Some bugs about internal conversion transition in the Neutron HP (Geant4.9.2 + G4NDL3.13)

&

An approach to “E0 692keV transition of 72Ge” issue

Chao Zhang, Dongming Mei
Feb. 25, 2009
Chao.Zhang@usd.edu
Dongming.Mei@usd.edu

Internal Conversion in Geant4

- Internal conversion code is included in Geant4(There are some bugs with it though. Please see our proposed correction for reference). However, sometimes we are unable to generate such process duo to the well-known insufficient cross-section data information.
- Specifically, for the neutron data of element 72Ge, it doesn't include γ /e- transition data in each energy level(Data Type = 12). So “E0 692keV transition” process is missing.
- We calculate this part of data by using Talys-1.0 simulation, and add them to Geant4 neutron data so that E0 692keV transition can be activated in Geant4 simulation.

Bug Report and Proposed Correction

==========

Geant4.9.2 bug report and proposed correction

Feb. 25, 2009 Chao Zhang (Chao.Zhang@usd.edu), Dongming Mei(Dongming.Mei@usd.edu)

==========

****

In file: $G4Install/source/processes/hadronic/models/neutron_hp/src/G4NeutronHPInelasticCompFS.cc

++++++++++

Bug 1

Identified error: besides photons, there are electron generated through internal conversion transition.

Line 578 - 581:

```
578       theSec = new G4DynamicParticle;
579//       theSec->SetDefinition(G4Gamma::Gamma());                    //proposed correction: turn this line off
          theSec->SetDefinition(thePhotons->operator[](i)->GetDefinition()); // and add this line for all transition
580       theSec->SetMomentum(thePhotons->operator[](i)->GetMomentum());
581       theResult.AddSecondary(theSec);
```

****

In file: $G4Install/source/processes/hadronic/models/neutron_hp/src/G4NeutronHPPhotonDist.cc

++++++++++

Bug 1

Identified error: for electron, the total energy E_tot = ΔE + m_e – E_b where ΔE is energy level difference and
E_b is the binding energy in the atomic orbital.

Line 542 - 550:

```
542    G4double totalEnergy = theBaseEnergy - theLevelEnergies[it];
543    G4ReactionProduct * theOne = new G4ReactionProduct;
544    theOne->SetDefinition(G4Gamma::Gamma());
545    random = G4UniformRand();
546    if(theInternalConversionFlag==2 && random>thePhotonTransitionFraction[it])
547    {
548      theOne->SetDefinition(G4Electron::Electron());
       totalEnergy += G4Electron::Electron()->GetPDGMass();    //proposed correction: add this line for electron
549    }
550    theOne->SetTotalEnergy(totalEnergy);
```

++++++++++

Bug 2

Identified error: isotropic distribution of theta angle, uniformly. (Identified by Jason Detwiler from LBL)

Line 571 – 574:

```
571       // isotropic distribution
572//       G4double theta = pi*G4UniformRand();                  // proposed correction: turn this line off
          G4double theta = std::acos(2.*G4UniformRand()-1.);   // and add this line for uniform distribution
573       G4double phi = twopi*G4UniformRand();
574       G4double sinth = std::sin(theta);
```

The kinetic energy of conversion electron, $T_e = \Delta E - E_b$. The emitted gamma-ray interacts with one of the innermost orbital electrons, as a result, the orbital electron is ejected. Subsequently, an outer orbital electron drops to fill in the vacancy with the emission of characteristic X-ray. Such process should have been included in the code. But this is a complicated process that involves in using binding energy which needs to be edited for all elements in G4 format. For the sake of simplicity, we just treat $T_e = \Delta E$ temporarily here because the X-ray and the conversion electron will contribute the total energy deposition that equals $T_e = \Delta E$ (germanium detector only). More general solution needs to be done for other materials.

++
Bug 3
Identified error: for electron, total energy is not equal to total momentum.
Line 553 - 559:

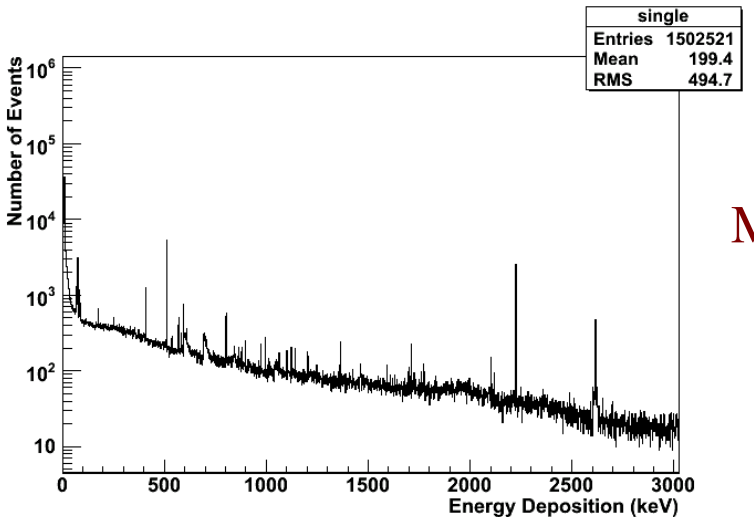
```
553     G4double costheta = 2.*G4UniformRand()-1;
554     G4double theta = std::acos(costheta);
555     G4double phi = twopi*G4UniformRand();
556     G4double sinth = std::sin(theta);
557//     G4double en = theOne->GetTotalEnergy();       //proposed correction: turn off this line
        G4double en = theOne->GetTotalMomentum();  //and add this line
558     G4ThreeVector temp(en*sinth*std::cos(phi), en*sinth*std::sin(phi), en*std::cos(theta) );
559     theOne->SetMomentum( temp ) ;
```
++
Bug 4
Identified error: the same as Bug 2
Line 572 - 577:

```
572     G4double theta = pi*G4UniformRand();
573     G4double phi = twopi*G4UniformRand();
574     G4double sinth = std::sin(theta);
575//     G4double en = theOne->GetTotalEnergy();       //proposed correction: turn off this line
        G4double en = theOne->GetTotalMomentum();   //and add this line
576     G4ThreeVector tempVector(en*sinth*std::cos(phi), en*sinth*std::sin(phi), en*std::cos(theta) );
577     theOne->SetMomentum( tempVector ) ;
```
++
Bug 5
Identified error: the same as Bug 2
Line 593 - 598:

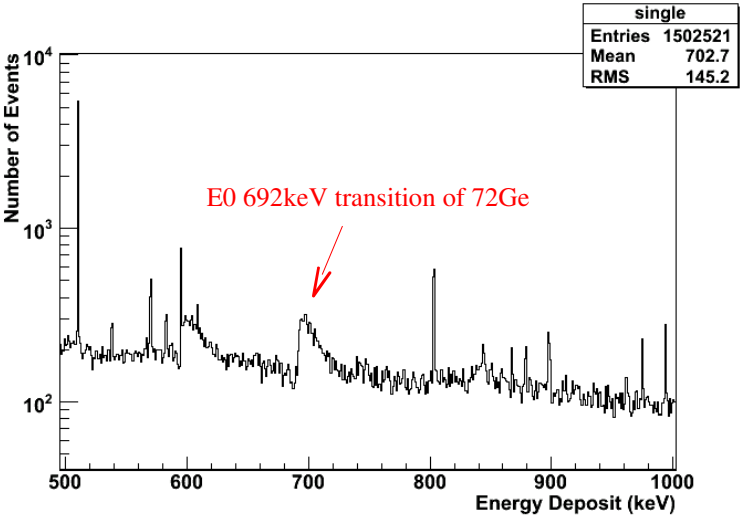
```
593     G4double theta = std::acos(cosTh);
594     G4double phi = twopi*G4UniformRand();
595     G4double sinth = std::sin(theta);
596//     G4double en = theOne->GetTotalEnergy();       //proposed correction: turn off this line
        G4double en = theOne->GetTotalMomentum();  //and add this line
597     G4ThreeVector tempVector(en*sinth*std::cos(phi), en*sinth*std::sin(phi), en*std::cos(theta) );
598     theOne->SetMomentum( tempVector ) ;
```
++
Bug 6
Identified error: the same as Bug 2
Line 611 - 616:

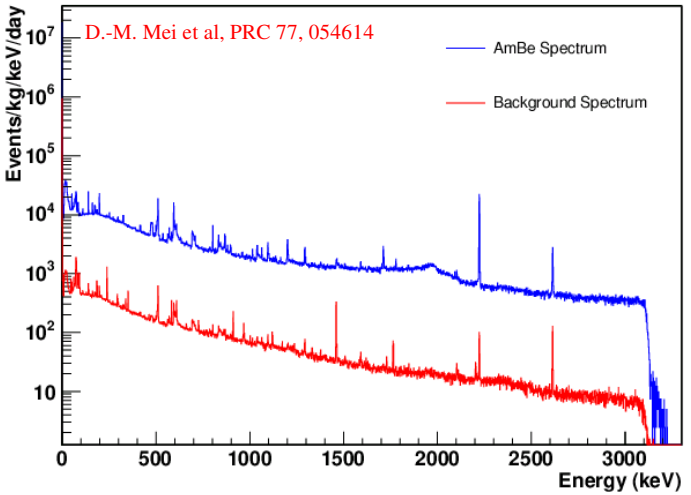
```
611     G4double theta = std::acos(costh);
612     G4double phi = twopi*G4UniformRand();
613     G4double sinth = std::sin(theta);
614//     G4double en = theOne->GetTotalEnergy();       //proposed correction: turn off this line
        G4double en = theOne->GetTotalMomentum();  //and add this line
615     G4ThreeVector tmpVector(en*sinth*std::cos(phi), en*sinth*std::sin(phi), en*costh );
616     theOne->SetMomentum( tmpVector ) ;
```

E0 692keV Transition of 72Ge

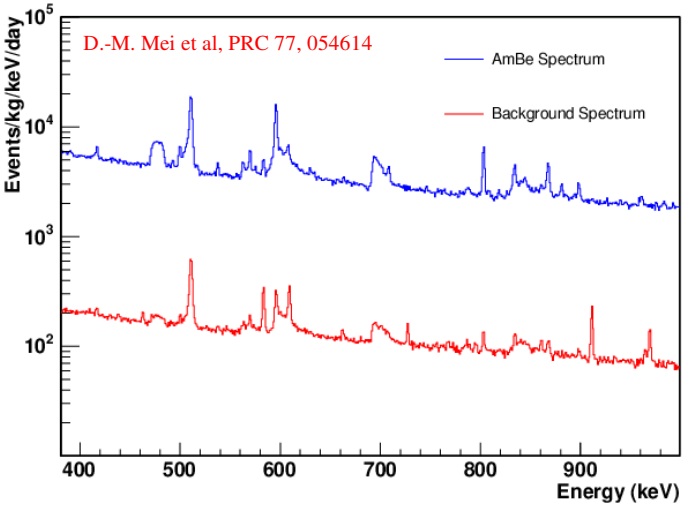


MC Result





Experimental Data



- Append our data “AddedData.txt” to the tail of $G4NEUTRONHPDATA/Inelastic/F01/32_72_Germanium
- AmBeClover simulation is conducted. The MC result has a good agreement with the experimental data.