RF Wave Simulation Using the MFEM Open Source FEM Package

J. Stillerman, S. Shiraiwa¹, J. C. Wright¹, J. P. Lee¹, P. T. Bonoli¹ and T. Koley², ¹MIT-PSFC, ²LLNL

Background: Need RF field solver for 3D cold SOL plasma

Motivation

Core-edge RF solver coupling (BP10.00041) opens possibilities of:

- Coax-to-core - self-consistent RF solution
- Unified solver for ICRF, HHF, and LHCD
- Hot asymmetrical core + cold 3D edge/antenna

However, we still need scalable FEM simulation for plasma wave simulation in whole 3D SOL cold plasma

- Short scale length of dielectric property in the perpendicular direction in SOL
- Slow wave excitation
- Nonlinear interaction would also excite short wave length modes.
- Scalability to massively parallel computer and transparency of implementation will be required

MFEM library

A free, lightweight, scalable library for finite element methods

- Higher-order Finite Element Spaces: H1-, H(div), H(curl)- conforming spaces, and more
- Triangular, quadrilateral, tetrahedral and hexahedral elements
- Tightly integrated with Hyphyl-code scalable library
- MPI-based parallelism throughout the library
- Various examples including Maxwell. eq. runs outs-of-box
- Written in C++

PyMFEM (Python wrapper)

- Allows for construct, manipulate MFEM C++ objects
- Supports both parallel (with MPI) and serial (w/o MPI) MFEM
- Allows for defining FunctionCoefficient using python class
- Supports passing numpy array as argument and return value

Create HyphylLibs/Python wrapper using distributed scipy.sparse matrix

Various examples (11/12 serial and 7/15 parallel) made from C++ examples, including time dependent problems and non-linear problems

Implementation

- Client/Server model
  - On Client: EM problem defined through python interface is "preprocessed" to make a input file
  - Resolve boundary index
  - Analysis geometry data
  - Set user defined variables, n, e, r, etc. profiles, equilibrium, etc.
  - Generate an input file (either python script or pickled python obj.) to define model object on the server.
  - On Server:
    - Support both serial/parallel operation.
    - Can run on cluster
    - Allow for editing the model
    - No graphical user interface

Solve inhomogeneous Maxwell eq. in 3D frequency domain

Various examples (11/12 serial and 7/15 parallel)

Create HypreParCSR (Hyphyl library)

Weak form

\[ \int_{\Omega} \nabla \cdot (\nabla \times E) \, dx = - \int_{\partial \Omega} \mathbf{E} \cdot \mathbf{n} \, ds \]

- Maxwell eq.
- Weak form

- Hypre ParCSR (Hyphyl library)
- Use on Open Source

Frequency Domain EM physics layer development

- Cold plasma + collisions as imaginary part of mass
  - Sca tensor with rotations
  - \( \varepsilon_0 = \rho R (\nabla \theta)^2 \)
  - \( \vec{E} \cdot \vec{H} = \nabla \times \vec{B} \)
  - \( \rho_0 = \rho_0 (\nabla \theta)^2 \)
  - Angles of magnetic field from \( \theta \)
  - Density and temperature at 3D array of the implemented as python class, which is called back from MFEM C++ MatrixVectorFunctionCoefficient object
- Port
  - \( \varepsilon_0 = \rho R (\nabla \theta)^2 \)
  - TE/TEM Case are implemented
  - \( \varepsilon_0 = \rho_0 \) is a given parameter
  - Linear system is extended using \( \varepsilon_0 \) as Lagrange multiplier
- Periodic BC
  - \( \varepsilon_0 = \rho R (\nabla \theta)^2 \)
  - \( \varepsilon_0 \) is eliminated from linear system using projection matrix \( \varepsilon_0 = P^* \)
  - \( \varepsilon_0 \) is a given parameter

C++ MFEM library is wrapped for access from python

- Construct/Access C++ objects method and members from python
- Add complex number support
- Add UMMS/MP (Direct Parallel complex sparse) Solver support
- Provide a framework to implement physics
- Requires nearly zero efforts to support both serial and parallel code

User interface built on nScape

nScape is All-in-one style python data analysis environment

- Shell/Editor/Enabler/Data Browser
- Project management
- Matplotlib figure with GUIs
- OpenGL, based 3D graphics
- GNU+ x license

GUI is built in order to provide:

- Easy model construction of EM problem
- Automated run of core code and
- Native support to browse MPI+Gpar data system reduces efforts required to run the simulation using experimental data.

Verification

- RF wave field pattern in the core is very similar on low field side.
- The difference on the high field side is due to the difference absorption mechanism (MFEM - cold plasma, TORIC - RF heating

H-minority ICRF heating on C-Mod

- 2D lower hybrid (LH) grill launcher
- 2D stratified cold plasma model
- 8 wave guide with 90 deg phasing
- Linear density profile

Solution obtained using MFEM is nearly identical to a cold plasma model using COSVOLD

Summary

Core spectral and edge FEM coupling opens the possibility “from coax to core” RF simulations (See BP10.00041)

- Model will permit realistic antenna models for ICRF and LH full-scale studies.
- With the addition of a triangular mesh for the vacuum region instead of the extended flue surface geometry now used, TORIC could model non-conforming antenna placement, accurate limiter geometries and wave propagation in the scrape off layer.

Large scale 3D application motivated to marge to open source scalable MFEM

- Python wrapper (PyMFEM) is developed for rapid development of physics layer and other missing pieces such as complex number UMMS support
- GUI developed on existing nScape scientific workbench.
- 3D EM frequency domain physics model is implemented
- Cold collisional plasma
- Various boundary conditions
- Verification are done
- H-minority ICRF heating
- LH grill launcher

Next Step

- Integrate into TORIC-FEM workflow (2D)
- Start investigating 3D field

This material is based upon work supported by the US DoE under the auspices of the US DoE under Contract DE-AC52-07NA27344. The work at Lawrence Livermore National Laboratory was performed under the auspices of the US DoE under Contract DE-AC52-07NA27344. This material is based upon work supported by the US DoE under Contract DE-AC52-07NA27344. The work at Lawrence Livermore National Laboratory was performed under the auspices of the US DoE under Contract DE-AC52-07NA27344.