Application of GEANT4's Importance Biasing in radiogenic background simulations

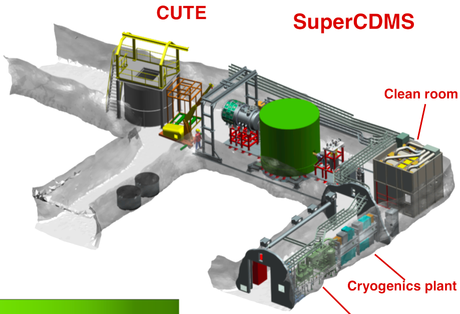
VIEWS24, Vienna, Austria

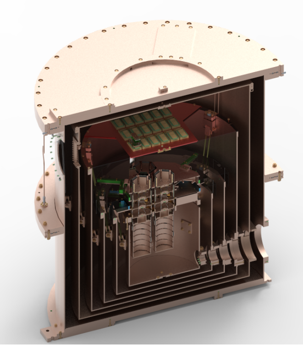
Birgit Zatschler
on behalf of the SuperCDMS collaboration

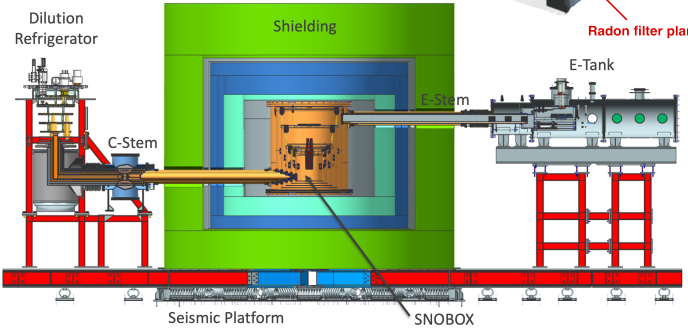
26th April 2024

SuperCDMS experiment at SNOLAB

- The **S**uper **C**roygenic **D**ark **M**atter **S**earch experiment is aiming for direct detection of DM interactions.
- Complementary technique using 18 Ge and 6 Si detectors under cryogenic conditions.
- Commissioning planned for 2024.



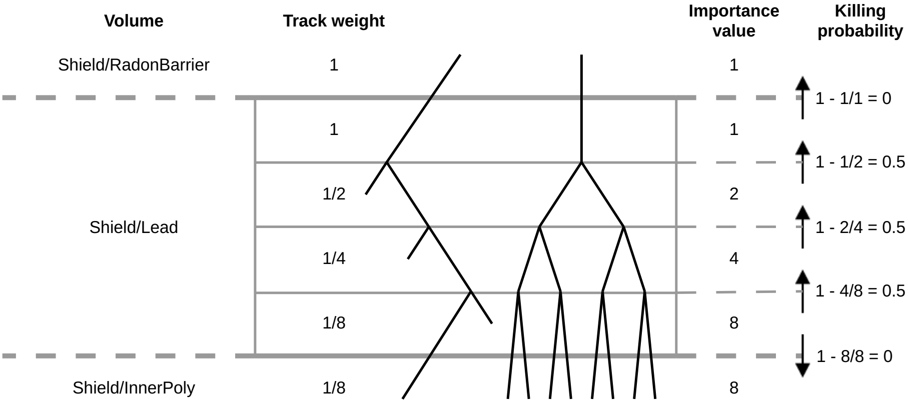




SuperCDMS lead shield simulation in SuperSim

- Graded lead shield:
  - Ultra low background (ULB):
    0 - 1 cm – 0.3 Bq/kg
  - Low background (LB):
    1 - 10 cm – 21 Bq/kg
  - Regular background (RB):
    10 - 20 cm – 157 Bq/kg
- Simulating radioactive contaminations inside the lead shield (e.g. ^{210}Pb) or in volumes surrounding the lead shield (e.g. ^{40}K in Al Radon Barrier) would consume $\sim \mathcal{O}$(10k) of cpu years.

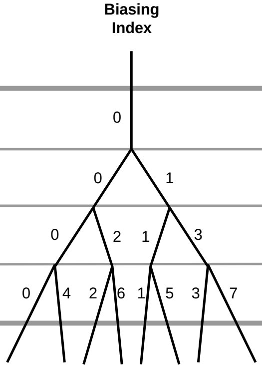
Importance Biasing



- GEANT4's importance biasing splits (kills) particles going to (away from) the detectors.
- The split particle is an identical copy of the original particle, both their weights are halved.
- A backwards going particle is either killed or its weight is adjusted.
- Only one particle type is biased, i.e. in our case gammas inside the lead shield.

Event Numbering – Biasing Index

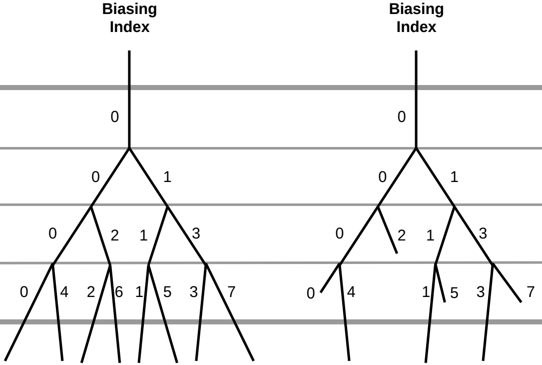
Importance value	Track weight	Biasing Index
1	1	0
1	1	0
2	1/2	+1
4	1/4	+2
8	1/8	+4
8	1/8	



- Distinguish between different split-track topologies within a single generated event.
- The *Biasing Index* of a split particle is increased according to the original particle's *Biasing Index* and the importance value of the just entered importance layer.

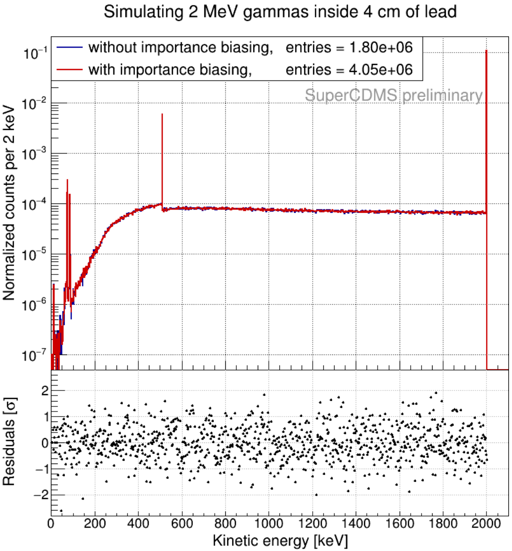
Event Numbering – Biasing Index

Importance value	Track weight	Biasing Index
1	1	0
1	1	0
2	1/2	+1
4	1/4	+2
8	1/8	+4
8	1/8	



- Distinguish between different split-track topologies within a single generated event.
- The *Biasing Index* of a split particle is increased according to the original particle's *Biasing Index* and the importance value of the just entered importance layer.
- Sum up detector hits for same *Biasing Index* and weight with track weight.

Validation of Importance Biasing and Biasing Index

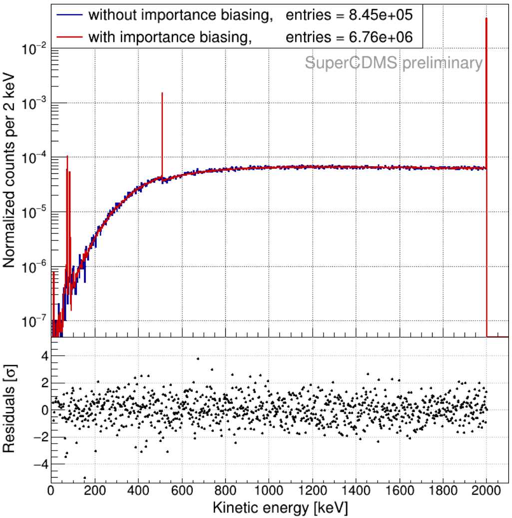


2 MeV gammas emitted inside the lead shield

- Lead shield thickness shrunk to 4 cm to achieve sufficient statistics without Importance Biasing.
- Record all particle tracks leaving the lead shield.
- Biasing simulation run with 4 importance layers of 1 cm thickness each.

Validation of Importance Biasing and Biasing Index

Simulating 2 MeV gammas passing through 4 cm of lead



2 MeV gammas started outside of the lead shield

- Lead shield thickness shrunk to 4 cm to achieve sufficient statistics without Importance Biasing.
- Record all particle tracks leaving the lead shield.
- Biasing simulation run with 4 importance layers of 1 cm thickness each.

Efficiency of Importance Biasing – 2 MeV gammas

- Shoot gammas onto Lead Shield and count events leaving it with/without Importance Biasing.
- Efficiency improvement strongly depends on number and thickness of importance layers.

Lead Shield thickness [cm]	Number of imp. layers	Efficiency gain	**Efficiency gain normalized on runtime**
5	4	8.00	2.78
10	8	128.2	38.4
20	16	33454	1023.1

- Importance layer thickness needs to be adjusted depending on the gamma energy.
- Very tricky choice for isotopes emitting different gamma energies, especially in decay chains.
- SuperCDMS' backgrounds include e.g. ^{210}Pb emitting a 46 keV γ and ^{232}Th with a 2.6 MeV γ.
- Determine optimal importance biasing settings with simulation studies for effective simulations.

Efficiency of Importance Biasing – isotopes and decay chains

- Contaminate Radon Barrier and count events leaving Lead Shield w/wo Importance Biasing.
- **Idealistic simulations without any stem holes** in 20 cm thick lead shield with 16 imp. layers.

isotope or decay chain	Efficiency gain	**Efficiency gain normalized on runtime**
^{40}K	28177	22697
^{226}Ra	33211	21104
^{232}Th	33708	17173
^{238}U	40398	27692

- Closed shielding without holes is ideal for Importance Biasing simulations.

Efficiency of Importance Biasing – isotopes and decay chains

- Contaminate Radon Barrier and count events leaving Lead Shield w/wo Importance Biasing.
- **Realistic simulations with stem holes** in 20 cm thick lead shield with 16 imp. layers.

isotope or decay chain	Efficiency gain	**Efficiency gain normalized on runtime**
^{40}K	10988	905.7
^{226}Ra	1705	99.1
^{232}Th	2252	104.6
^{238}U	918	554.1

- Stems holes decrease efficiency improvement for simulations with Importance Biasing.

Implementation – Physics list

Implementation in SuperSim based on GEANT4's example *biasing/B02* using modular physics lists and geometry sampler:

```
G4RunManager* rm = G4RunManager::GetRunManager()
G4VModularPhysicsList* pl = rm->GetUserPhysicsList();

G4GeometrySampler* pgs = new G4GeometrySampler("ImportanceBiasing","gamma");
pgs->SetParallel(true);

G4VPhysicsConstructor* pbias = new G4ImportanceBiasing(pgs,"ImportanceBiasing");
pbias->SetVerboseLevel(pl->GetVerboseLevel());
pl->RegisterPhysics(pbias);

G4VPhysicsConstructor* pworld = new G4ParallelWorldPhysics("ImportanceBiasing");
pworld->SetVerboseLevel(pl->GetVerboseLevel());
pl->RegisterPhysics(pworld);
```

Caveat: Only one particle type can be biased at a time. Construct your own generic biasing for multiple particles.

Implementation – Parallel World

Construct importance layers as nested physical volumes in a parallel world and assign importance values in an importance store:

```
std::vector<G4VPhysicalVolume*> iStorePhysicalVolumes;
for (G4int i=0; i<biasPVs; i++) {
G4VPhysicalVolume* biasPV = new G4PVPlacement(...);
iStorePhysicalVolumes.push_back(biasPV);}

G4VPhysicalVolume* worldPV = G4VUserParallelWorld::GetWorld();
G4IStore* istore = G4IStore::GetInstance("ImportanceBiasing");

G4double imp = 1;
istore->AddImportanceGeometryCell(imp, *worldPV, 0);

G4int numBiasPV = iStorePhysicalVolumes.size();

for (G4int cell=0; cell<numBiasPV; cell++) {
  imp = std::pow(2.0,cell);
  istore->AddImportanceGeometryCell(imp, *iStorePhysicalVolumes.at(cell), 0);}
```

Advantage: GEANT4 does the splitting and aborting of particle tracks automatically according to the assigned importance values.

Implementation – Biasing Index

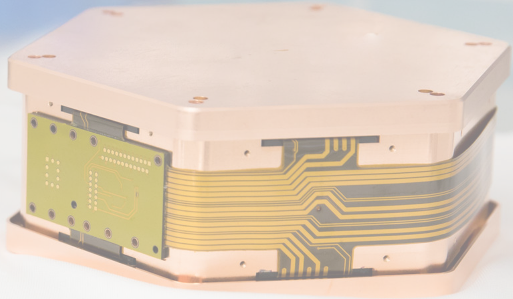
Store Biasing index in G4VUserTrackInformation and assign it in your G4UserTrackingAction::PostUserTrackingAction(G4Track* track):

```
TrackInfo* pInfo = (TrackInfo*)(track->GetUserInformation());
G4double biasingIndexParent = pInfo->GetBiasingIndex();

G4TrackVector* secondaries = fpTrackingManager->GimmeSecondaries();

for (size_t i=0; i < secondaries->size(); ++i) {
  G4Track* daughter = (*secondaries)[i];
  if(daughter->GetCreatorProcess()->GetProcessName()== "ImportanceProcess") {

    ParallelWorld* impWorld = GetParallelWorld("ImportanceBiasing");
    G4VPhysicalVolume* impPV = GetPhysicalVolume(daughter->GetPosition(), impWorld);

    G4double impValue = impWorld->GetImportanceValue(impPV);
    G4double biasingIndexDaughter = biasingIndexParent+impValue/2.;

    TrackInfo* dInfo = daughter->SetUserInformation(new TrackInfo());
    dInfo->SetBiasingIndex(biasingIndexDaughter);}}
```

Biasing index is handled like other track information such as position and can be added to the simulation output for analysis processing.

Summary & Outlook

- Sufficient statistics are necessary to model background contributions with simulations.
- Importance biasing can improve simulation efficiency significantly.
- Choice of importance biasing settings depends on multiple parameters.
- Realistic application still achieves a sufficient efficiency gain.



Flag icons from flaticon.com

SuperCDMS Collaboration

134 · 77 · 5 · 4 · 4 · 2 · 1

@SuperCDMS

https://supercdms.slac.stanford.edu