Geant4 Simulation of Proton Absorption Correction

Alicia Postuma April 2025

1 Motivation and Overview

The materials of the SHMS detector stack will act as proton absorbers, resulting in some primary tracks being stopped before reaching the second set of hodoscopes. Therefore, a fraction of real events do not form a 3/4 trigger. For accurate cross-section measurements, this affect must be quantified, and applied as a correction to the experimental yield.

Prior to this work, the total fraction of proton tracks which interact in the materials before the S2 hodoscopes was calculated from known material properties. The calculation takes the form $A = 1 - e^{-\Sigma_i X_i/\bar{\lambda}_i}$, where Σ_i denotes the sum over the materials of the focal plane, X_i the material thickness (g/cm^2), and $\bar{\lambda}_i$ is the average of the nuclear collision length λ_T and nuclear interaction length λ_I (g/cm^2), as given by PDG [1]. For details, refer to section 4.8 of John Matter's thesis [2] for the original correction. The most recent estimates of this value are given in a spreadsheet titled `SHMS_Proton_Absorption_jmatter.ods` (provided in the `shmsPA` GitHub repository).

The goal of this work is to create a Geant4 simulation incorporating the layers of materials in the SHMS focal plane to calculate the fraction of primary particles which are absorbed, causing a missed 3/4 trigger. The main advantage of using Geant4 is the ease of calculating separate corrections for different primary particle types or detector setups. Additionally, the Geant4 result is expected to be more precise since Geant4 includes energy and angular dependencies of cross-sections, whereas the spreadsheet relies on total cross-section. The simulation also encorporates real event kinematics (δ variation and focal plane coordinates). Finally, the simulation can distinguish between stopped particles and missed triggers, replicating 3/4 triggers created by secondary particles.

The key result of the simulation is that the number of stopped primary tracks is *not equal* to the number of missed triggers. "Proton absorption" is perhaps not the most appropriate name for this correction, as it does not include all stopped tracks. The fraction of primary tracks which stop before the S2 hodoscopes in the Geant4 simulation is 8.7±0.5% with the Noble Gas Cherenkov installed, compared to 9.33% using a direct spreadsheet calculation.

This document is primarily intended to provide instructions for the user on how to run the simulation. It should be possible to compute corrections while using the Geant4 code as a black box, where the user simply sets input parameters, runs the simulation, and analyzes the output. However, information is also provided about the details of the simulation. Users with some knowledge of Geant4 are welcome to modify the source code.

2 Running Instructions

The following instructions assume that your computer already has the Geant4 toolkit installed. If not, you will need to install Geant4: I recommend downloading the RPM and following the instructions for building from source. Make sure to compile with OpenGL/X11 enabled and install the optional data directories.

First, clone the simulation from GitHub: `git@github.com:acpostuma/shmsPA` The program is installed using the standard procedure with cmake. Create a separate `build` directory inside of the `shmsPA` folder, then run `cmake .. && make` inside `build`.

You can then call the simulation in one of two ways from the main `shmsPA` directory.

Interactive mode is called with the command: `./build/shmsPA`

This brings up a GUI with visualization, a command line, and a library of all possible commands with some guidance. Most tasks are not efficient in the GUI, but it is useful for familiarizing yourself with the simulation elements and commands.

Alternatively, the simulation can be run in batch mode by giving the name of a macro file as an argument. For example: `./build/shmsPA macros/runPA.mac`

In this case, the simulation will execute all the commands in the macro file without visualization. Calculation of missed trigger corrections will include running a large number of events at the correct kinematics in batch mode.

A separate correction needs to be computed every time you change (a) the experimental setup, for example changing the aerogel tray, (b) the SHMS central momentum, or (c) the primary particle type. To compute each correction, follow the steps:

1. Edit `detectors.dat` to reflect your experimental setup
2. Edit `macros/runPA.mac` (or create a new macro file) to specify the appropriate particle type, central momentum, and output file name
3. Run the simulation: `./build/shmsPA macros/runPA.mac` (or specify a new macro file)
4. Consult Appendix A to pick appropriate PID cuts for your particle and momentum
5. Edit `CalcMissed.C` to reflect your analysis cuts (lines 67 to 81)
6. Run `CalcMissed.C` on your output file. The script takes as arguments the input file name (as a string) and the central momentum in GeV (as a double)

Both `detectors.dat` and `runPA.mac` are heavily commented and should be self-explanatory, however for the sake of completeness they are also explained here. The file `detectors.dat` includes two options. `Use-NGC:` is a boolean variable for whether or not the noble gas Cherenkov is installed. `Aero-Tray:` takes an integer as an argument, corresponding to the index of refraction of the aerogel tray. The options are 11 (n=1.011), 15 (n=1.015), 20 (n=1.020), and 30 (n=1.030).

The macro file `runPA.mac` is separated into two blocks. The first is general setup commands and should not need to be modified. The second contains the commands that should be modified for every correction. `/PA/generator/momentum <value> <unit>` sets the SHMS central momentum. `/gun/particle <name>` sets the primary particle type. The options most relevant to this simulation are `proton`, `pi+`, and `kaon+`. `/analysis/setFileName <filename.root>` specifies the name of the output file to be created. Note that if a file already exists with the name specified, it will be overwritten. Finally, `/run/beamOn <nEvents>` specifies the number of events to run. Multiple `beamOn` commands can be given in one macro file, however each time a new run starts, a new file will be opened. If a separate file name is not given between the first and second `beamOn`, the second run will overwrite the first.

3 Technical Details

`shmsPA` is created from the Geant4 example program `basic/B5`, which is distributed with the Geant4 source code. `B5` consists of two spectrometer arms, with drift chambers, hodoscopes, and calorimeters as detectors. The skeleton of `shmsPA` is very similar to B5, but significant modifications have been made to all critical classes.

The physics list used is FTFP_BERT (Geant4 default as of 2025), with added G4OpticalPhysics to enable Cherenkov radiation, and G4EMLivermore for higher precision of electromagnetic interactions at low energies. The random engine used is Rancecu, a random number generator with a relatively long period and good statistical properties [3]. The random seed is reset every run so that the exact same input will provide slightly different results according to the appropriate probability distribution functions. As of April 2025, the simulation does not support running in multithreaded mode. (This is because the code was predominantly written on Alicia's single core laptop. Fixing the errors which result from running in MT mode is planned for a future improvement to the code. However, since the program only takes a few hours to run 100K events in single-core mode, this is not the highest priority.)

The program contains the following custom classes:

- `PAPrimaryGeneratorAction:` Event generation, option to use Geant4 particle gun, or to read in focal plane variables and create realistic particle trajectories
- `PADetectorConstruction:` Defines geometry and materials of detectors
- `PARunAction:` Actions taken per run, creates output Ntuple
- `PAEventAction:` Actions taken per event, writes data of each event to Ntuple
- `PASteppingAction:` Events taken per step, collects photoelectrons in Cherenkovs and energy in calorimeter
- `PAActionInitialization:` Initializes Run, Event, and Stepping actions
- `PAHodoscopeSD:` Records hits in each hodoscope
- `PAHodoscopeHit:` Defines what data is recorded in a hodoscope hit

3.1 Detector Construction

This is not a full simulation. Since the goal is to simulate particles stopping in materials, the critical aspects to include are realistic material properties and thicknesses of materials encountered by the primary particle. Materials and thicknesses are modelled after the spreadsheet, but where possible, primary sources (DocDB and prior theses) are consulted instead of taking values directly from the spreadsheet.

The coordinate system is set with (0,0,0) as the focal plane, and the z-axis along the direction of particle motion. The detector stack consists of a series of rectangular slabs along the z-axis. The hodoscopes are simulated with accurate x–y dimensions, but all other spectrometer components are given the arbitrary x–y dimensions of $(2\times2)\mathrm{m}^2$. The target is cylindrical, with a radius of 20cm. (The rationale for this choice is discussed in Section 3.2, Event Generation.)

Additional simplifications are made; first, each hodoscope is modelled as a single layer of PVT, instead of a series of overlapping bars. Similarly, an equivalent thickness is calculated for the drift chamber field wires and sense wires, and these are included as thin sheets of material. The curvature of Cherenkov mirrors is not simulated; these are taken to be flat.

The hodoscopes are set up as sensitive detectors, and a step limit of 1mm is implemented, forcing Geant4 to compute at least 5 steps inside the 0.5cm thick PVT hodoscope bars, and 25 in the 2.5cm thick quartz plane.

Materials are defined in one of three ways: (a) elements are called as a pre-defined `G4Element`, (b) some common materials (such as PVT, air, and lead glass) are available as a pre-defined `G4Material`, and (c) others are manually constructed as custom

`G4Materials`. The aerogel, in particular, is created as a custom material with density dependent on the index of refraction set by the user as $\rho = (n-1)/0.21 g/cm^3$.

In order to include the Cherenkov process, the refractive index of each material must be manually defined over a range of photon energies. The index is taken to be constant over the energy range for the three Cherenkov detectors, but is defined to have some variation for the quartz [4].

Oddly, the materials as defined in Geant4 differ from PDG values [1], specifically in the nuclear interaction and collision lengths. However, the distribution of stopped tracks throughout different volumes (see Table 1) is very similar between Geant4 and the spreadsheet, so these small discrepancies do not seem to have a huge effect.

Component	Material	Thickness (cm)	Geant4 (%)	Calc (%)
target	LH_2	5.00	6.0	7.5
HB entrance	Al	0.030	1.0	0.8
dipole exit	Al	0.050	1.6	1.6
		Pre-SHMS	**8.6**	**9.9**
entrance	Tedlar	0.010	0.2	0.1
gas	70/30 Ar/Ne	200.00	3.4	3
mirror	SiO_2	0.30	7.6	8.3
support	Rohacell	1.80	3.4	2.9
exit	Tedlar	0.010	0.2	0.1
		Noble Gas Cherenkov	**14.6**	**14.1**
entrance	Mylar	0.0025×2	0	0.05
gas	50/50 Ar/Ethane	3.91×2	0.2	0.07
wires	W	0.0005×2	0.02	1.1
cathode	kapton	0.9×2	3.4	3.6
exit	Mylar	0.0025×2	0.04	0.05
		Drift Chambers	**3.6**	**4.8**
entrance	Al	0.10	2.9	3.1
gas	C_4F_{10}	104.44	14.7	13.2
mirror	SiO_2	0.30	8.3	8.3
exit	Al	0.10	2.9	3.1
		Heavy Gas Cherenkov	**28.7**	**27.7**
entrance	Al	0.13	3.6	4
tray (n=1.030)	Aerogel	9.00	15.6	16.2
air gap	Air	17.10	0.3	0.3
exit	Al	0.16	4.4	5
		Aerogel Cherenkov	**24.0**	**25.5**
S1X	PVT	0.50	6.6	5.7
S1Y	PVT	0.50	7.5	5.7
S2X	PVT	0.50	6.4	5.7
		Hodoscope	**20.5**	**17.1**

Table 1: Distribution of total stopped tracks across the different volumes of the simulation, with the material and thickness of each volume specified. The spreadsheet calculation predicts 9.33% of protons being absorbed, whereas Geant4 reports 8.6±0.5% across $2 < P_{SHMS} < 10$ GeV (with the NGC installed).

3.2 Event Generation

There are two ways to generate events: first, the standard Geant4 particle gun functionality can be used, or realistic trajectories can be simulated from real event data.

Real event kinematics are turned on/off by the command `/PA/generator/usGenerated <bool>`. If `useGenerated` is set to `true`, the particle gun position and direction can no longer be changed, and the momentum can be changed only using `/PA/generator/momentum <value> <unit>`, however the particle type is still changed using `/gun/particle <name>`.

Real event generation uses focal plane variables from real data (run 5055): x_{fp}, y_{fp}, x'_{fp}, y'_{fp}, and δ. The focal plane is chosen as $z = 0$, such that the focal plane coordinates are $x_{fp} = x(0)$, $y_{fp} = y(0)$. The events are taken to originate from the centre of the target in z, denoted z_{tar}. Given the definitions $x' = dx/dz$ and $y' = dy/dz$, we can approximate positions at the target as:

$$x_{tar} = x_{fp} + x'_{fp} * z_{tar}$$

$$y_{tar} = y_{fp} + y'_{fp} * z_{tar}$$

Then, the event is generated at coordinates $(x_{tar}, y_{tar}, z_{tar})$ with momentum direction $(x'_{fp}, y'_{fp}, 1)$ and momentum $\mathrm{P}(1 + \delta/100)$, where P is the SHMS central momentum.

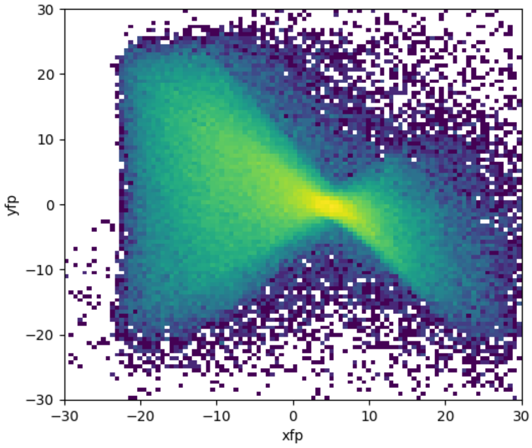

Figure 1: Filling of the focal plane by the events of run 5055.

These back-calculated pseudo-target positions fill a region roughly $-20 < y_{tar} < 10$cm and $-10 < x_{tar} < 20$cm, therefore the target in Geant4 is given a radius of 40cm to ensure all events originate inside the LH_2 volume (as opposed to the 4cm inner radius of the real target cell).

3.3 Simulation Output

Simulation output takes the form of a CERN ROOT TNtuple titled `PA`, containing the following branches:

- **S1XEnergy**: total energy deposited in S1X (MeV)
- **S1XTime**: time of hit in S1X (ns)
- **S1YEnergy**: total energy deposited in S1Y (MeV)
- **S1YTime**: time of hit in S1Y (ns)
- **S2XEnergy**: total energy deposited in S2X (MeV)
- **S2XTime**: time of hit in S2X (ns)
- **S2YEnergy**: total energy deposited in S2Y (MeV)
- **S2YTime**: time of hit in S2Y (ns)
- **S2YNPE**: total NPE generated in S2Y
- **NGCNPE**: total NPE generated in NGC active volume
- **HGCNPE**: total NPE generated in HGC active volume
- **AGCNPE**: total NPE generated in aerogel tray
- **CalEnergy**: total energy deposited in calorimeter (MeV)

The simulated data is calculated and collected in two ways: using sensitive detectors and user actions. The energy deposit and time of hit in each hodoscope is recorded by a sensitive detector `PAHodoscopeSD`. A sensitive detector records interations with materials in terms of a Hits Collection, which is initialized at the start of an event, and then the hits are processed at the end of an event. The hits themselves also have a class, `PAHodoscopeHit`, determining what data is recorded per event.

The advantage of using an SD and hits is that the hits can be accessed from anywhere in the simulation, and sample hodoscope SD and hit classes are already provided in the `B5` example program, on which this work was based. However, not all information (notably NPE) can be accessed via sensitive detector. An alternative method is to manually save the information in the user action classes. The NPE in each Cherenkov detector is therefore recorded per step in `PASteppingAction`, as is the energy deposited in the calorimeter. The hodoscope S2Y uses both methods of data collection, with time of hit recorded in the SD, and NPE counted in the stepping action.

Several approximations are made. First, no PMTs are simulated, and so energy and Cherenkov photons are summed all throughout the active volume with 100% detection efficiency and no geometric constraints. Second, there is no wavelength selection, so all photons produced are recorded regardless of energy. Third, there is no attenuation or similar effects in the PVT bars or reabsorption in the Cherenkovs.

This idealized situation means that the output is not truly comparable to physics data, but that simulated detector responses are a good reflection of the simulation truth (PID and momentum) of the particle track. Since efficiencies are separately calculated and applied to data, it makes sense not to include these in the calculation of the missed tracks correction.

References

[1] Particle Data Group. Atomic and nuclear properties of materials. Webpage: http://pdg.lbl.gov/2017/AtomicNuclearProperties/index.html.

[2] John Matter. *Ruling Out the Onset of Color Transparency up to $Q^2 = 14.2$ GeV^2 in Quasielastic $^{12}C(e, e'p)$ Scattering.* PhD thesis, University of Virginia, 2021.

[3] P. L'Ecuyer. Efficient and portable combined random number generators. *Commun. ACM*, 31(6):742–751, June 1988.

[4] Václav Prajzler, Václav Chlupatý, and Zuzana Šaršounová. The effect of gamma-ray irradiation on bulk optical plastic materials. *Journal of Materials Science: Materials in Electronics*, 31:1–17, 12 2020.

A Diagnostic Output

The following plots show simulated output of the detectors for different particles (p/π^+/K^+/e^+) across a range of momentum (2<P<8 GeV). Unfortunately, the detector output varies significantly with momentum, so it is necessary to change the PID cuts used in `CalcMissed.C` with significant changes in SHMS momentum. Suggested cuts are described below and summarized in Table 2.

It is not recommended to use the same cut values here as in your physics analysis. Since the simulation makes a number of approximations, such as recording all NPE generated in a volume (regardless of energy or position/direction of motion) not incorporating detector efficiencies, the simulated output is not expected to replicate the real thing. It is, however, expected to accurately represent the simulated truth of a track.

The heavy gas Cherenkov (Fig. 3) has an index of refraction n=1.0014, so a threshold velocity of $\beta = 0.9986$, which for a pion corresponds to a momentum of 2.7 GeV. Therefore, below 3 GeV or so, the HGC can not reliably be used for pion identification.

The aerogel Cherenkov (Fig. 2) is the most highly momentum dependent, and also depends on which tray is installed. Below 3 GeV, only pions and positrons Cherenkov. On the other hand, the momentum threshold for protons is only 6.3 GeV, so at high SHMS momentum all particles will Cherenkov in the AGC. The NPE produced by a proton, however, remains less than that produced by a meson. There still exists a kaon/proton threshold at high momentum, but it changes rapidly, and the user may have to run both kaon and pion simulations to choose where to place the cut for their analysis.

The noble gas Cherenkov (Fig. 4) has a clean and consistent signal across momenta. The suggested cut is NPE>5 for positrons, and NPE<5 for hadrons.

The calorimeter (Fig. 5) energy distributions overlap significantly for hadrons, but positrons have a unique distribution. A cut of Edep/PSHMS¿0.8 is suggested for positron identification, with no cut suggested for hadrons.

	NGC (NPE)	HGC (NPE)	AGC (NPE)	CAL (E_{dep}/P_{SHMS})
p	<5	<200	see plot	–
K	<5	<200	see plot	–
π	<5	see plot	>60	–
e	>5	>500	>60	>0.9

Table 2: Summary of suggested PID cuts. Those marked “see plot” are highly momentum dependent. Refer to Fig. 2 for the AGC and Fig. 3 for the HGC.

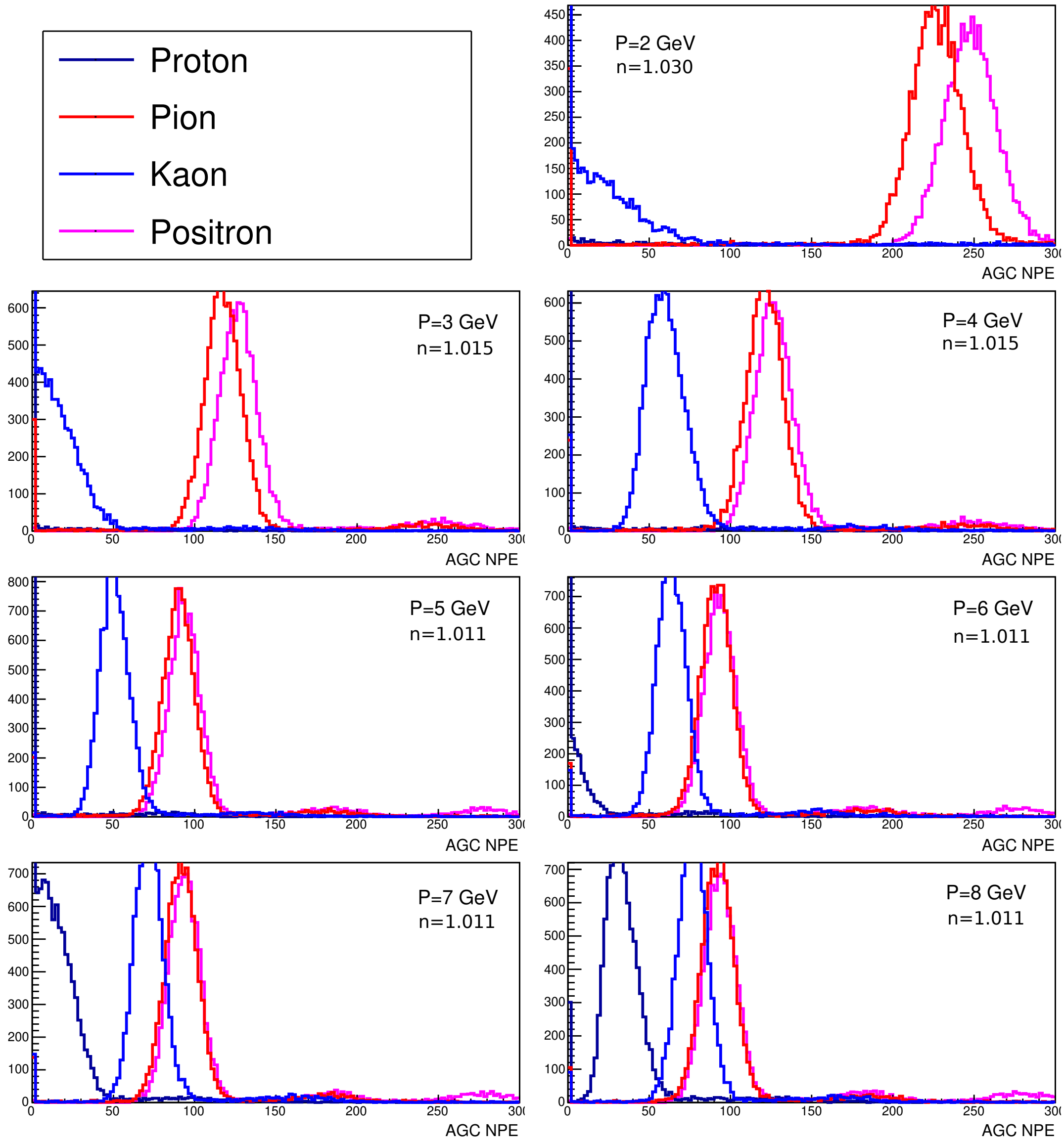

Figure 2: Simulated AGC output (NPE). The aerogel tray used for each momentum is indicated on the plot, and was chosen based on the KaonLT/PionLT run plans. Note that the kaon/proton cutoff is extremely momentum dependent, with approximate values of: 10 (P_{SHMS}=4–5 GeV), 30 (P_{SHMS}=6 GeV), 45 (P_{SHMS}=7 GeV), 55(P_{SHMS}=8 GeV).

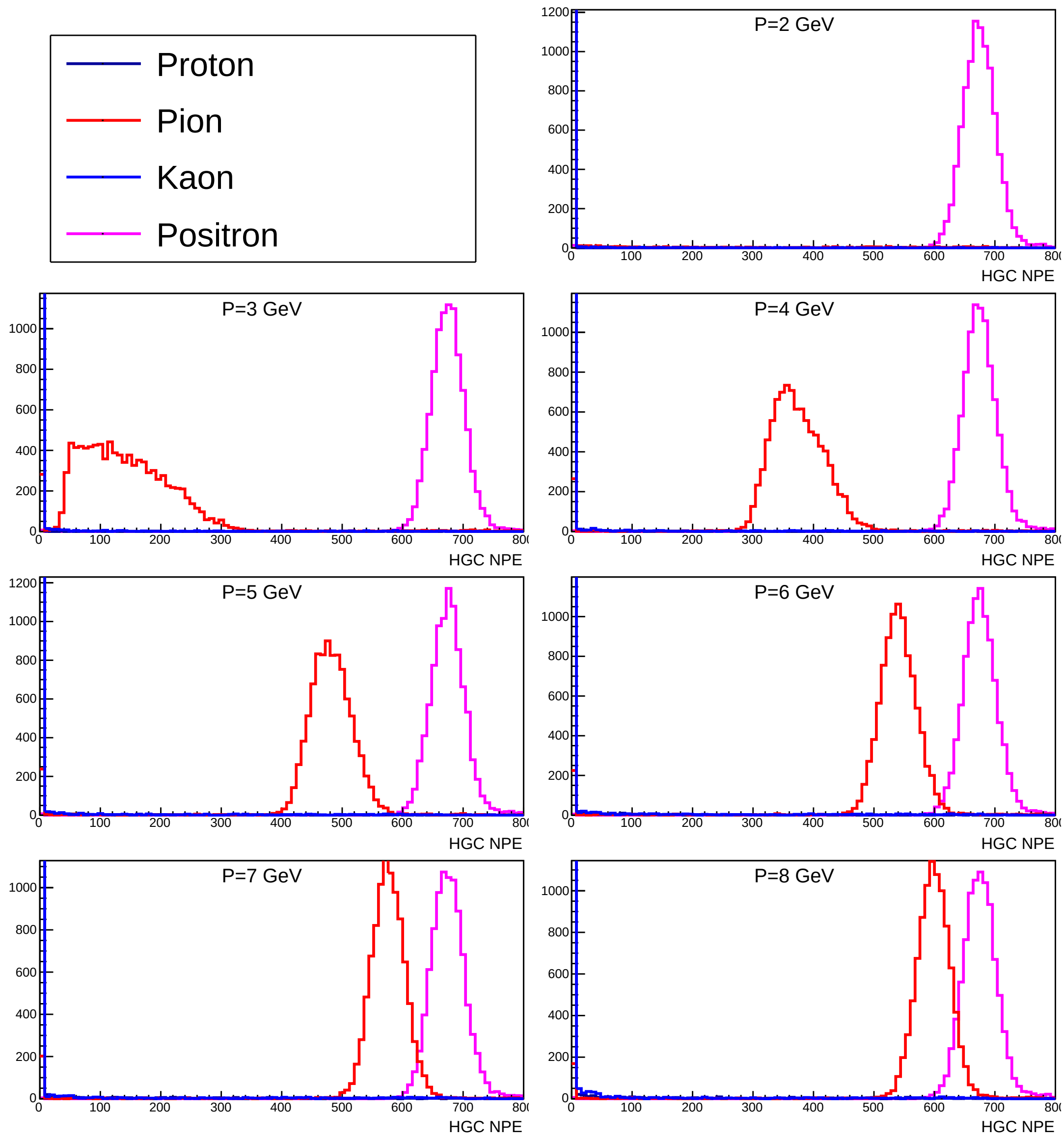

Figure 3: Simulated HGC output (NPE). Note that under 3 GeV, pions do not Cherenkov in the HGC, and at 3 GeV the average NPE for pions is lower.

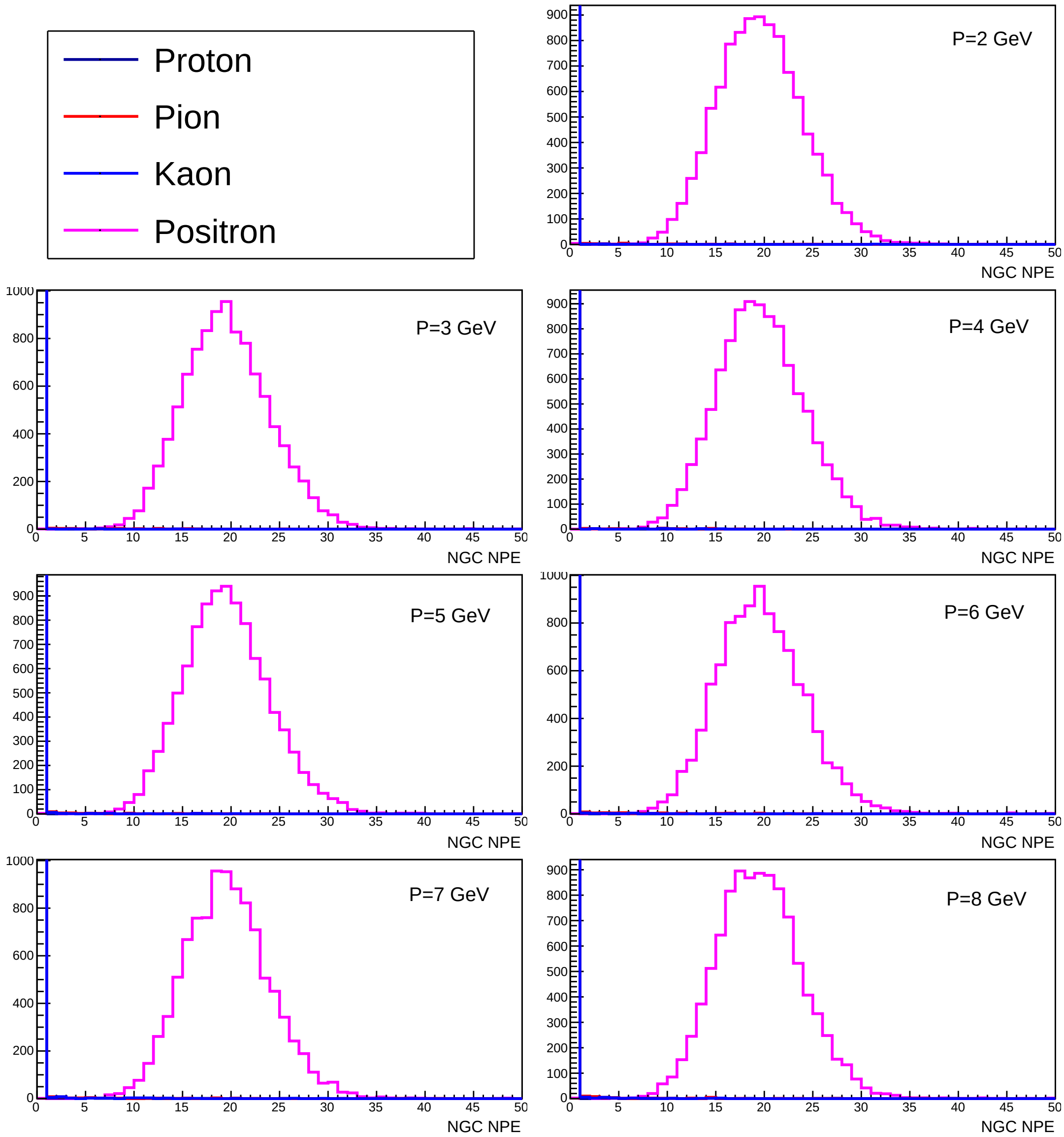

Figure 4: Simulated NGC output (NPE). A simple cut at 5 NPE should be sufficient to separate positrons from hadrons.

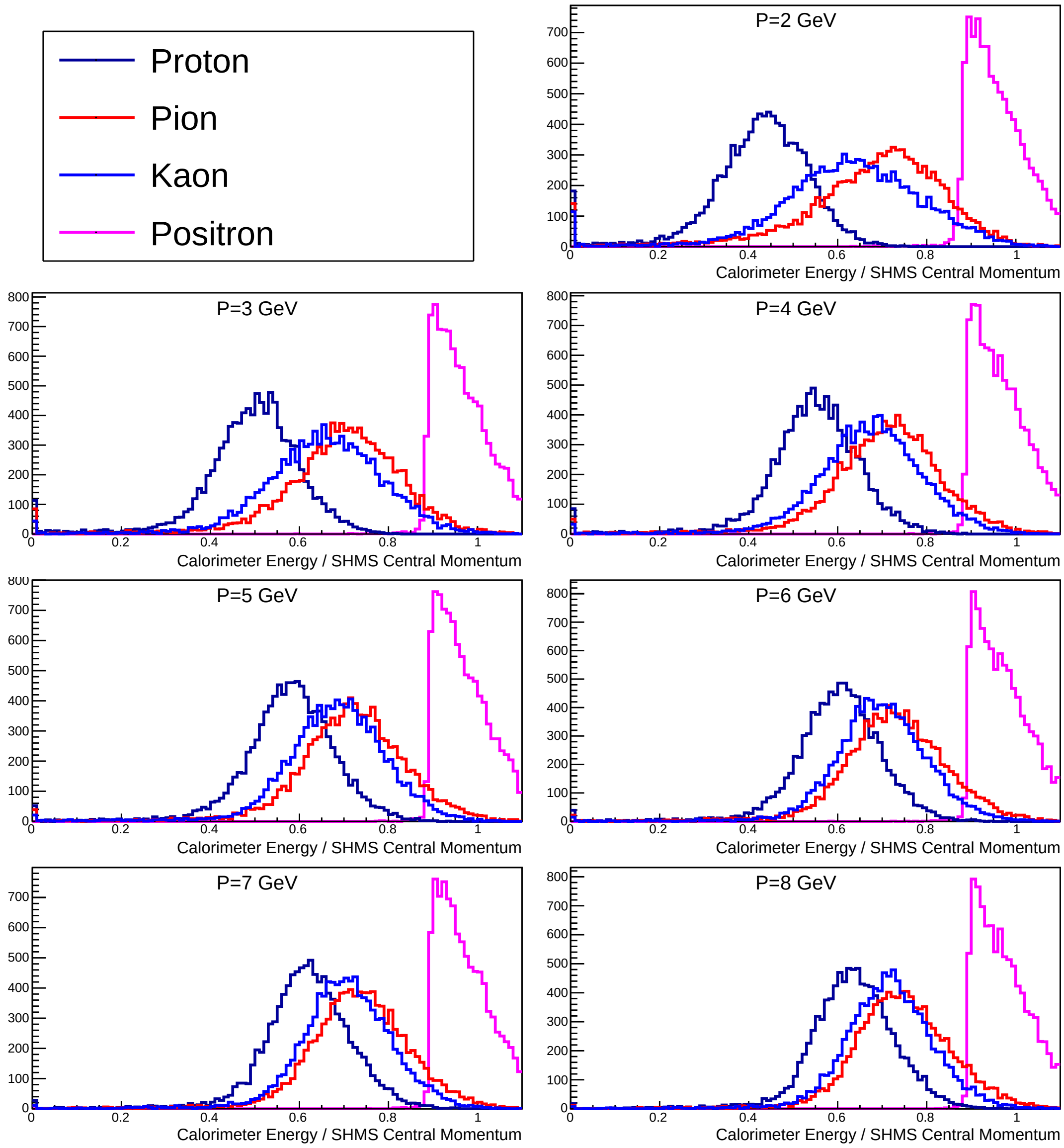

Figure 5: Simulated Calorimeter output (E_{dep}/P_{SHMS}). The calorimeter signal is not very effective for distinguishing hadrons, but positrons are consistently above a threshold of 0.8.