

# ND-GAr 3D Electrostatic Field Simulation Update

Nicholas Johnston & Chris Hayes

IU Bloomington

August 3, 2020



**INDIANA UNIVERSITY**  
BLOOMINGTON

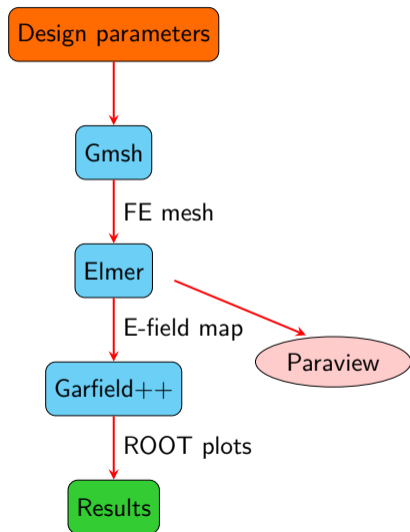


# Project goals

- Develop a full 3D electrostatic simulation of the ND-GAr TPC
  - Using Elmer FEM (Finite Element Method) to solve for the electric field in the TPC
  - Using Garfield++ for simulation of gas ionization and electron drift in a gaseous TPC
  - Using Gmsh to generate the geometry and finite element mesh for Elmer and Garfield++
- Use this 3D simulation to:
  - Test and compare TPC geometries, primarily focusing on the field cage
  - Constrain design parameters such as field strip width and pitch
  - Provide numerical studies on any aspect of the detector design involving the electric field
- Future extensions could include thermal and gas flow simulations with Elmer

# Simulation Workflow Overview

- Design parameters: Includes TPC width, radius, length, field strip width and pitch, etc
- Gmsh: Constructs geometry and generates Finite Element (FE) mesh for Elmer
- Elmer: Applies voltages to parts of the mesh and solves for the electric field
- Garfield++: Uses the electric field map from Elmer to simulate the detector response
- Paraview: 3D data visualization, primarily for debugging
- Results: Garfield++ provides results primarily in the form of ROOT plots

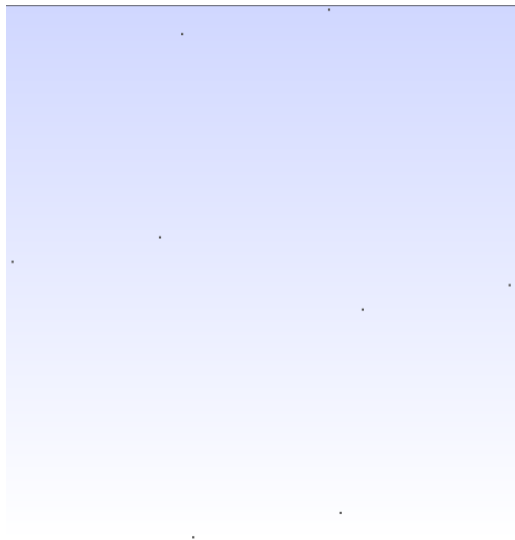


# Gmsh Overview

- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine

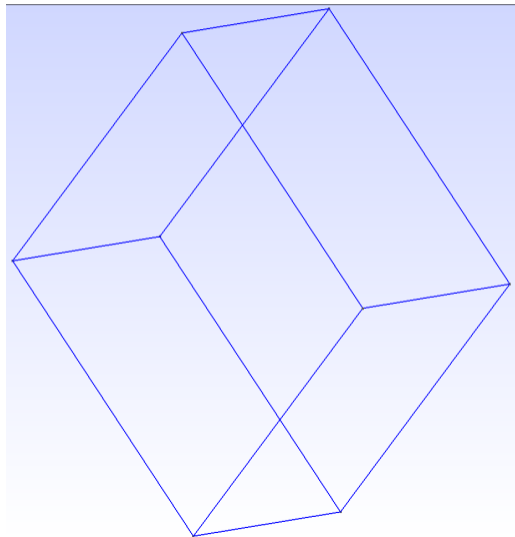
# Gmsh Overview

- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine
  - Start by defining points



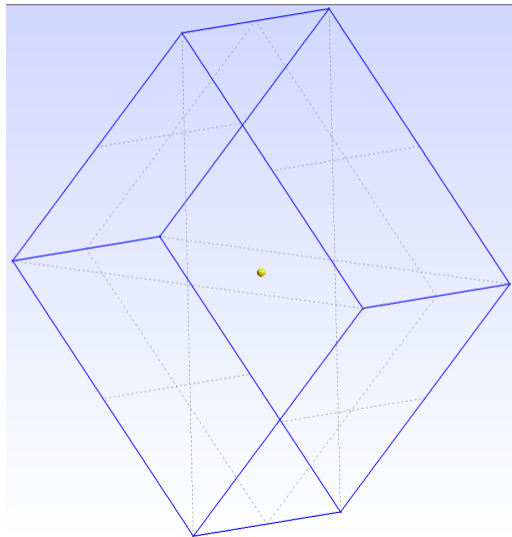
# Gmsh Overview

- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine
  - Start by defining points
  - Connect the points into lines defining the edges of surfaces



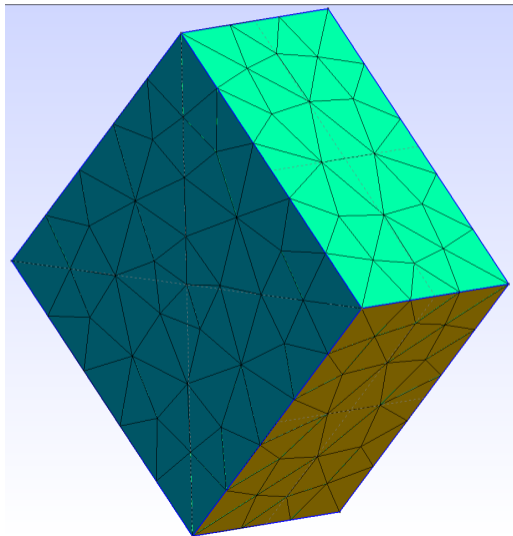
# Gmsh Overview

- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine
  - Start by defining points
  - Connect the points into lines defining the edges of surfaces
  - Define surfaces and connect into volumes



# Gmsh Overview

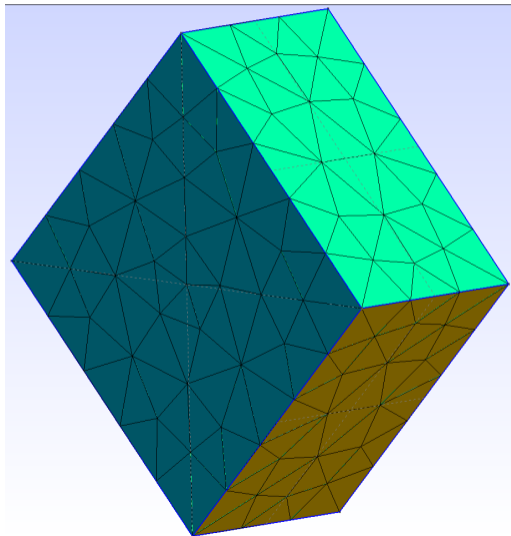
- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine
  - Start by defining points
  - Connect the points into lines defining the edges of surfaces
  - Define surfaces and connect into volumes
- A FE mesh is a method of discretizing a model into uniform elements, consisting of nodes at the corners and lines connecting them



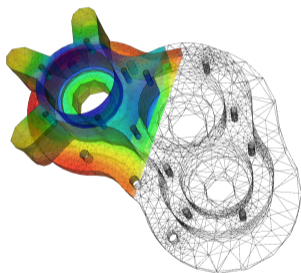


# Gmsh Overview

- Gmsh is a 3D finite element (FE) mesh generator
- Includes a relatively low level (but scriptable) geometry engine
  - Start by defining points
  - Connect the points into lines defining the edges of surfaces
  - Define surfaces and connect into volumes
- A FE mesh is a method of discretizing a model into uniform elements, consisting of nodes at the corners and lines connecting them
- Designs can be created manually with a GUI, scripted in Gmsh's language, or programmed using Gmsh's API



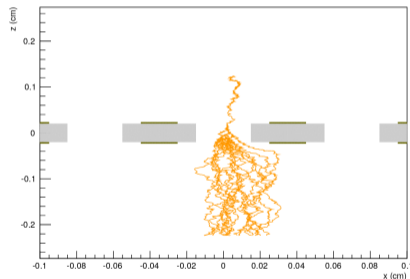
# Elmer FEM Overview



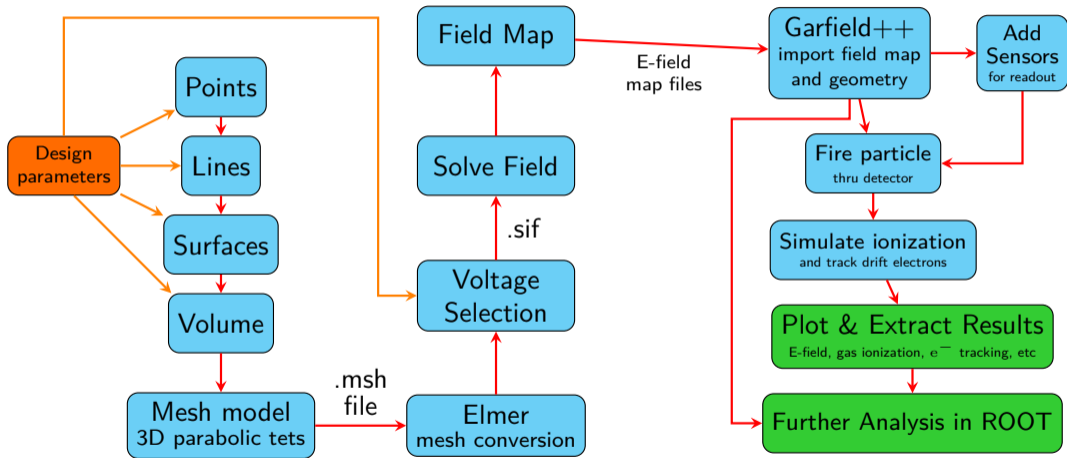
- Elmer is an open source physics simulation package, which includes models for fluid dynamics, heat transfer, electromagnetics, and more as PDEs
- Elmer imports a FE mesh in multiple formats (Ansys, universal format, gmsh, etc)
- Materials, equations, and boundary conditions are configured either via a GUI or human readable script (.sif)
- The finite element method (FEM) iteratively approximates solutions to infinite PDEs describing physical interactions by only allowing the mesh elements to interact on nodes
- Elmer iteratively solves the PDEs using the finite element method, producing a results file containing the final physical state of the simulation

# Garfield++ Overview

- Garfield++ is a package for simulation of particle detectors and successor to Garfield
- Features a limited native geometry capability, primarily imports field maps from FEM software
- Garfield++ simulates ionization in detector material from high energy particles
- Tracks electrons and ions, simulating their drift under and electric field
- Can also simulate the charge collection on sensor elements, allowing a full detector response study

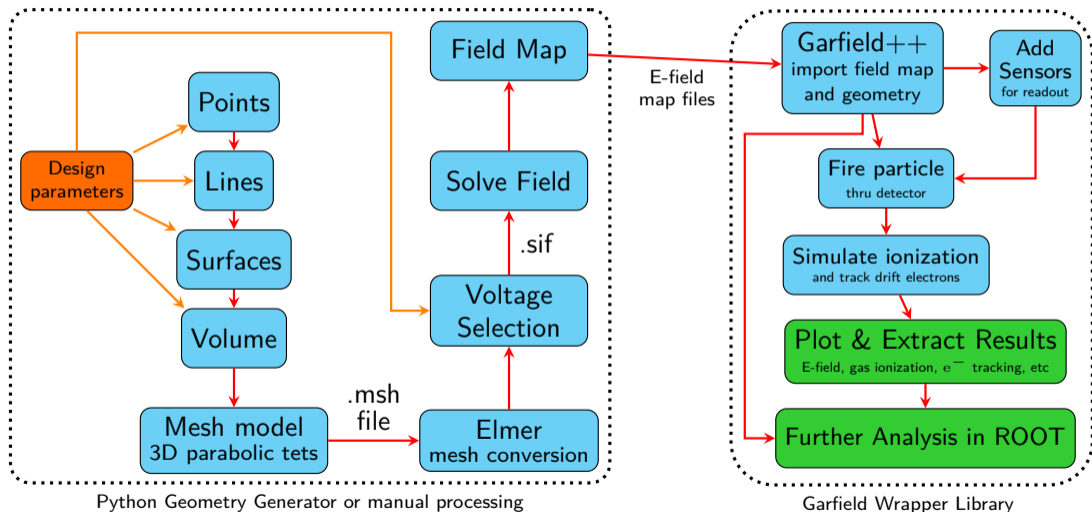


# Simulation Flowchart



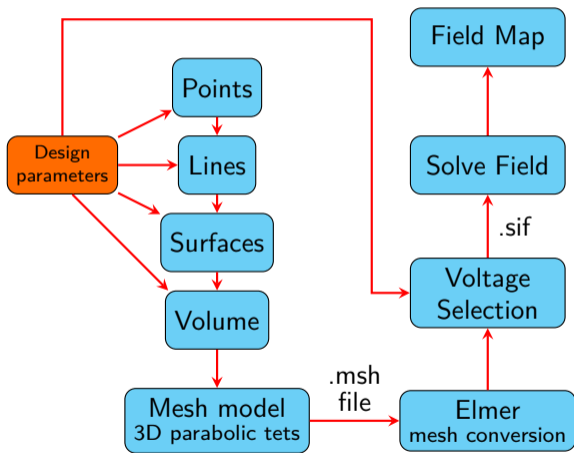
Crimson arrows indicate a step that must be performed manually or automated

# Simulation Flowchart



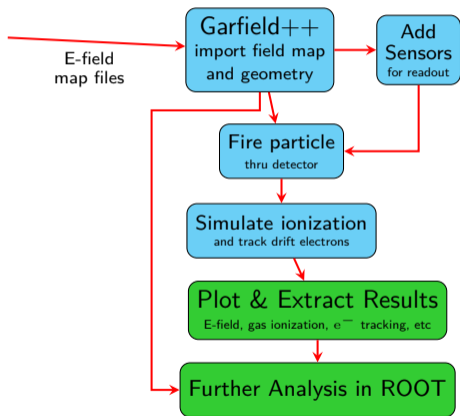
Crimson arrows indicate a step that must be performed manually or automated

# Geometry Generator and Elmer Control Module



- Manually executing this sequence is time consuming and infeasible for a systematic study, necessitating a program to automate the process
- Written in Python 3 using the GMSH Python API
- Parametrically defines a simple TPC can model (field strips, anode, and cathode)
- Generates the 3D parabolic tetrahedron mesh, and saves to a .msh file
  - The mesh node spacing can be controlled for higher resolution results
- Converts the mesh to the ElmerSolver format (needed by Elmer and Garfield++)
- Configures the FEM simulation (materials, voltages, etc)
- Runs ElmerSolver to create the electric field map

# Garfield++ Wrapper Library



- **Garfield++ is designed to study one simulation, but we need to compare many at once**
- **Condensing management of Garfield++ steps into a library facilitates this and simplifies writing new analysis**
- Written in C++, using ROOT and Garfield++
- Each analysis includes the wrapper library and is compiled as a standalone program
- Currently features a few analysis programs, including a mesh refinement study and electric field plotting
- Capabilities currently under development:
  - Gas ionization simulation
  - Electron drift
  - Signal readout (initially just charge collection on cathode plane)

## Technical Details

- Entire simulation is currently run on the IU Carbonate cluster computer (RHEL 7)
  - I also wrote a script to automate running many simulations simultaneously using the job system on Carbonate
- All supporting software was compiled from source
- Elmer requires a large suite of supporting software, Garfield++ only needs ROOT 6 and Geant4
- The mesh size is variable, generating Elmer-format meshes between  $\sim 10$  MB and  $\sim 4$  GB
- Maximum resolution is a mesh element size of approximately 30 mm, requiring approximately 6 hours to mesh and solve
- The simulation requires about 40GB of RAM per instance (at the highest mesh resolution)
- Currently single threaded, but Elmer supports parallel field solving

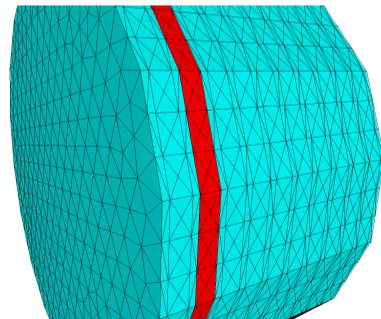
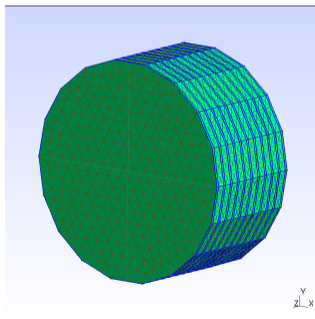
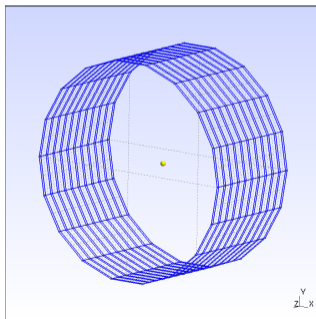


## Current capabilities

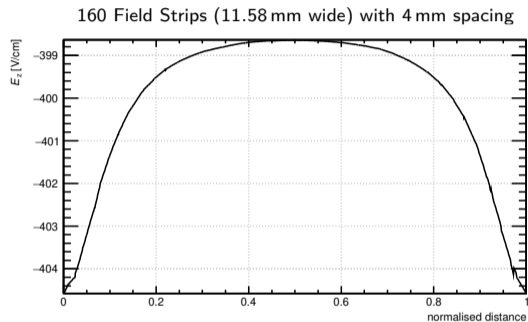
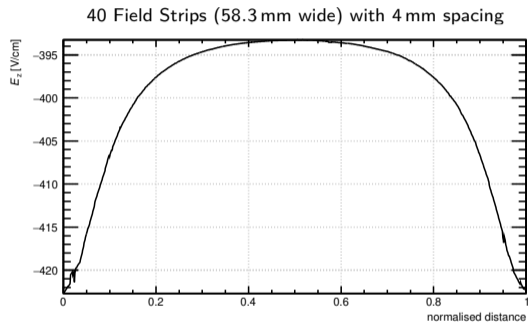
- We can perform full 3D simulations of the electric field in the ND-GAr detector with completely open source software
- It is presently parameterized by the number of field strips, sides, and strip spacing, but can easily be changed to perform other studies
- Our simulation currently includes voltage, dielectric materials, and gas properties, with the ability to add more in the future

# Current Detector Design

- Currently using a relatively simple "can" model as a proof of concept
- The design copies the ALICE TPC, but only simulates one side of the TPC (mirror symmetry)
- Radially divides the sides of a can into strips, alternately representing the field strips and the gap between them
- Elmer sets the field strip sections to the specified voltage, and the "gap strips" are ignored



## Example Results: Strip number comparison



- Plot the z-component of the E-field as a function of z, at points  $9/10^{\text{th}}$  of the TPC radius from the axis ( $y = 2.18$  m)
- The TPC model with more field strips shows less overall variation in the  $E_z$  field strength

# Summary

- Future capabilities:
  - Full particle ionization in Garfield++ is close to being fully implemented in our sim, allowing simulation of ionization electron drift and ground anode charge collection
  - More complex geometries either from more advanced gmsh scripts or professional 3D CAD packages can be imported with relative ease into the current FEM/Garfield++ system
- We have a number of studies in progress and expect to report results soon, including:
  - Electric field stability dependency on field strip pitch and width
  - Alternate drift chamber configurations (see Chris Hayes' talk)