

Geant V Detector Construction Development

Stable Release 17.7.20

Ryan Schmitz -- University of Minnesota, EP-SFT

A quick refresher

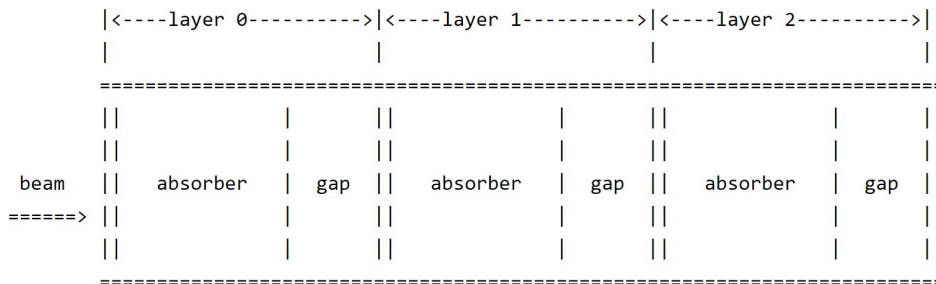
- Geant4 is useful, but can't use multiple cores
- The GeantV project attempts to rewrite Geant4 using modern algorithms with support for parallelization
- When I arrived, GeantV couldn't define its own detectors

Current project status

- An application has been written which creates a user-defined calorimeter

- Application inputs:

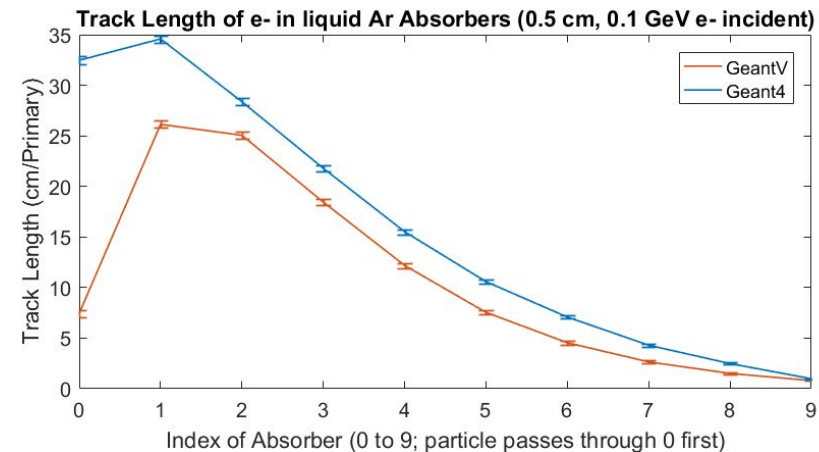
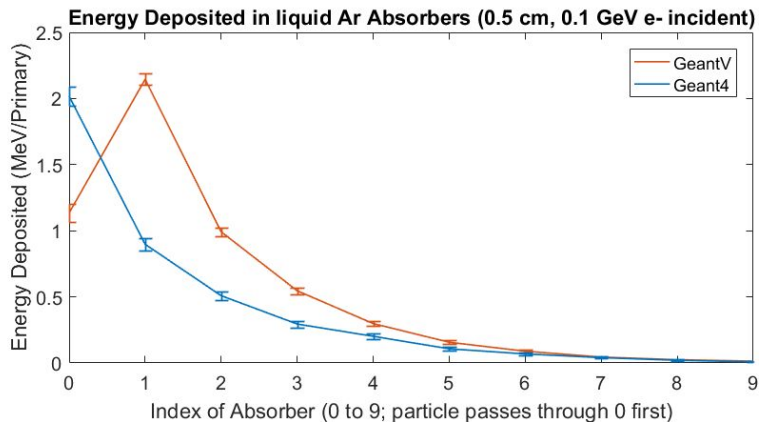
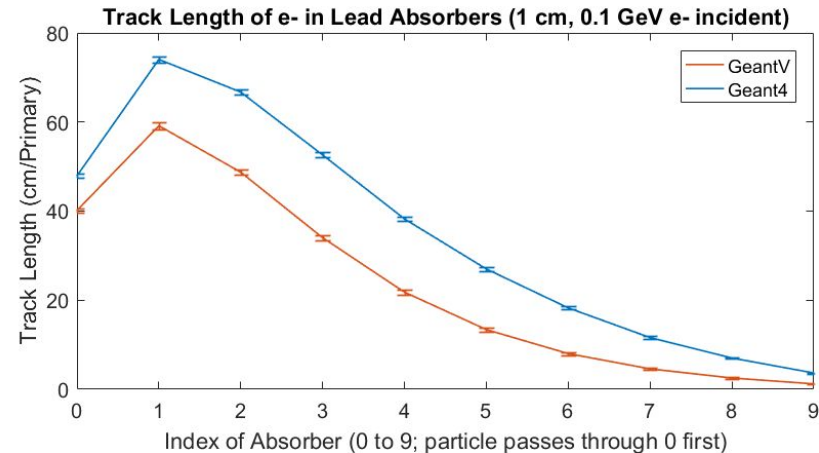
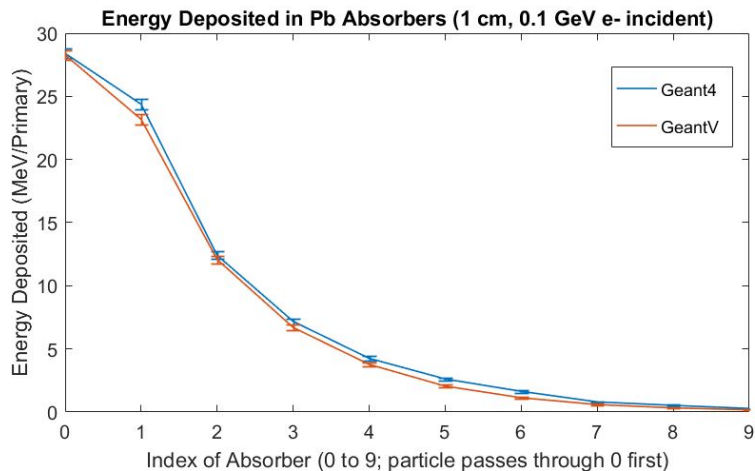
- Number of layers
- Number of absorbers (per layer)
- Absorber properties (thickness, material)
- Calorimeter dimensions
- Production cuts (length, energy)



- Application Outputs (per primary per absorber):

- Energy deposited
- Track length

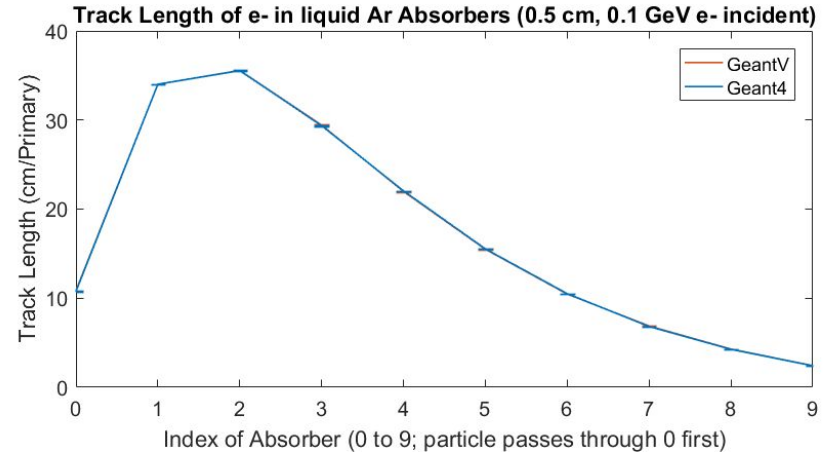
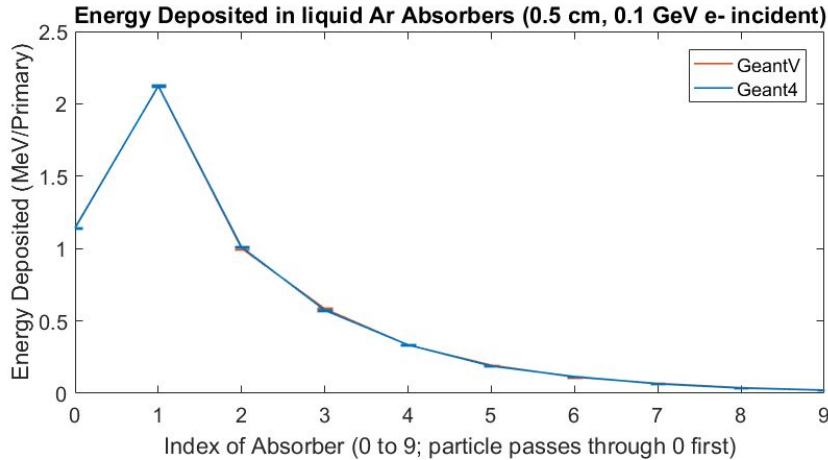
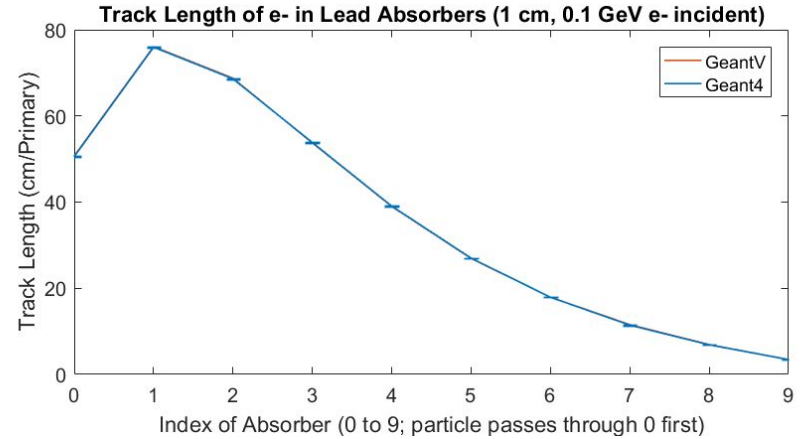
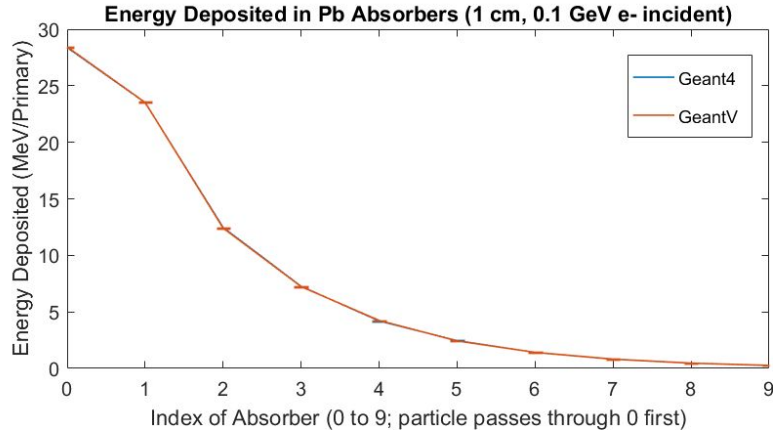
First round of outputs -- Success?



Issues

- Defining N layers and N absorbers didn't change the hardcoded "2 absorber, 10 layer" data analysis format/outputs
- No way to pass information between detector construction and other classes
- Initially unclear why the physics was so wrong

Round 2 -- No doubt



Future Plans

- Adding more data types to outputs
- Implementing improved scoring algorithm which collects std. dev. of data
- Generalizing the application -- possibly different geometry or more complicated calorimeter?