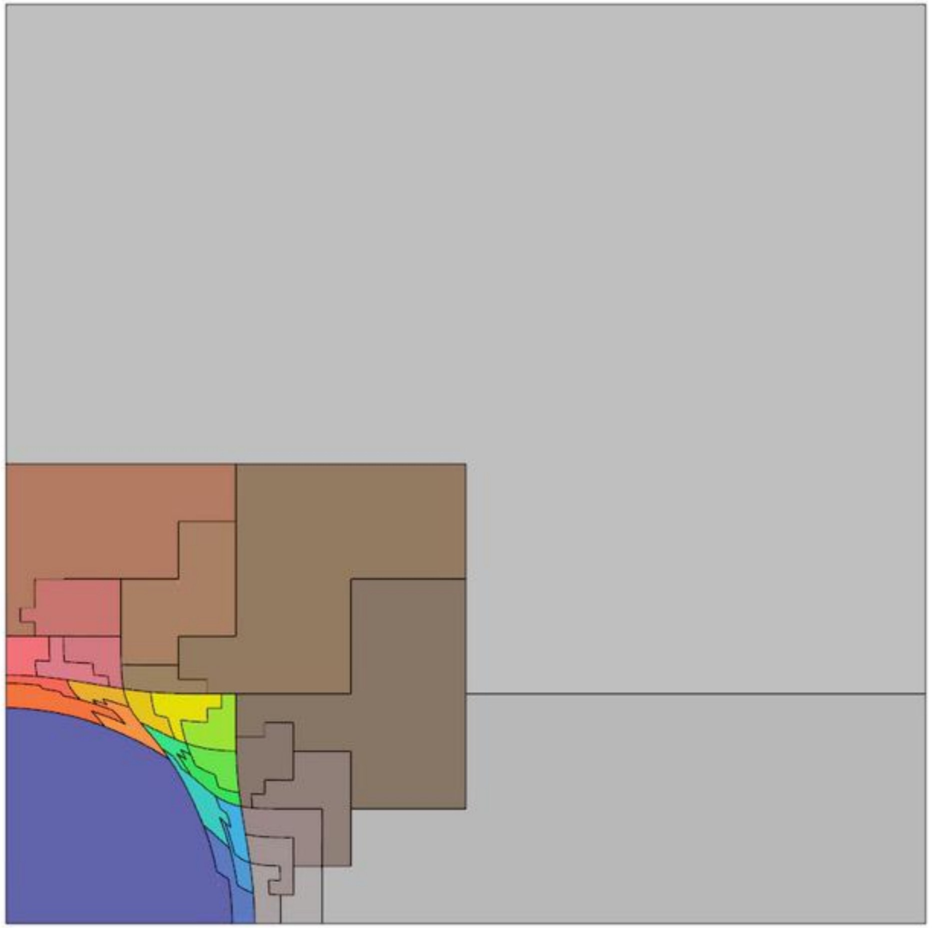
Same AMR algorithms can be applied to a variety of high-order physics

Adaptive Mesh Refinement

Parallel dynamic AMR, 2nd order Lagrangian Sedov problem



Adaptive, viscosity-based refinement and derefinement

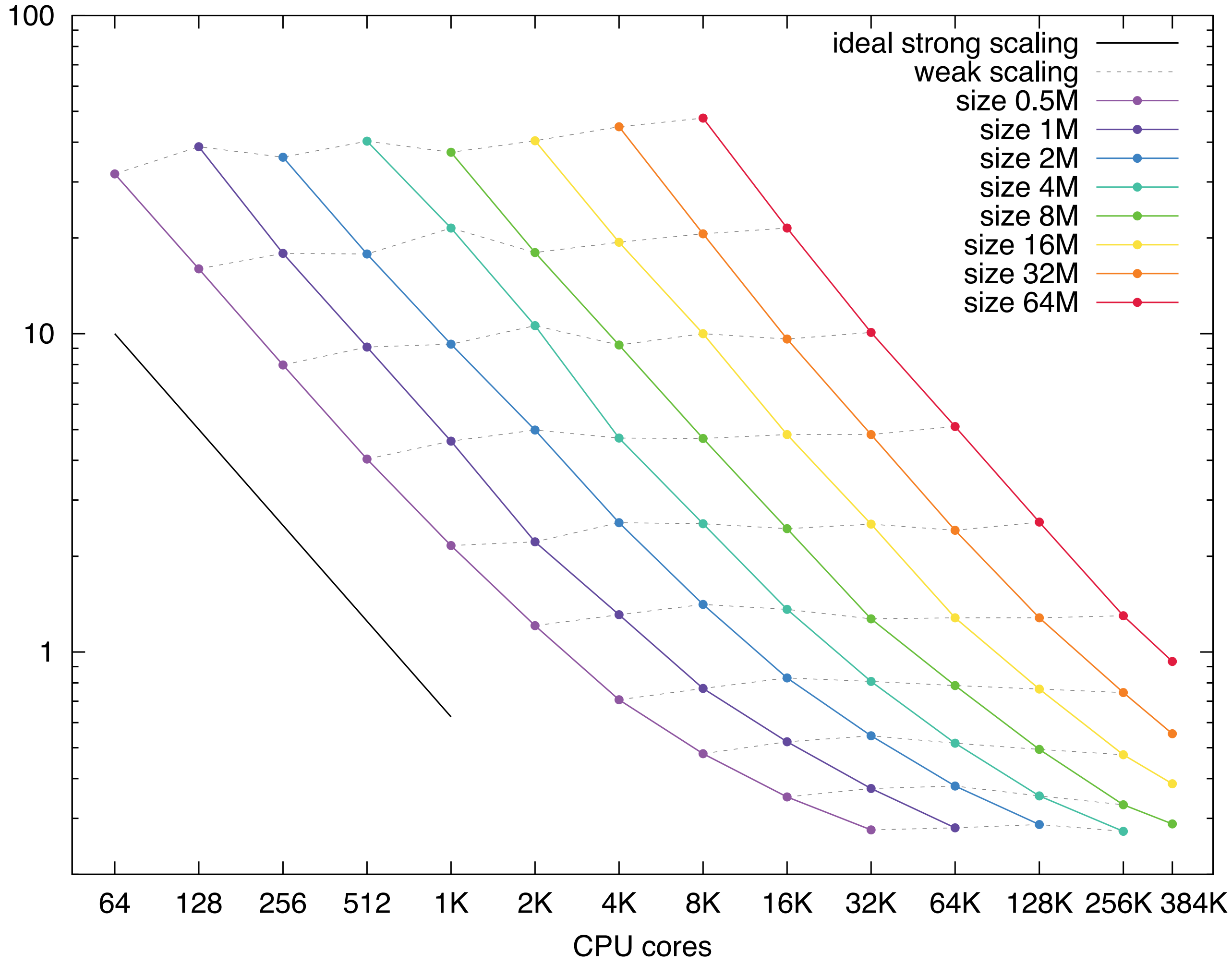


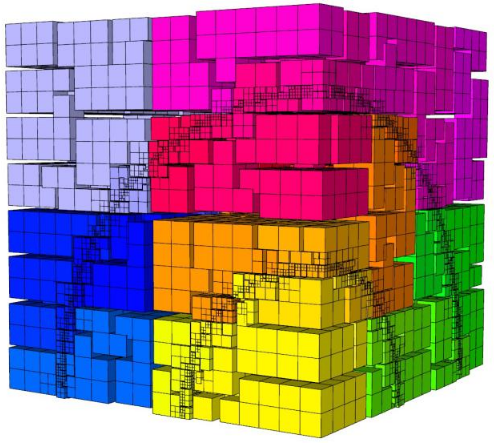
Parallel load balancing, 16 cores

Adaptive Mesh Refinement

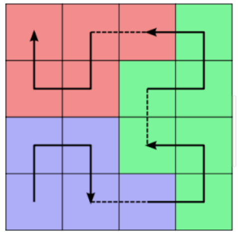
Parallel AMR scaling to ~400K MPI tasks

- **Weak + strong scaling**
  - ~400K MPI tasks
  - BG/Q
- **Measure only AMR components**
  - Interpolation matrix
  - Assembly
  - Marking
  - Refinement
  - Rebalancing
  - No linear solves
  - No “physics”
- **Documented in 2019 SISC paper**





Parallel decomposition (2048 domains shown)

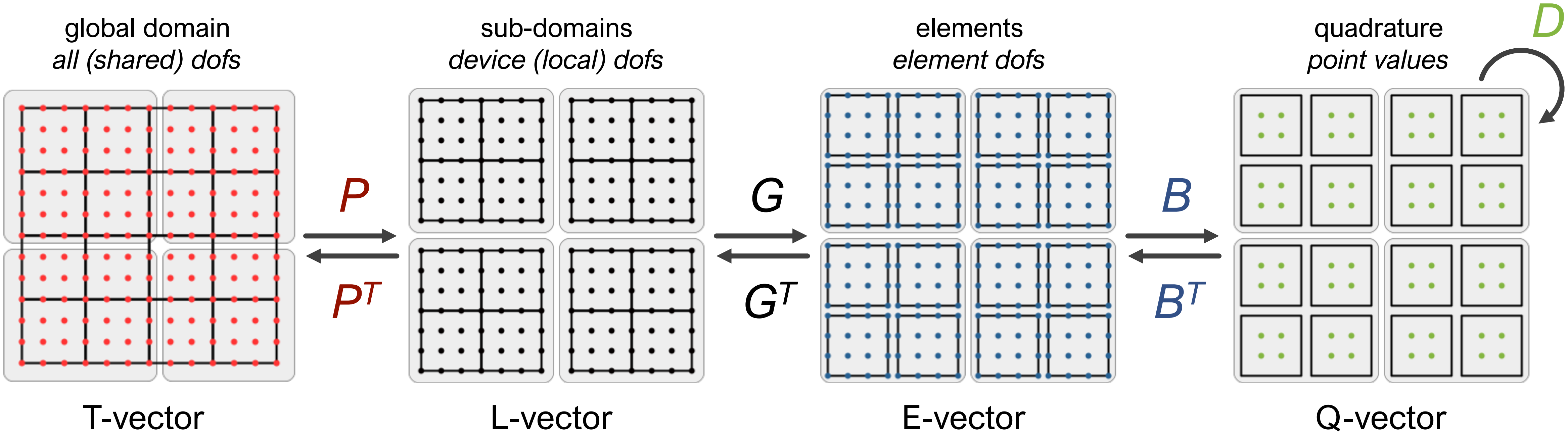


Parallel partitioning via Hilbert curve

FEM Operator Decomposition + Partial Assembly

Decompose A into parallel, mesh, basis, and geometry/physics parts

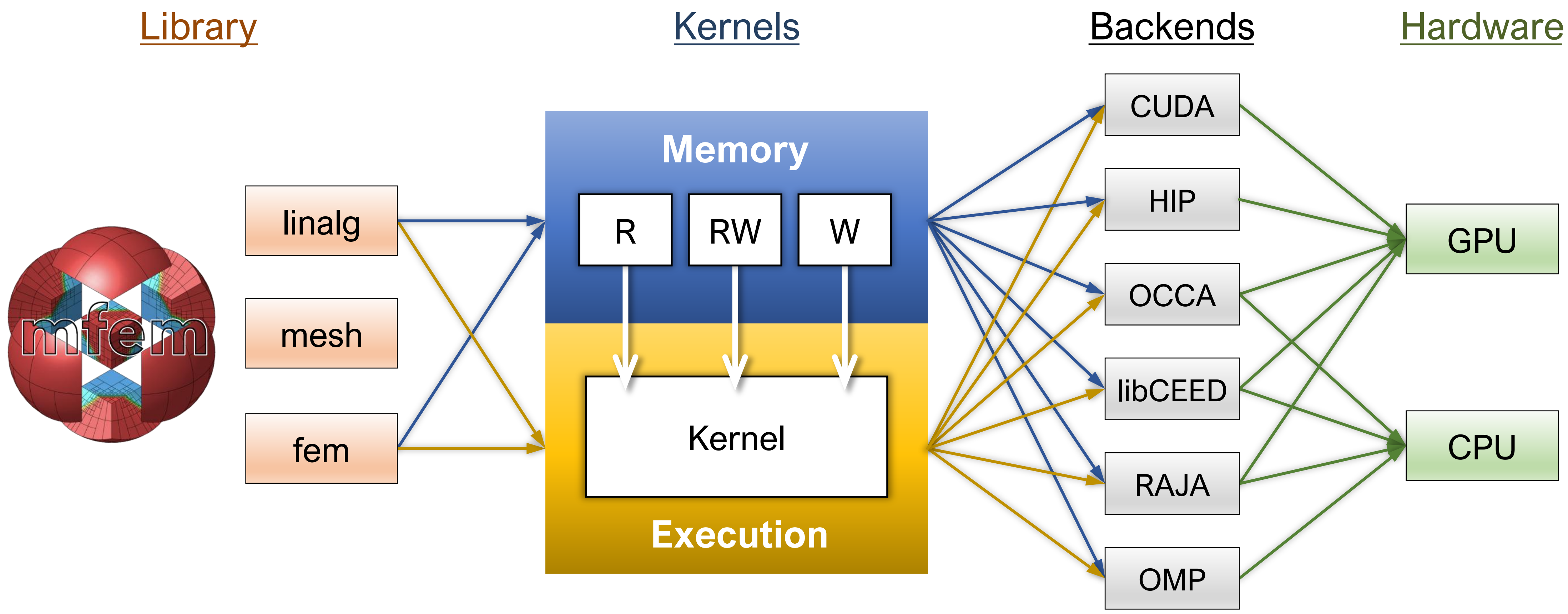
$$A = P^T G^T B^T D B G P$$



- **Partial assembly = store only D, evaluate B**
- **Optimal memory, near-optimal FLOPs compared to A**

GPU Support

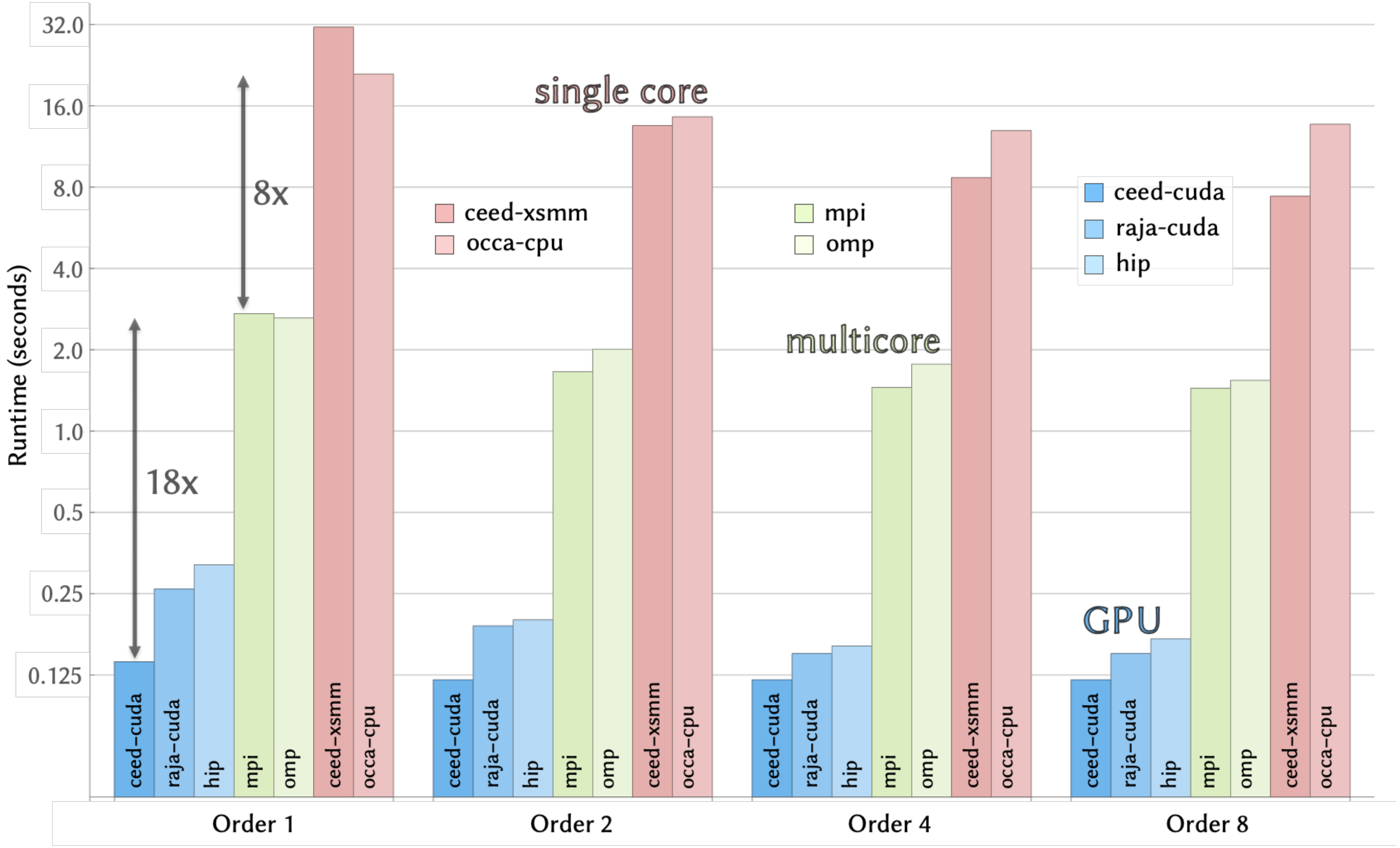
MFEM has provided GPU acceleration for over 2 years (since mfem-4.0)



- **Backends are runtime selectable, can be mixed**
- **Coming soon: support for Intel/SYCL**

GPU Support

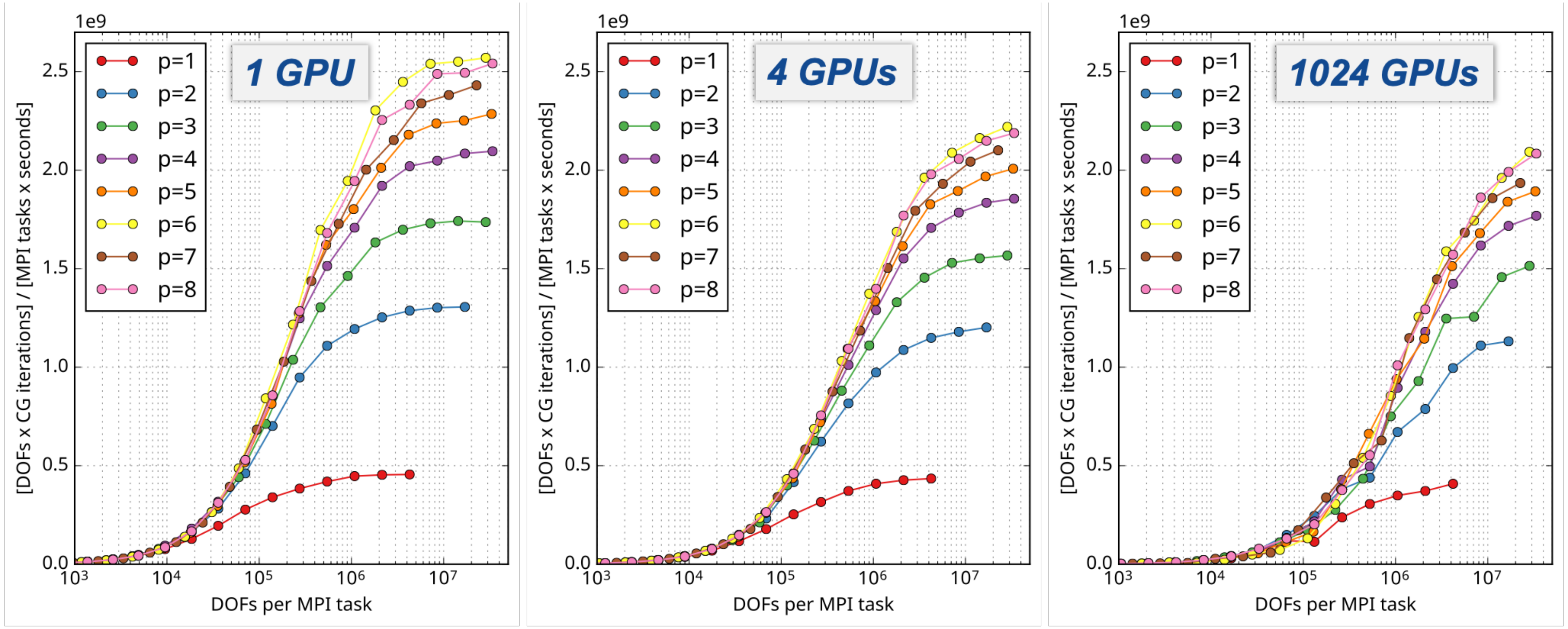
Device backends in MFEM, desktop performance

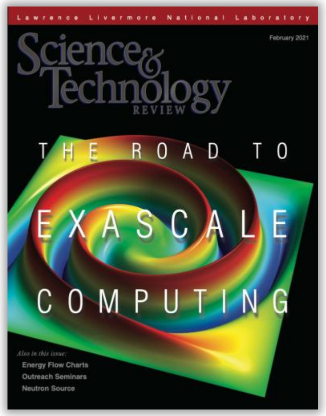


- MFEM-4.2
- October 2020
- Example 1
- 2D, 1.3M dofs
- 200 CG-PA iterations
- Intel Skylake 16 core (Linux)
- AMD MI60 (Corona)
- NVIDIA GV100 (Linux)
- sm_70, CUDA 10.1
- gcc-8.4.0

GPU Support

MFEM performance on multiple GPUs, 3D scalar diffusion, Lassen





- **Parallel scaling on Lassen (4 V100 GPUs/node)**
- **Local performance vs. local problem size**
- **Best performance: 2.1 TDOF/s**
- **Largest size: 34 billion dofs**

CEED
EXASCALE DISCRETIZATIONS

- Benchmarks (BPs)
- Miniapps (Laghos)
- libCEED

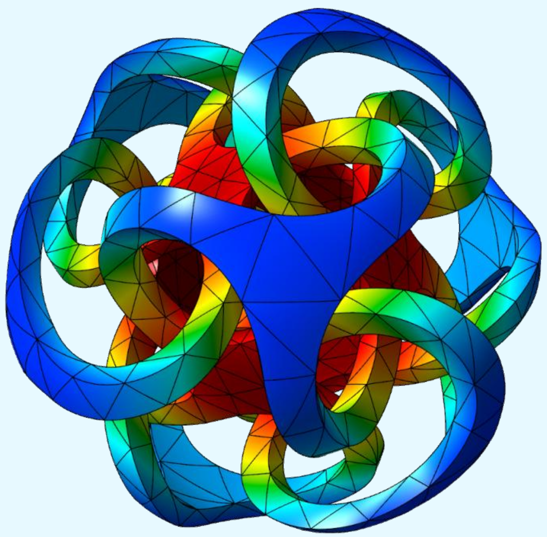
ceed.exascaleproject.org

Visualization

MFEM supports several options for accurate + flexible finite element visualization

GLVis

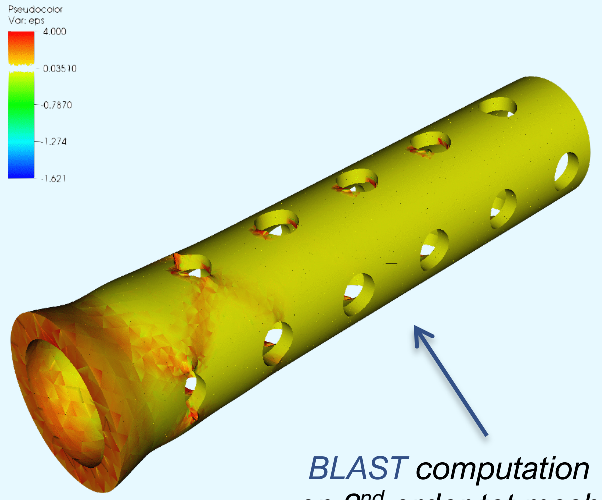
native lightweight visualization



glvis.org

VisIt

general data analysis tool

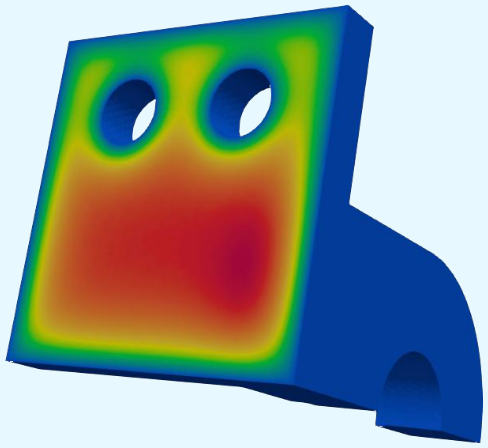


BLAST computation on 2nd order tet mesh

visit.llnl.gov

ParaView

general data analysis tool

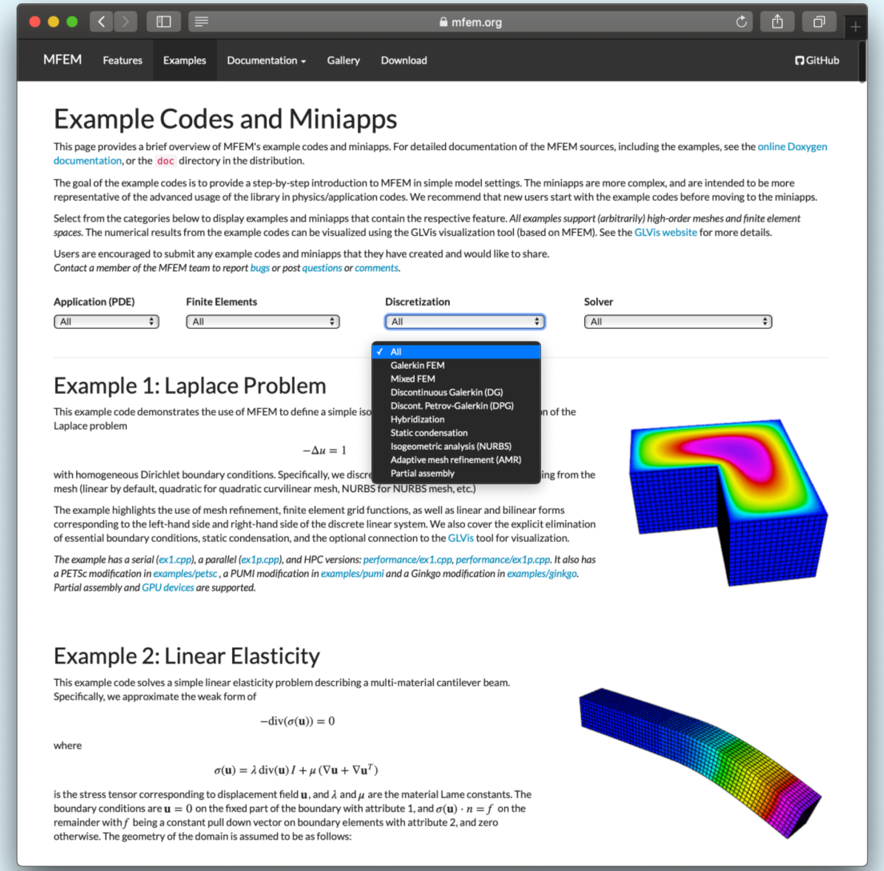


paraview.org

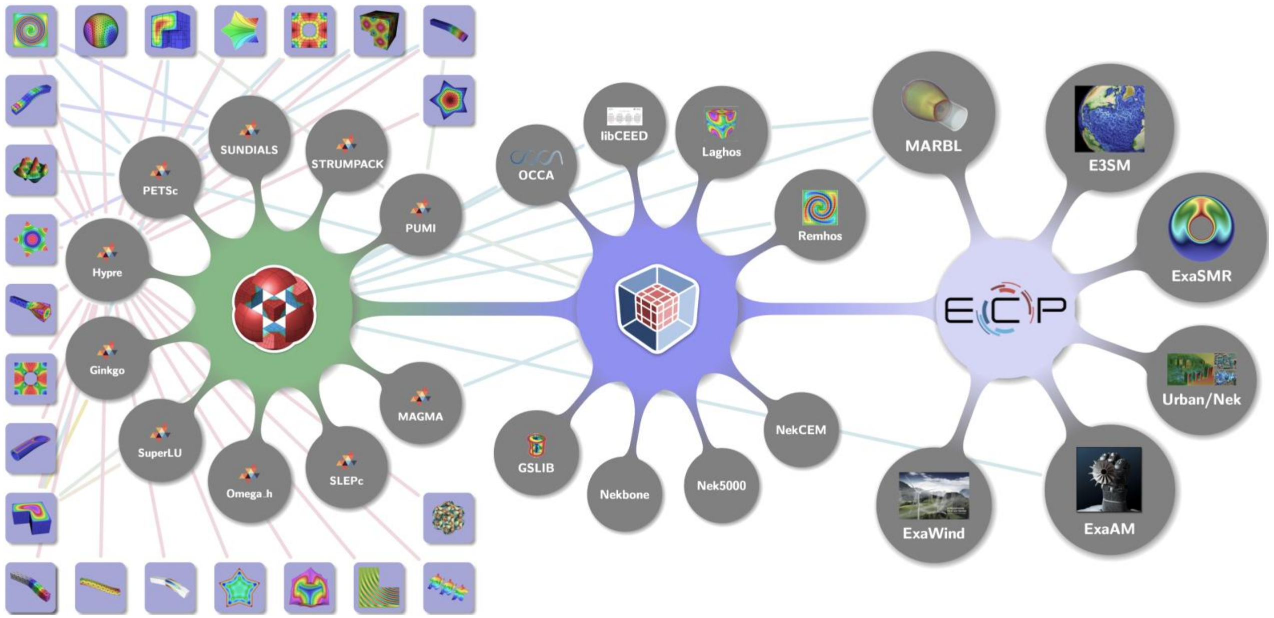
Additional I/O support: Conduit, ADIOS, VTK, FMS, …

Examples

The first stop for new users



mfem.org/examples



- 30 example codes, most with both serial + parallel versions
- Tutorials to learn MFEM features
- Starting point for new applications
- Show integration with many external packages, miniapps

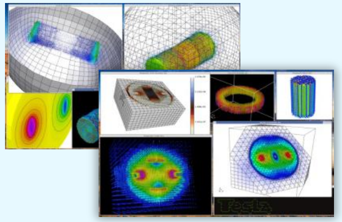
Miniapps

More advanced, ready-to-use physics solvers

Volta, Tesla, Maxwell and Joule Miniapps

Static and transient electromagnetics

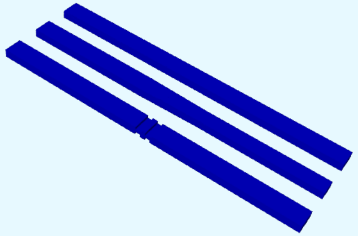
- **Volta** $-\nabla \cdot \epsilon \nabla \varphi = \rho - \nabla \cdot \vec{P}$
- **Tesla** $\nabla \times \mu^{-1} \nabla \times \vec{A} = \vec{J} + \nabla \times \mu^{-1} \mu_0 \vec{M}$



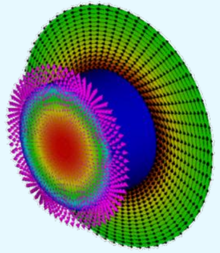
- **Maxwell** · *transient full-wave EM*

$$\frac{\partial(\epsilon \vec{E})}{\partial t} = \nabla \times (\mu^{-1} \vec{B}) - \sigma \vec{E} - \vec{J}$$

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E}$$



- **Joule** · *transient magnetics + Joule heating*
- Arbitrary order elements + meshes
- Adaptive mesh refinement



mfem.org/electromagnetics

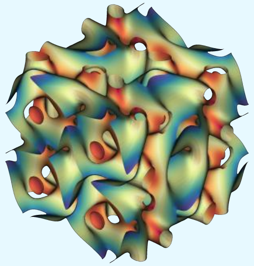
Navier Miniapp

Transient incompressible Navier-Stokes equations

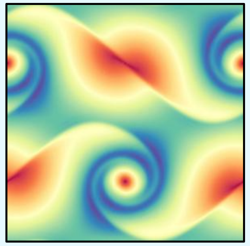
$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f}$$

$$\nabla \cdot \boldsymbol{u} = 0$$

- Arbitrary order elements
- Arbitrary order curvilinear mesh elements
- Adaptive IMEX (BDF-AB) time-stepping algorithm up to 3rd order
- State-of-the-art HPC performance
- GPU acceleration
- Convenient user interface



3D Taylor-Green vortex, 7th order

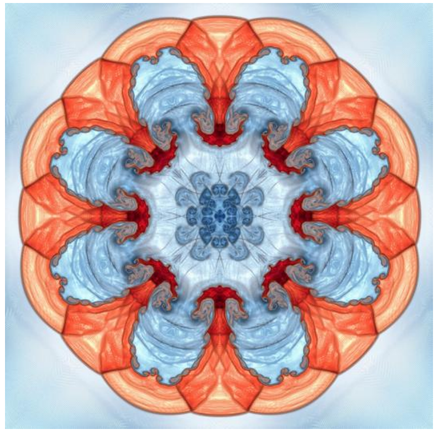


Double shear layer, 5th order, Re = 100000

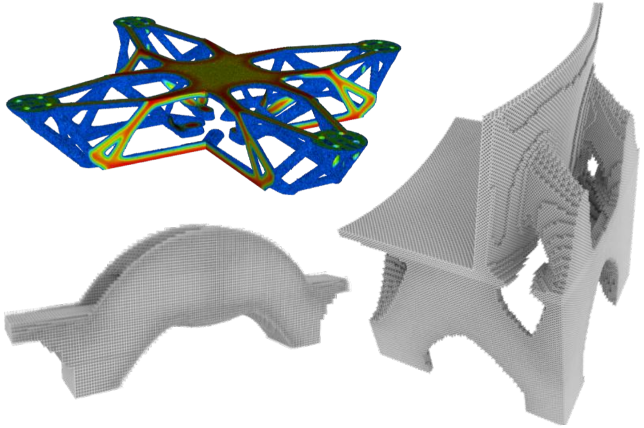
mfem.org/fluids

Applications

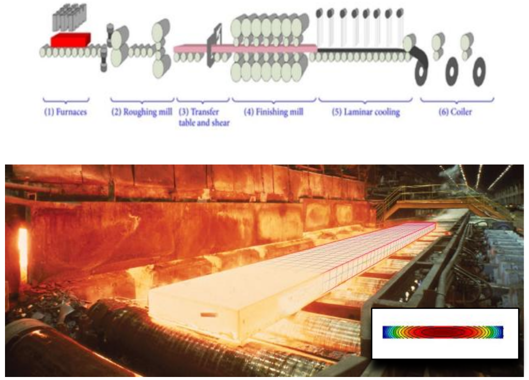
Some of the large-scale simulation codes powered by MFEM



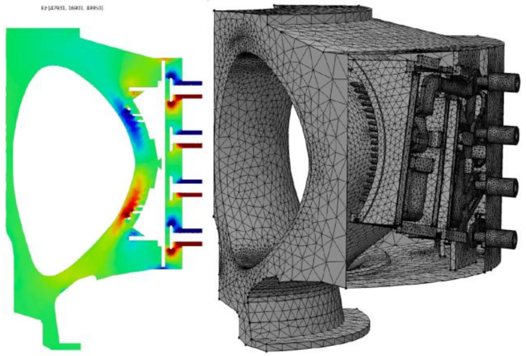
Inertial confinement fusion (BLAST)



Topology optimization for additive manufacturing (LiDO)



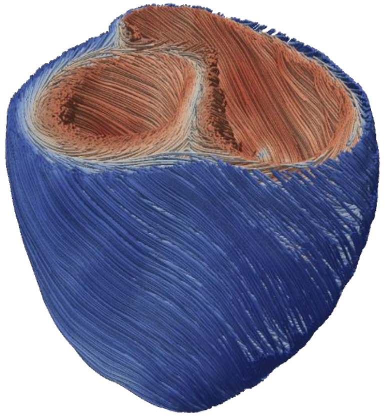
Hot strip mill slab modeling (U.S. Steel)



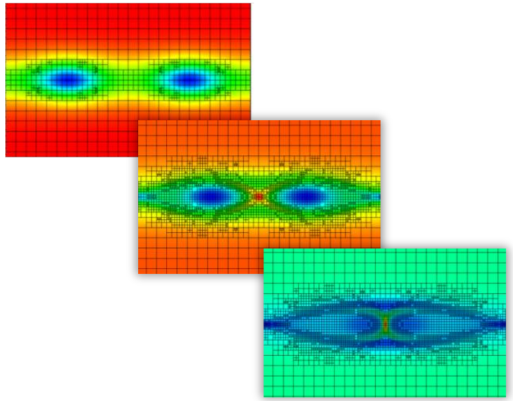
Core-edge tokamak EM wave propagation (SciDAC, RPI)

Electric aircraft design (RPI)

MRI modeling (Harvard Medical)



Heart modeling (Cardioid)



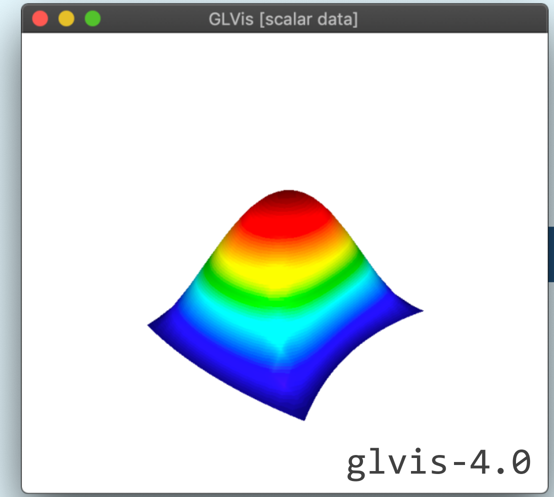
Adaptive MHD island coalescence (SciDAC, LANL)

Visualization

Web + Python support

Modern OpenGL

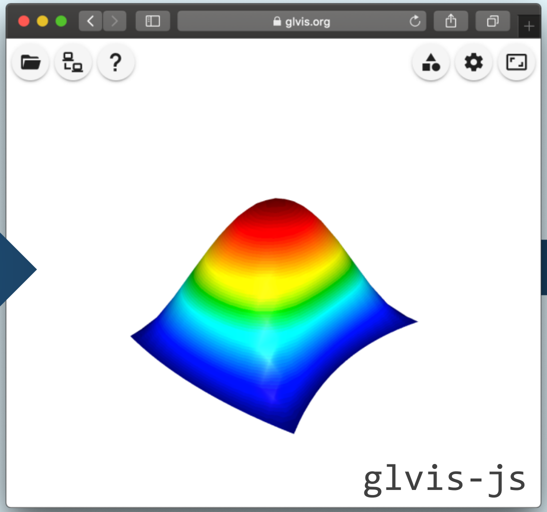
OpenGL3, SDL



glvis.org

Web Visualization

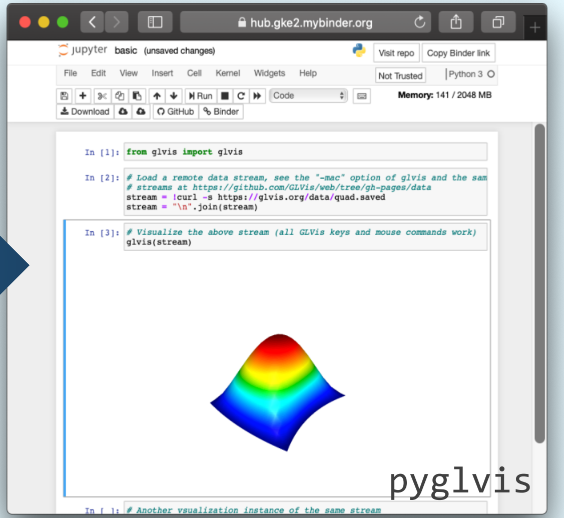
JS/WebAssembly, Emscripten



glvis.org/live

FEM in Python

PyMFEM, Jupyter, Binder



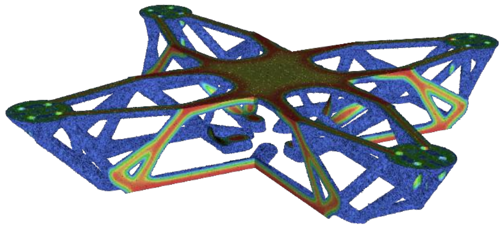
pip install glvis

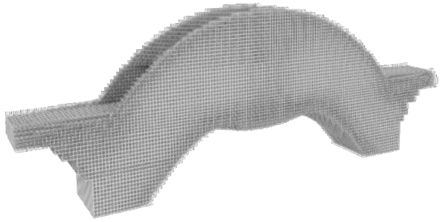
Try `glvis-js` and `pyglvis` in your desktop or mobile browser

Roadmap for Next Year

Plans for FY22

- **GPU support**
  - Parallel GPU assembly · Scalable solvers for the full HO de Rham complex
  - Performance on AMD GPU · Intel GPU support
  - Easier to write GPU kernels · Continued performance improvement · Cleaner PA code
- **Application needs**
  - Subdomain extraction · Different physics in different domains
  - User-defined flexible variational forms
  - Better handling of nonlinear problems
  - Large initial meshes · Mesh partitioning miniapp · Parallel re-partitioning
  - Automatic differentiation · Mixed meshes · ML/DRL · Interface sharpening
- **Code quality**
  - Improve Doxygen documentation · Additional examples + miniapps
  - Unify interfaces · Polish instead of adding new features
- **New releases**
  - v4.4 in January · v5.0 coming soon – *expect breaking changes!*





mfem-4.4
Due by January 31, 2022

mfem-5.0
Due by July 31, 2022

Future Roadmap

What should we tackle next?

- **Topics we are considering**
  - Design optimization
  - Cloud computing
  - MFEM in industry
  - Connections with ML
  - Automated compilation of variational forms · JIT support
  - ARM and post-GPU hardware
- **Release schedule**
  - Official release every 6 months
  - GLVis release 1 month after MFEM release

- **What would you like to see?**
  - Geometry pre-processing
  - Better error estimators
  - Meshes with elements of different dimensions
  - Parallel anisotropic AMR
  - 4D support
- **Please let us know**
  - Slack: #meet-the-team
  - GitHub: github.com/mfem/mfem/issues
  - Email: mfem@llnl.gov

Thank you from the MFEM team at LLNL!

Bob
Anderson
@rw-anderson

Julian
Andrej
@jandrej

Jamie
Bramwell
@jamiebramwell

John
Camier
@camierjs

Jakub
Cerveny
@jakubcerveny

Dylan
Copeland
@dylan-copeland

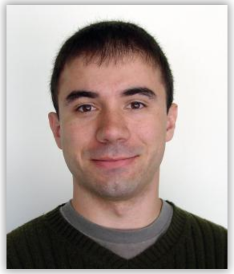
Veselin
Dobrev
@v-dobrev

Yohann
Dudouit
@YohannDudouit

Aaron
Fisher
@acfisher

Milan
Holec
@homijan

Brendan
Keith
@brendankeith



Tzanio
Kolev
@tzanio

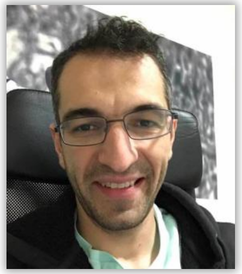
Boyan
Lazarov
@bslazarov

Ketan
Mittal
@kmittal2

Will
Pazner
@pazner

Socratis
Petrides
@psocratis

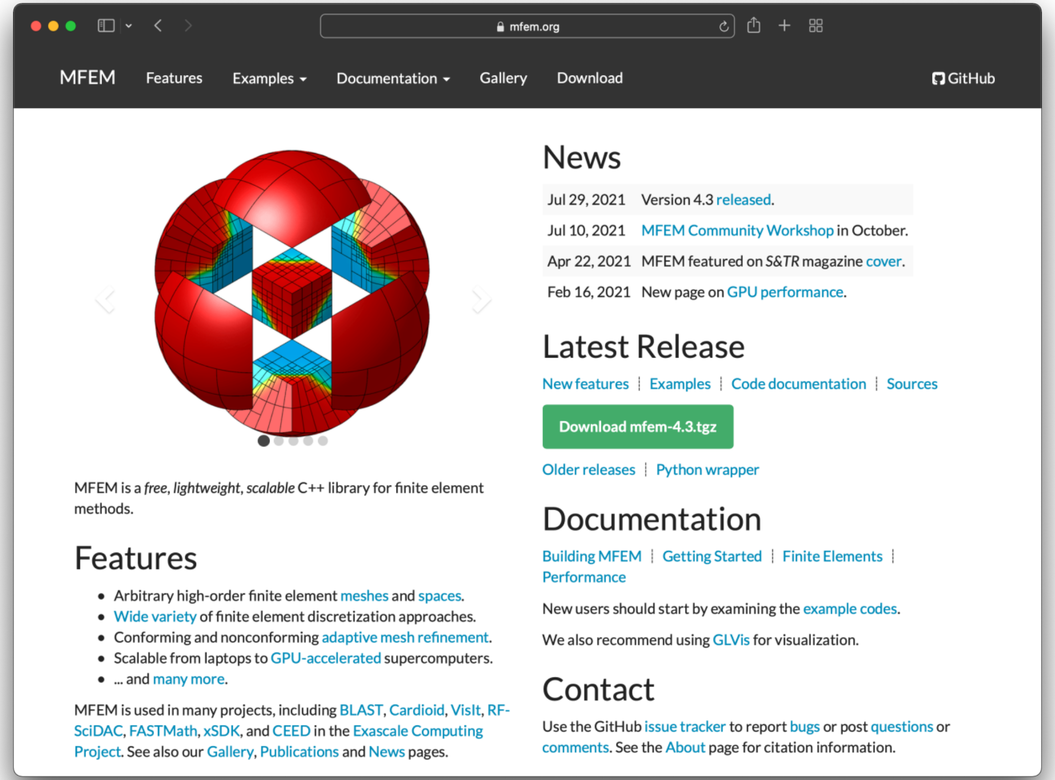
Mark
Stowell
@mlstowell



Vladimir
Tomov
@vladotomov

Chris
Vogl
@cjvogl

MFEM Resources



Website:
mfem.org

Software:
github.com/mfem

Publications:
mfem.org/publications

Email:
mfem@llnl.gov

- Contact us with questions + feedback
- Contribute to the code
- Explore our publications

mfem.org

Lawrence Livermore National Laboratory