

Diving in: How to generate MC

Qing He

Princeton University

Dayabay Collaboration

- What to generate: IBD, Radioactivity, Muon
- Where to generate.
- Monte Carlo generation process:
 - Configure generator
 - Detector simulation
 - Electronic simulation
 - Trigger simulation
 - Readout
- Analyze data, write a TTree object.

- Configure K^{40} decay in GenDecay:

```
from GenDecay.Helpers import Decay
K40=Decay()

#configure K40 decay properties
K40.decay.CorrelationTime = 1*units.second
K40.decay.ParentNuclide = 'K-40'
K40.decay.ParentAbundance = 3.01e17
K40.decay.SecularEquilibrium = True
```

- CorrelationTime: Nuclide with shorter lifetime will be correlated with parent
- ParentNuclide: K-40/K40, U238, Th232, Co60, etc.
- ParentAbundance: the number of nuclide.

Where to decay in detector

- In Gd-doped liquid scintillator

```
#configure where to generate in detector
volume = "/dd/Structure/AD/db-oil1"
K40.positioner.Volume = volume
K40.positioner.Strategy = "Material"
K40.positioner.FillMaterials = ["GdDopedLS"]
K40.positioner.Position = [0,0,0]
K40.positioner.Spread = 10*units.meter
```

- In PMT glasses (currently generate events in PMT vacuum due to technical problems)

```
#configure where to generate in detector
volume = "/dd/Structure/AD/db-oil1"
K40.positioner.Volume = volume
K40.positioner.Strategy = "VolumeType"
K40.positioner.FillVolumes = ["lvPmtHemiVacuum"]
```

- In stain-less steel

```
#configure where to generate in detector
volume = "/dd/Structure/AD/db-ade1"
K40.positioner.Volume = volume
K40.positioner.Strategy = "Material"
K40.positioner.FillMaterials = ["StainlessSteel"]
```

Register the generator

- Set the transform volume (usually same as the volume set before)

```
#transform to global values  
K40.transformer.Volume = volume
```

- Register the generator:

```
#register the generator  
import GenTools  
gtc = GenTools.Configure(genname="K40")  
gtc.register(K40)
```

Inverse Beta Decay

```
from InvBetaDecay.Helpers import Decay

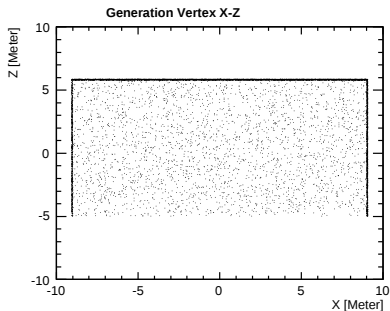
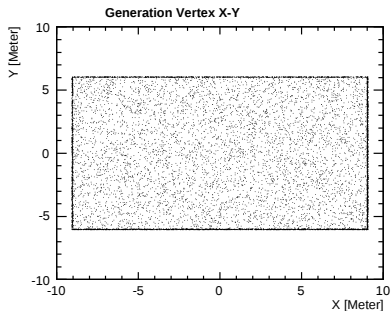
decay=Decay(name)
decay.decay.RandomSeed = "12345"
volume = '/dd/Structure/AD/db-oil1'
decay.positioner.Volume = volume
decay.positioner.Strategy = "Material"
decay.positioner.FillMaterials = ['GdDopedLS']
decay.positioner.Mode = 'Uniform'
decay.positioner.Position = [0,0,0]
decay.positioner.Spread = 10*units.m
decay.timerator.LifeTime = 78.4*units.second
decay.transformer.Volume = volume
```

```
seed = "12345"  
nevs = "2000"  
volume = "/dd/Structure/AD/db-ade1"  
musicite = 'DYB'  
music_path = 'your/path/to/music_data'  
muonGen = "Muon.exe -n "+nevs+" -s "+musicite+" -v ADE  
-seed "+seed+" -r Yes -music_dir "+music_path+" |"
```

```
from GenTools.Helpers import HepEVT  
decay = HepEVT(hepEvtDataSource = muonGen)  
decay.positioner.Volume = volume  
decay.positioner.Mode = "Relative" #this is important  
decay.positioner.Position = [0,0,0]  
decay.positioner.Spread = 10*units.meter  
decay.timerator.LifeTime = 1*units.second  
decay.transformer.Volume = volume
```

- Have to download the data file from Music simulation first
<http://dayabay.ihep.ac.cn/svn/dybsvn/data/trunk/DryRunGenerator/data/>
- Type “Muon.exe” under NuWa enviroment to check out all the options.

How Muon simulation is done



- Three options for the volume: rock, RPC, and ADE.
- Music data file provides you muon data in a thin box which surrounds the volume you specified.

Detector simulation and so on

- Most simple case:

```
#configure detector simulation
import DetSim
detsim = DetSim.Configure(site="dayabay")

#electronic simulation
import ElecSim
elecsim = ElecSim.Configure()

#trigger simulation
import TrigSim
trigsim = TrigSim.Configure()

import ReadoutSim
rosim = ReadoutSim.Configure()
```

MC truth

- Particle history:

```
detsim.historian(trackSelection="(pdg==11)",vertexSelection="(pdg==11)")
```

- Unobservable statistics:

```
params = {  
  'GD': "MaterialName == '/dd/Materials/GdDopedLS'",  
  'LS': "MaterialName == '/dd/Materials/LiquidScintillator'",  
  'lastvtx': "IsStopping == 1",  
  'Neutron': "pdg == 2112"  
}  
  
detsim.unobserver(stats=[  
  ["EDepInGdLS", "dE", "%(GD)s"%params],  
  ["EDepInLS", "dE", "%(LS)s"%params],  
  
  ["tCap", "t", "%(Neutron)s and %(lastvtx)s"%params],  
  ["xCap", "x", "%(Neutron)s and %(lastvtx)s"%params],  
  ["yCap", "y", "%(Neutron)s and %(lastvtx)s"%params],  
  ["zCap", "z", "%(Neutron)s and %(lastvtx)s"%params],  
])
```

- Variables in green squares are user defined, one have to use the same name when analyze generated MC.
- Variables shown in red ellipses are predefined.
- Consult offline manual for more information.

Write a TTree

- Tired of run again and again to make hitograms? Write a TTree object and then play with hitograms under ROOT.
- Key point to make a TTree in python: use “array”

```
from ROOT import TH1F, TH2F, TTree
from array import array
....
```

```
self.GenTime  = array('f',[0.])
self.GenPos   = array('f',3*[0.])
self.NBeta    = array('i',[0])
self.BetaKE   = array('f',100*[0.])
```

```
mytree = TTree("tree","A tree")
mytree.Branch("GenTime", self.GenTime, "GenTime/F")
mytree.Branch("GenPos", self.GenPos, "GenPos[3]/F")
mytree.Branch("NBeta", self.NBeta, "NBeta/I")
mytree.Branch("BetaKE", self.BetaKE, "BetaKE[NBeta]/F")
```

```
statsSvc = self.svc('IStatisticsSvc','StatisticsSvc')
status = statsSvc.put("/file/tree/data",mytree)
```

Example to run

- Make sure you have updated to the latest release

```
cd ~NuWa-trunk/tutorial  
svn update
```

- Copy the simple scripts to your working directory

```
~NuWa-trunk/tutorial/GenerateMC/python/GenerateMC
```

- Generate MC after NuWa environment is set (in your working directory):

```
nuwa.py -n 1000 -o test.root -m "SimpleK40"
```

- Analyze:

```
nuwa.py -n -1 -m "WriteTree output.root" test.root
```

- Plot hitograms under ROOT:

```
TFile f("output.root");  
TTree* t = (TTree*) f.Get("stats/tree/data");  
t->Draw("GenPos [0] : GenPos [1]");  
t->Draw("BetaKE");  
.....
```

- Common place for Radioactivity & IBD rates:

`~NuWa-trunk/dybgaudi/Production/SourceRate/python/SourceRate`

- Examples for using Sim15 package:

`~NuWa-trunk/dybgaudi/Production/MDC09b/python/MDC09b/runIBD15`

- Excellent place for other examples:

`~NuWa-trunk/dybgaudi/Production`