# GATE Simulation study

20 / 6 / 2014

Ryo Hamanishi

Contents

- GATE simulation
  - About ASCII form output

  - Getting u and v position

GATE simulation

- radial 7 < r < 19 cm
- axial (z) Length = 2 × 12 cm
  (divided by cathode)
- Electric Field in z direction 2 kV/cm
- Pad size : 3.175 x 3.175 mm^2
- Source ^{44}Sc (β^+, γ : 1.157 MeV)
- Source position
  (cylinder : 0 < r < 2.5 cm -7.5 < z < 7.5 cm)
- Drift velocity : 3 mm/usec
- PMTs
  - 2inch : 4 x 20
    (4.624 x 4.624 cm^2)
    (divide PhotoCathode by 4)
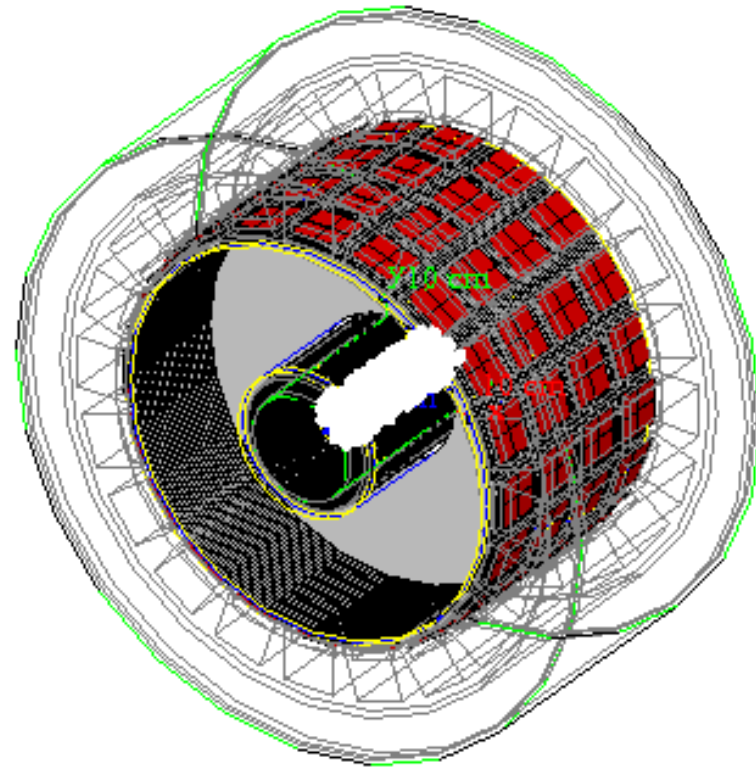
GATE ASCII form output

- To avoid memory error, I use ASCII form output to read values directly, instead of ROOT output.

- Speed down simulation, comparing with only ROOT output

- Output file size is large.
  - 2000 Bq X 5.0 sec -> about 1GB

GATE ASCII form output

- Contents of ASCII form output (Array[number of entries][25])
  - Column 1 : ID of the run (i.e. time-slice)
  - Column **2** : ID of the **event**
  - Column 3 : ID of the primary particle whose descendant generated this hit
  - Column 4 : ID of the source which emitted the primary particle
  - Column 5, 6, **7**, **8**, **9**, 10 : ID of volume attached to the "base, rsector, **module**, **submodule**, **crystal**, layer" level of the system
  - Column **11** : **Time** stamp of the hit
  - Column 12 : Energy deposited by the hit (which may be given as a percentage of the initial particle)
  - Column 13 : Range of particle which has generated the hit
  - Column **14**, **15** ,**16** : **XYZ position** of the hit in the world referential
  - Column 17 : Geant4 code of the particle which has generated the hit (11 for Electrons & 22 for Photons)
  - Column 18 : ID of the particle which has generated the hit
  - Column 19 : ID of the mother of the particle which has generated the hit
  - Column 20 : ID of the photon giving the particle which has generated the hit
  - Column 21 : Number of Compton interactions in phantoms before reaching the detector
  - Column 22 : Number of Rayleigh interactions in phantoms before reaching the detector
  - Column 23 : Name of the process which has generated the hit
  - Column 24 : Name of the last volume where a Compton effect occurred
  - Column 25 : Name of the last volume where a Rayleigh effect occurred
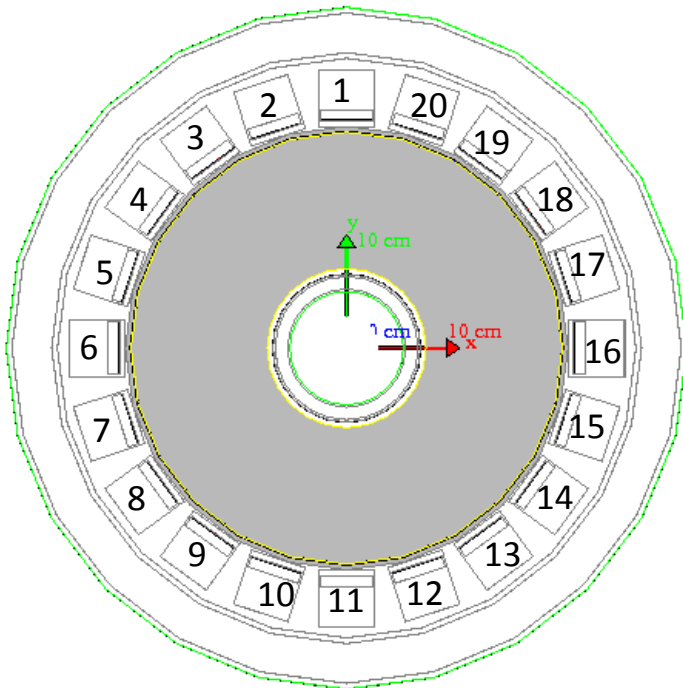
Getting u and v positions

- Method-1
  - 1. get the entry of crystalID, submoduleID and module ID (next page)
  - 2. generate the entry position from IDs

    (entry position corresponding to the IDs is

     prepared in advance)
  - 3. make two histograms (posU and posV)

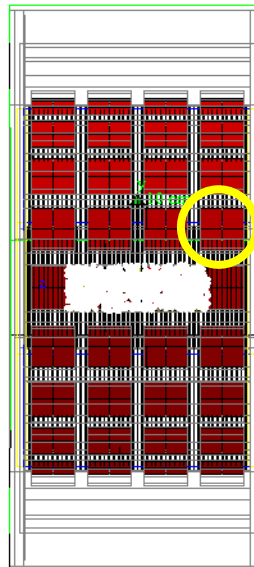* These are tests for checking distribution of entries

ID Position

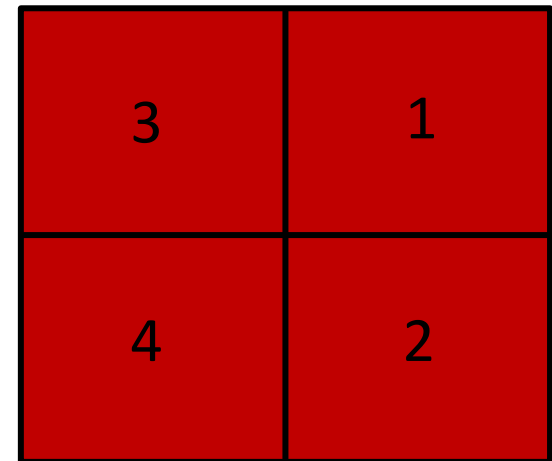- ## Three ID position

moduleID (PMTBox)

submoduleID (PMT)
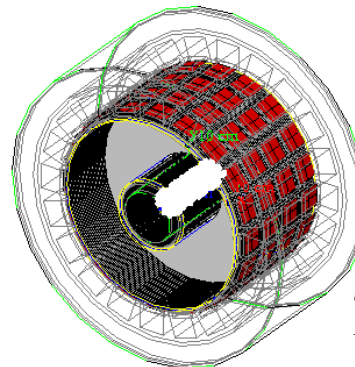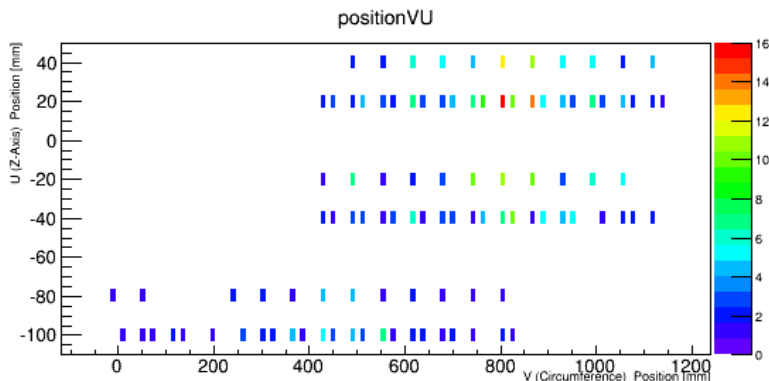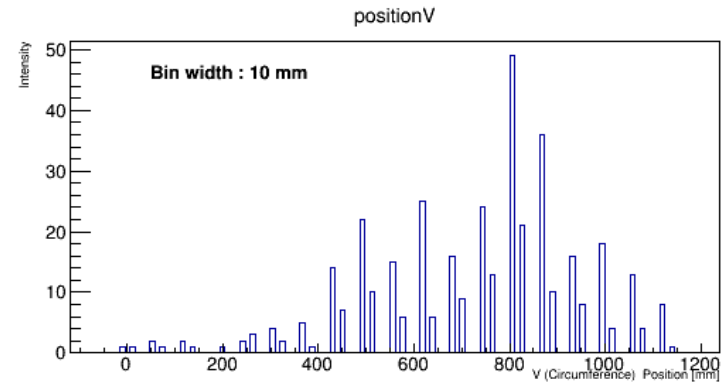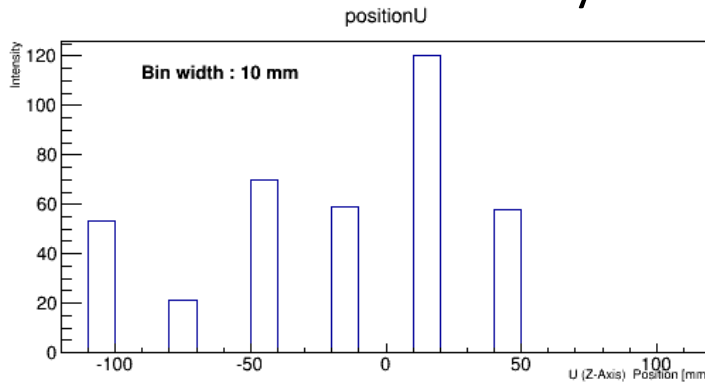
crystalID (PhotoCathode)

Getting u and v positions

- These are results of u and v positions. (Histograms)
  - event number (event : one decay process) is one.
  - Two γ (511 keV X 2) and 3rd γ (1157 keV)
  - Total number of entry is 381 in this case.

positionU

Bin width : 10 mm

positionV

Bin width : 10 mm

positionVU

u : z direction
v : circumference direction
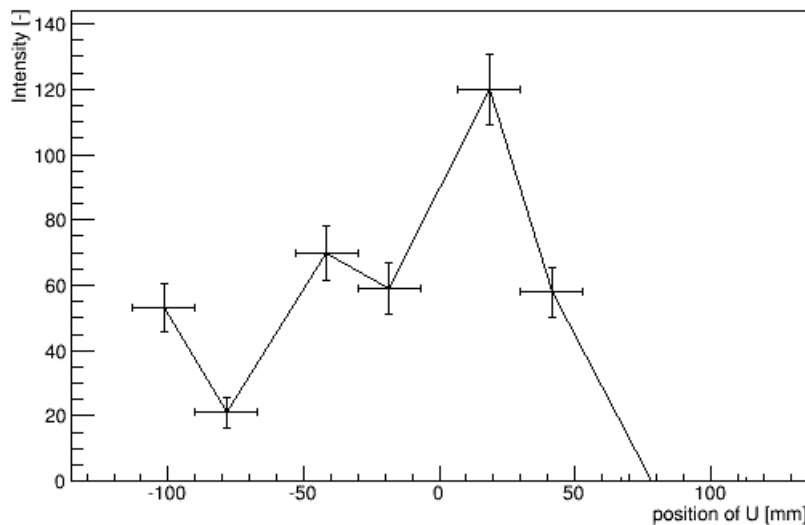(face position of PhotoCathode)

Getting u and v positions

- Method-2
  - 1. get the entry of crystalID, submoduleID and module ID (next page)
  - 2. generate the entry position from IDs
    (entry position corresponding to the IDs is
     prepared in advance)
  - 3. Get the number of entry in each position
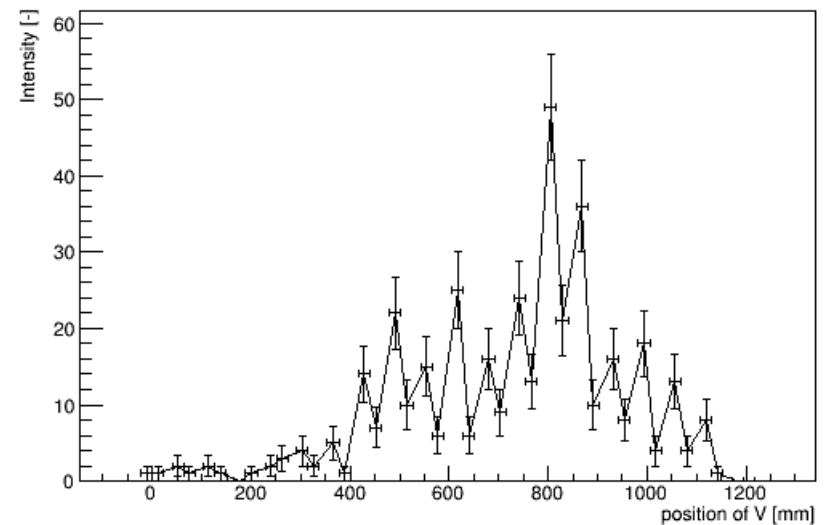 - 4. make arrays of posU(V)[i], posU(V)_count[i] and errors
  - 5. Graphed

Getting u and v positions

- These are results of u and v positions. (Histograms)
  - event number (event : one decay process) is one.
  - Two γ (511 keV X 2) and 3rd γ (1157 keV)
  - Error of X is resolution of PhotoCathode. (+/- 11.56 mm)
  - Error of Y is statistic error. ($\sqrt{N}$)

Conclusion

- Got the entry positions of photon in 2 dimensions

- Graphed 2 dimensions position

- Problem ?
  - Too low resolution to distiguish γ especially u?
  - For getting higher resolution, divide PhotoCathode by 8 (2 (v-direction) X 4(u-direction))?

Next

- Study binary output

- To get the some maximum positions, I will make the program.

- (Change geometry) ?

- Study clustering