

muX Monte Carlo

Mainz edition

Frederik Wauters
Johannes Gutenberg University Mainz

Monte Carlo for muX

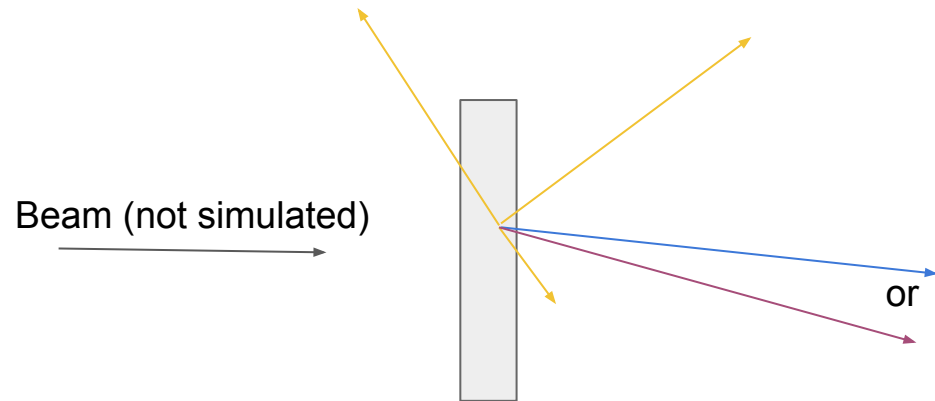
MC for muX: diverse / individual / poor overview/access

- Beam in gas cell (Andreas)
- Detector response (Andreas, Frederik)
- Transfer simulations (Adam)
- Beam + transfer (Jonas)
- Beam in Zn (Frederik)
- HPGe array simulation with full cascade (Frederik)
- (I forgot)

Simulating array response

Event generator:

- Starting point: muon stop (no beam)
- When simulating realistic coincidences, one needs
 - Full cascade **X-rays**
 - Michel decay **electron**
 - Capture **neutron** (γ 's)



Simulating array response

Event generator:

- Starting point: muon stop (no beam)
- When simulating realistic coincidences, one needs
 - Full cascade **X-rays**
 - Michel decay **electron**
 - Capture **neutron** (γ 's)

To simulate a realistic cascade, one needs the population at level \mathbf{N} , and all transition rates $(nL^j)_i \rightarrow (nL^j)_f$ with $n_i, n_f \leq N$

➤ CASCADE code, energies from MUON

Simulating array response

15, 0 - 14, 0	0.000	3.787	1.818	2.381E+14	15	0	1	10	3	7	14.027976	7.5376E-15
M-2T	2.1246E+14	M-3T	2.5690E+13		*							
15, 0 - 13, 0	0.000	1.893	1.208	5.122E+13	15	1	1	10	4	7	14.027976	5.4109E-15
M-2T	4.6357E+13	M-3T	4.8663E+12		*							
15, 0 - 12, 0	0.000	1.355	0.925	1.173E+13	15	0	1	10	3	5	14.038115	5.6532E-15
M-2T	1.1734E+13				*							
15, 0 - 11, 0	3.965	1.062	0.746	4.005E+13	15	1	1	10	3	5	14.038115	1.0002E-10
M-1T	3.6704E+13	M-2T	3.3413E+12		***							
15, 0 - 10, 0	1.488	0.863	0.617	1.127E+13	15	0	1	10	2	5	14.038115	1.6143E-10
M-1T	1.1272E+13				***							
15, 0 - 9, 0	1.009	0.714	0.516	3.592E+12	15	1	3	10	4	9	14.040295	9.0182E-15
M-1T	3.5921E+12				*							
15, 0 - 8, 0	0.760	0.593	0.432	***NO RATE***	15	2	3	10	5	9	14.040295	5.5189E-15
15, 0 - 7, 0	0.592	0.491	0.359	***NO RATE***	*							
...					15	6	13	10	9	19	14.041054	8.6233E-17
15, 0 - 1, 0	0.040	0.037	0.028	***NO RATE***	*							
15, 0 - 14, 1	0.040	3.818	1.824	3.026E+14	15	5	11	10	8	17	14.041545	5.5924E-16
D-RT	4.8394E+11	D-2T	2.4368E+14	D-3T	5.8447E+13	*						
15, 0 - 13, 1	0.040	1.897	1.210	2.909E+13	15	6	11	10	9	17	14.041545	7.5620E-17
D-RT	5.3320E+11	D-2T	2.2708E+13	D-3T	5.8446E+12	*						
15, 0 - 12, 1	0.040	1.358	0.926	8.458E+12	15	7	15	10	9	19	14.041696	8.9306E-13
D-RT	5.8734E+11	D-2T	5.6547E+12	D-3T	2.2164E+12	**						
15, 0 - 11, 1	4.030	1.063	0.747	2.065E+13	15	5	9	10	8	15	14.042030	4.8016E-16
D-RT	6.6205E+11	D-1T	1.7071E+13	D-2T	2.2215E+12	D-3T	7.0003E+11					
15, 0 - 10, 1	1.493	0.864	0.618	1.061E+13	15	4	9	10	7	15	14.042030	1.9162E-15
...					*							
					15	8	17	10	9	19	14.042195	4.3187E-09

For each Ni to Nf one file with all transition, no radiative BR info

For each N one file with radiative BR

This should be done less messy!

Simulating array response

Event generator:

- Starting point: muon stop (no beam)
- When simulating realistic coincidences, one needs
 - Full cascade **X-rays**
 - Michel decay **electron**
 - Capture **neutron** (γ 's)

To simulate a realistic cascade, one needs the population at level **N**, and all transition rates $(nL^j)_i \rightarrow (nL^j)_f$ with $n_i, n_f \leq N$

- CASCADE code, energies from MUON
- Parse this output with python script to C++ code, generating an object for each level, containing all transitions from that level

```
class Level
{
public:

    Level();
    Level(unsigned int n_val, unsigned int l_val, float j_val);
    Level(unsigned int n_val, unsigned int l_val, float j_val, std::string name_value);

    ~Level();

    void SetEnergy(G4double value) {E = value;};
    G4double GetEnergy() { return E; };
    std::string GetName() { return name; };
    ....
    unsigned int Get_n() { return n; }
    unsigned int Get_l() { return l; }
    float Get_j() { return j; }
    Level* GetTransition(G4double* e, G4bool* radiative); //returns next level

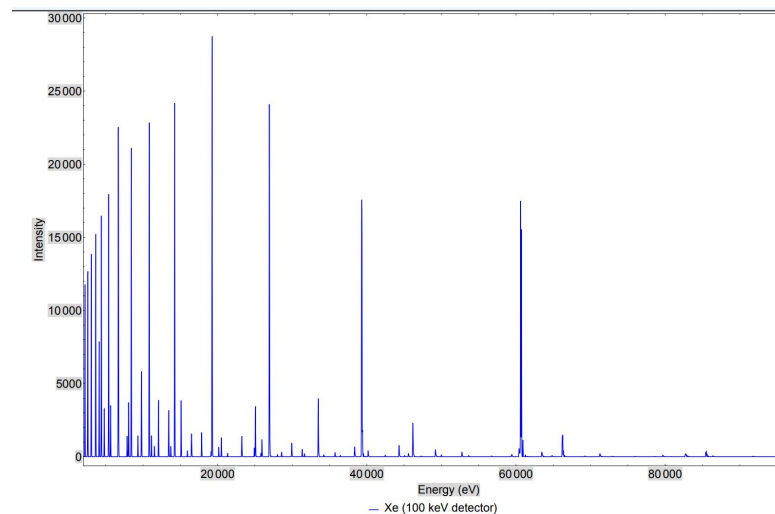
private:

    std::string name;
    unsigned int n;
    unsigned int l;
    float j;
    G4double E; //energy of level
    std::map<Level*, transition_param_t> transitions; //radiative transitions with transition
    probabilities
    std::vector<Level*> levels; //handy to keep track of the levels available
};
```

Simulating array response

Event generator:

- Starting point: muon stop (no beam)
- When simulating realistic coincidences, one needs
 - Full cascade X-rays
 - Michel decay electron
 - Capture neutron (γ 's)



To simulate a realistic cascade, one needs the population at level \mathbf{N} , and all transition rates $(nL^j)_{i \rightarrow f}$ with $n_i, n_f \leq N$

- CASCADE code, energies from MUON
- Parse this output with python script to C++ code, generating an object for each level, containing all transitions from that level
- Paul Indelicato's code?

Adding the Auger would mean a big change in the angular part of the mcdfgme code to have generalized matrix elements and arbitrary recoupling between bound electrons, exotic particles and Auger electrons. Not impossible but a major project (the code has ~250000 lines, 388 subroutines....)

??

Simulating array response

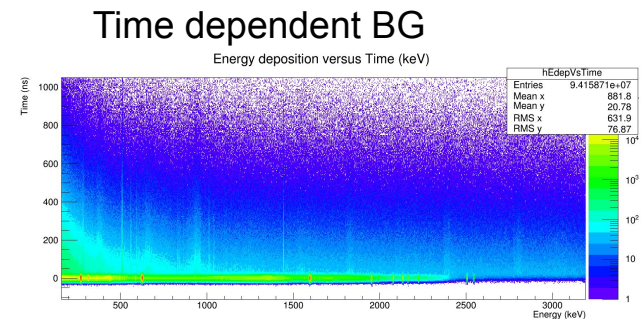
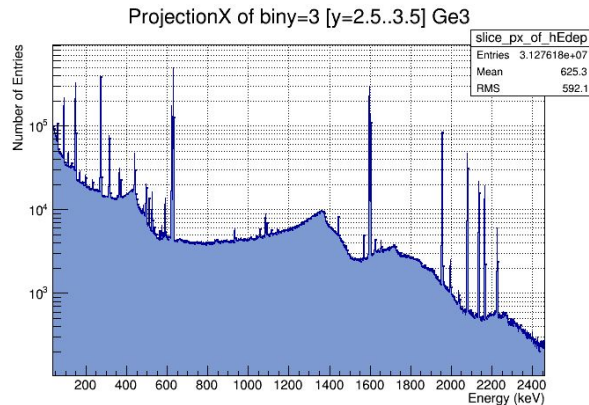
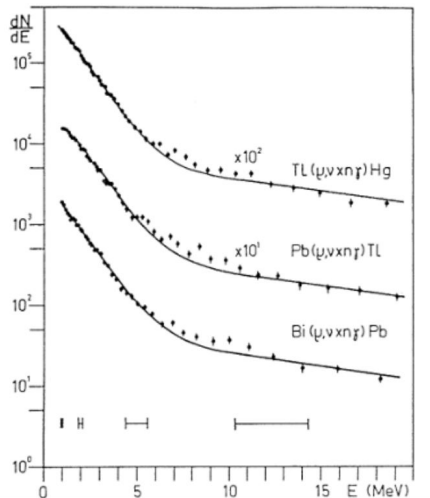
Event generator:

- Starting point: muon stop (no beam)
- When simulating realistic coincidences, one needs
 - Full cascade **X-rays**
 - Michel decay **electron**
 - Capture **neutron** (γ 's)

To simulate a realistic cascade, one needs the population at level N , and all transition rates $(nL^j)_i \rightarrow (nL^j)_f$ with $n_i, n_f \leq N$

- CASCADE code, energies from MUON
- Parse this output with python script to C++ code, generating an object for each level, containing all transitions from that level
- Generate cascade + electron from Michel spectrum or generic neutron spectrum (better for high Z then low Z)

D.F. Measday / Physics Reports 354 (2001) 243-409



Simulating array response

Physics list:

- `G4VModularPhysicsList* physicsList = factory.GetReferencePhysList("QGSP_BIC_PEN");`
`physicsList->SetDefaultCutValue(0.02*mm);`

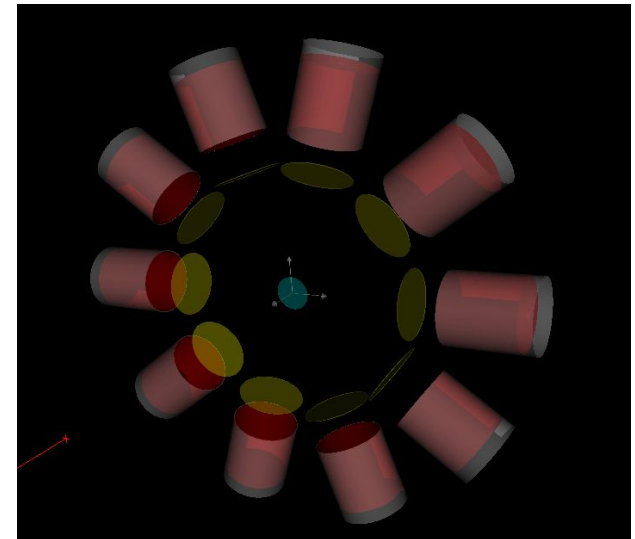
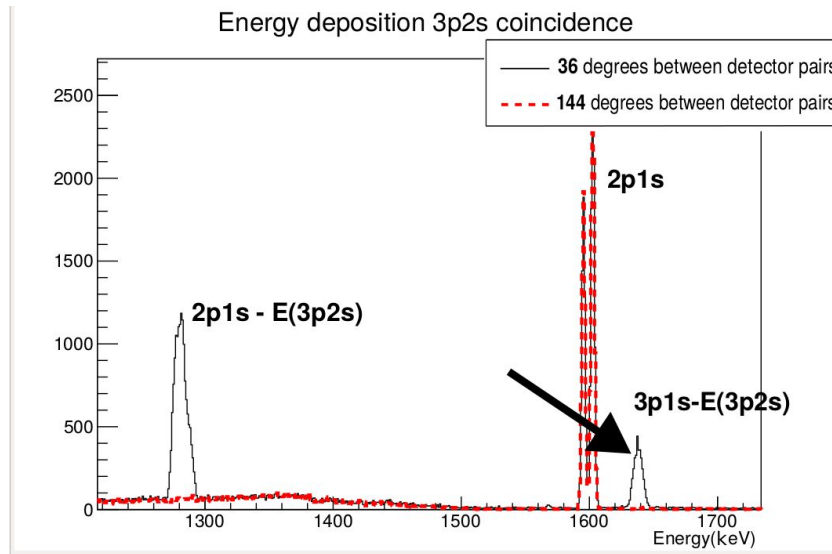
→ Penelope EM models

Not optimized for neutrons yet?

Simulating array response

Status and outlook:

- Used this code to get a handle on 3p1s + 2s1s pile up



- TODO: Full cascade (some things are still missing)
- TODO: Miniball detectors geometry.

Andreas: G4 beamline

- Stopping distribution.
- Limitations?

Jonas:

- Beam + transfer
- Used this code to get a handle on 3pIs + 2sIs pile

