



Variance reduction for neutral particles transport within Geant4

DE LA RECHERCHE À L'INDUSTRIE

Eric Dumonteil, Loic Thulliez

Institut de Recherche sur les Lois Fondamentales de l'Univers, CEA, Université Paris-Saclay

Commissariat à l'énergie atomique et aux énergies alternatives - www.cea.fr



Collaborative work!



Loic Thulliez

Staff physicist @ CEA (IRFU/DPhN)

Expertise in:

- Nuclear physics (experimental + numerical)
- Nuclear data
- Compact accelerator neutron sources
- Fission
- Search of sterile neutrino

Eric Dumonteil

Staff physicist @ CEA (IRFU/DPhN)

Expertise in:

- Statistical mechanics & Branching processes
- Neutron transport & reactor physics
- Monte Carlo for neutron transport
- Variance reduction & shielding
- Population control & criticality

- AMS principle
 - ✓ Basic principle
 - ✓ Adaptation for particle transport
- AMS within Geant4
- Validation using TRIPOLI4

- AMS principle
 - ✓ Basic principle
 - ✓ Adaptation for particle transport
- AMS within Geant4
- Validation using TRIPOLI4

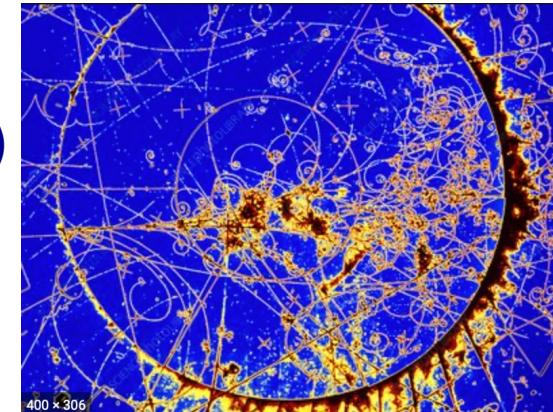
- **Originates from applied mathematics applied to molecular dynamic**
 - (Cerou et al, 2007)
 - (Cerou et al, 2011)
 - (Aristoff et al, 2015)
- **Adaptation to particle transport**
 - CEMRACS@CIRM 2013 (Lelievre & Dumonteil)
 - (Louvin, Dumonteil, Lelievre, Rousset 2017)
 - (Louvin, Mancusi & Dumonteil 2019)
- **Objective of the different developments presented is to fit in the framework of AMS for discrete Markov chains detailed in (Brehier et al, 2016)**

□ Term "tracks" originates from first bubble chambers

□ Parameter space (**position, direction, energy, time, particle type**)

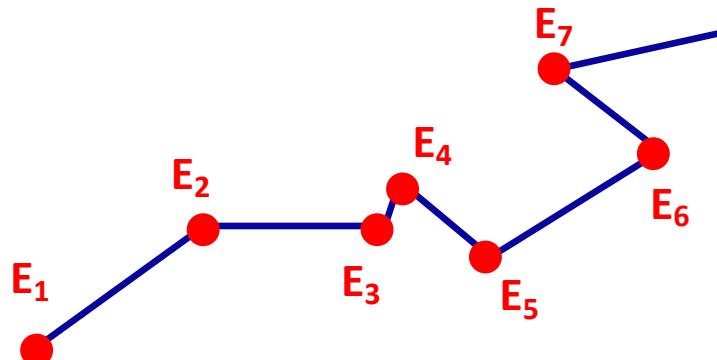
□ Non-charged particle tracks are

- Markovian everywhere in (**position, direction, energy**)
- Markovian at collision points $i \{(\text{position, energy})\}_i$
- Markovian at collision points $i \{(\text{position, direction, energy})\}_i$
- ...

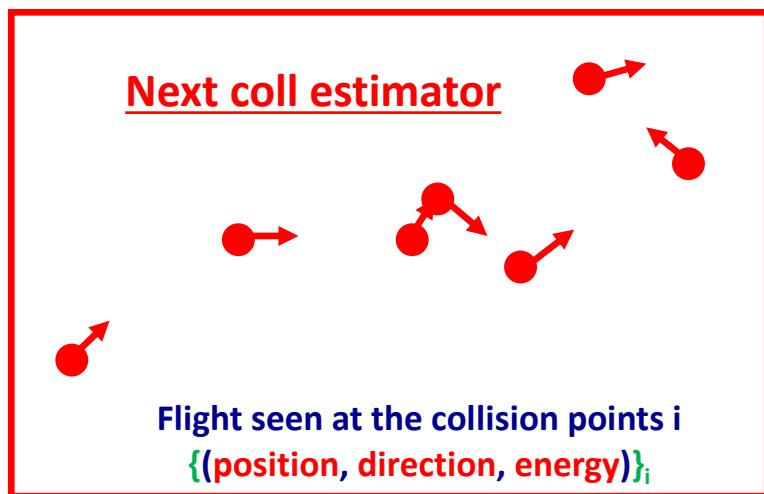


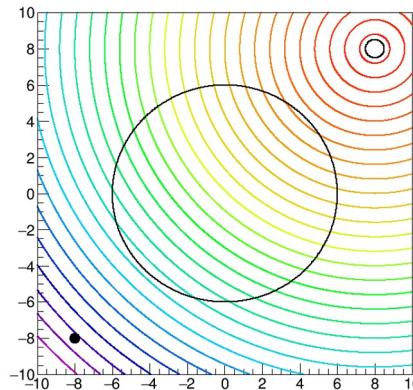
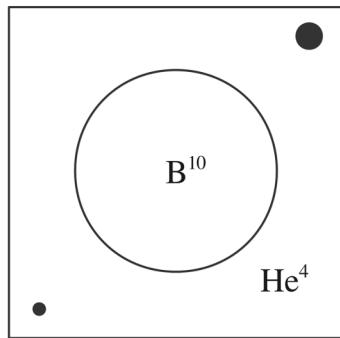
Tracks left by a neutrino interacting with a neutron

□ AMS for discrete Markov chains is the most suitable AMS « flavor »



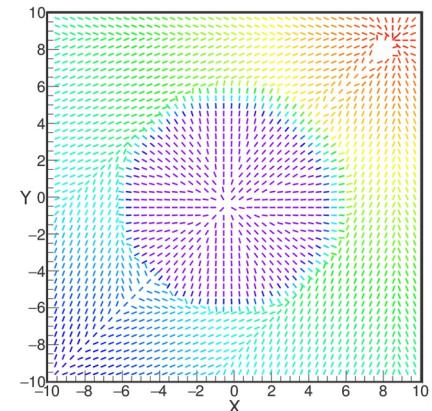
Typical exponential flight undergone by a neutron with anisotropic collision kernel $C(E)$



Example of cost functions:

Simple only-spatial
 $\xi = 1/(1+d)$

Material+energy taken
into account in ξ

Algorithm with 3 parameters

n (# of particles)
 k (# of particles duplicated/iteration)
 ξ (cost function)

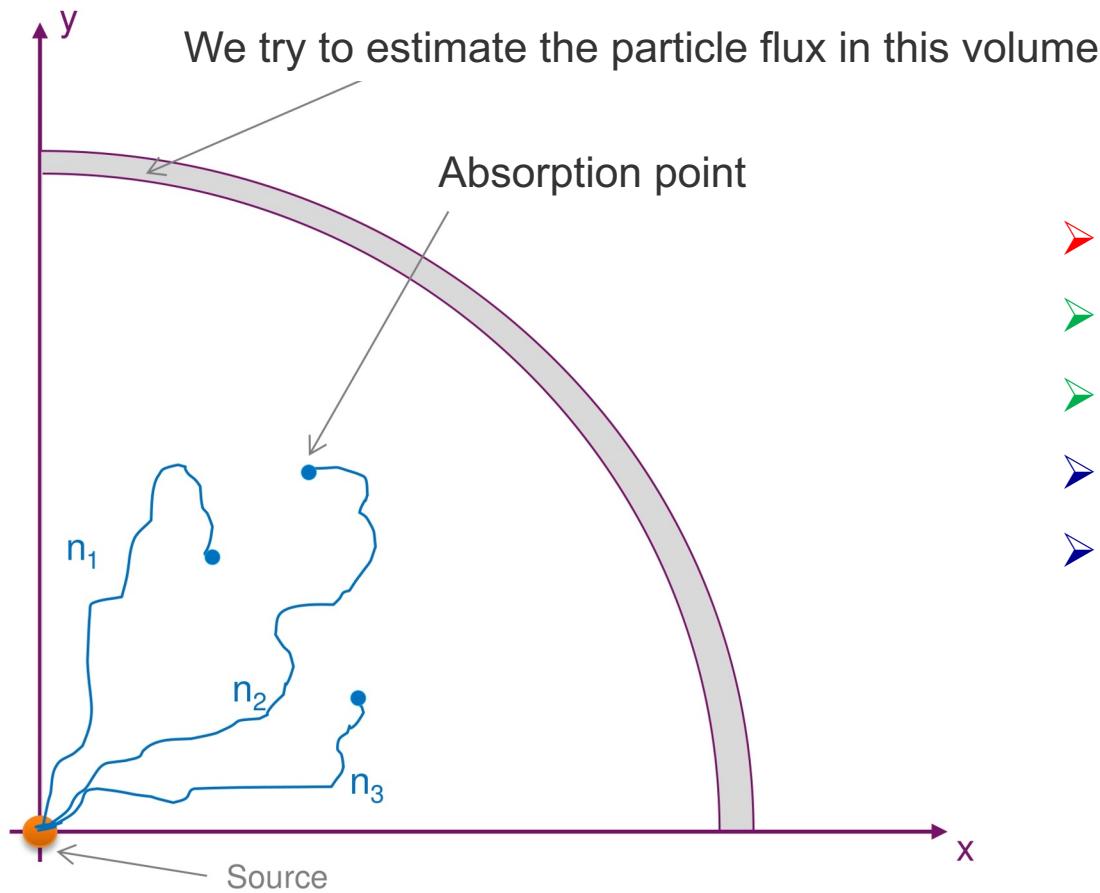
- n** particles simulated naturally => tracks are stored
- score is assigned to each neutron track (= Max of ξ over whole trajectory)
- tracks are ranked according to their score
- the k -th “worst” track defines the new splitting level
- the k tracks having scores lesser than this level are deleted
- k tracks are randomly selected and duplicated at the splitting level
- a new set of n particles is obtained, and we start the whole process again

□ Stopping criterium:

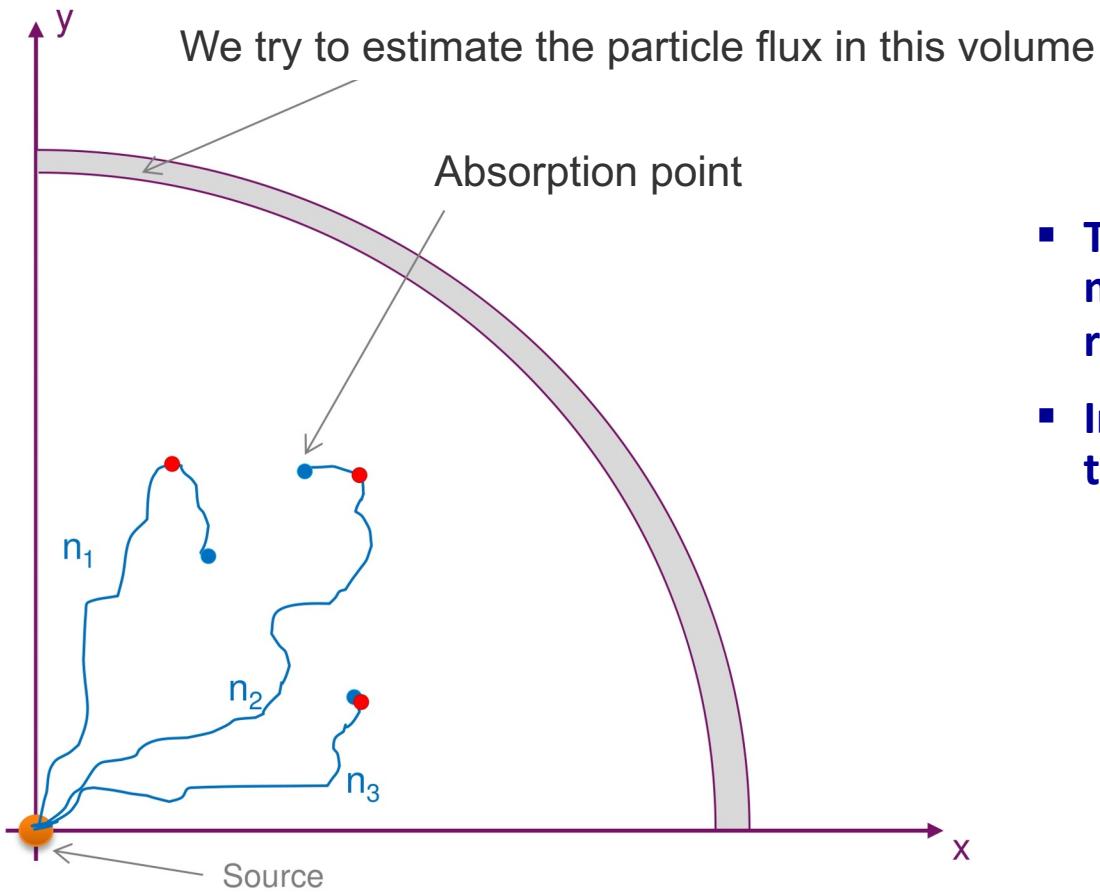
- When $n-k$ tracks have reached the “detector”, the algorithm stops
- The number of iteration corresponding to reach that criterium is N
- Each neutron is assigned a statistical weight α being:

$$\alpha = \left(1 - \frac{k}{n}\right)^N$$

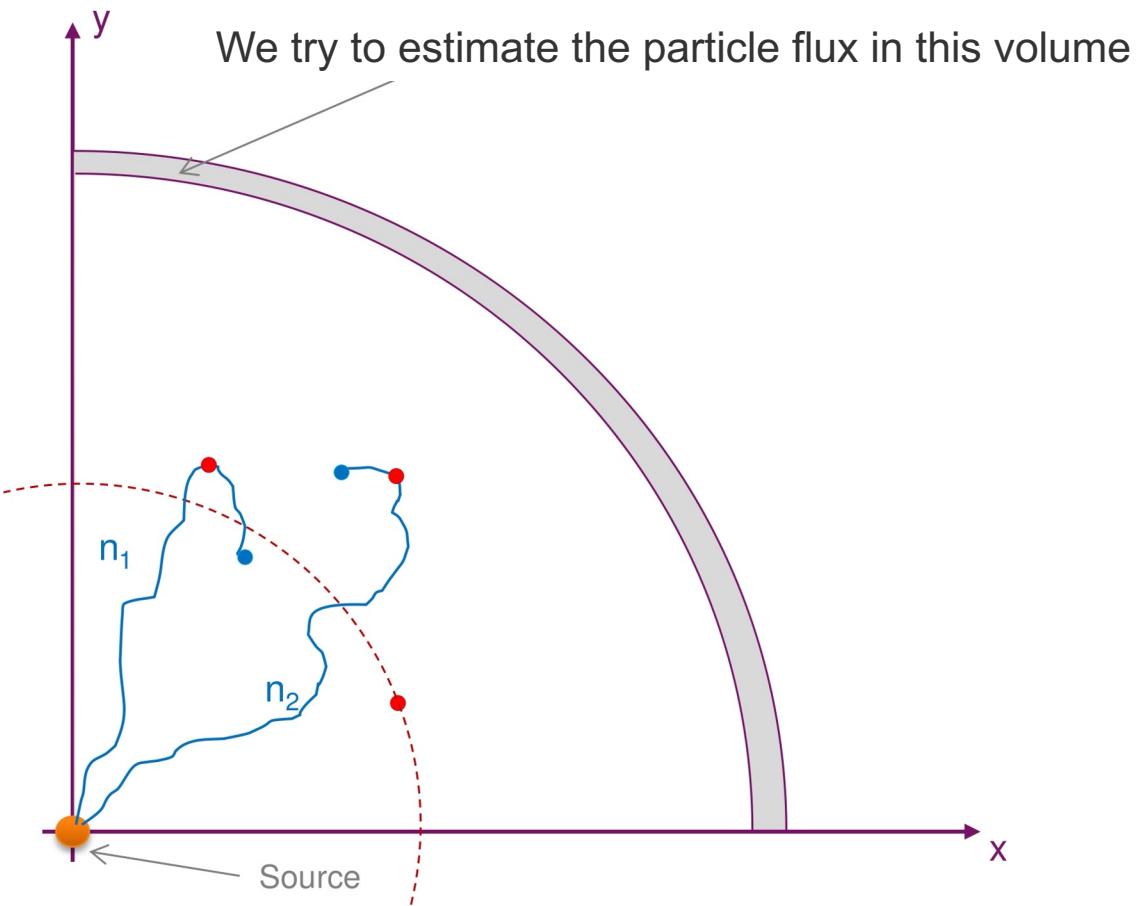
□ An unbiased estimator of the flux is calculated “as usual” using the tracks of the last iteration



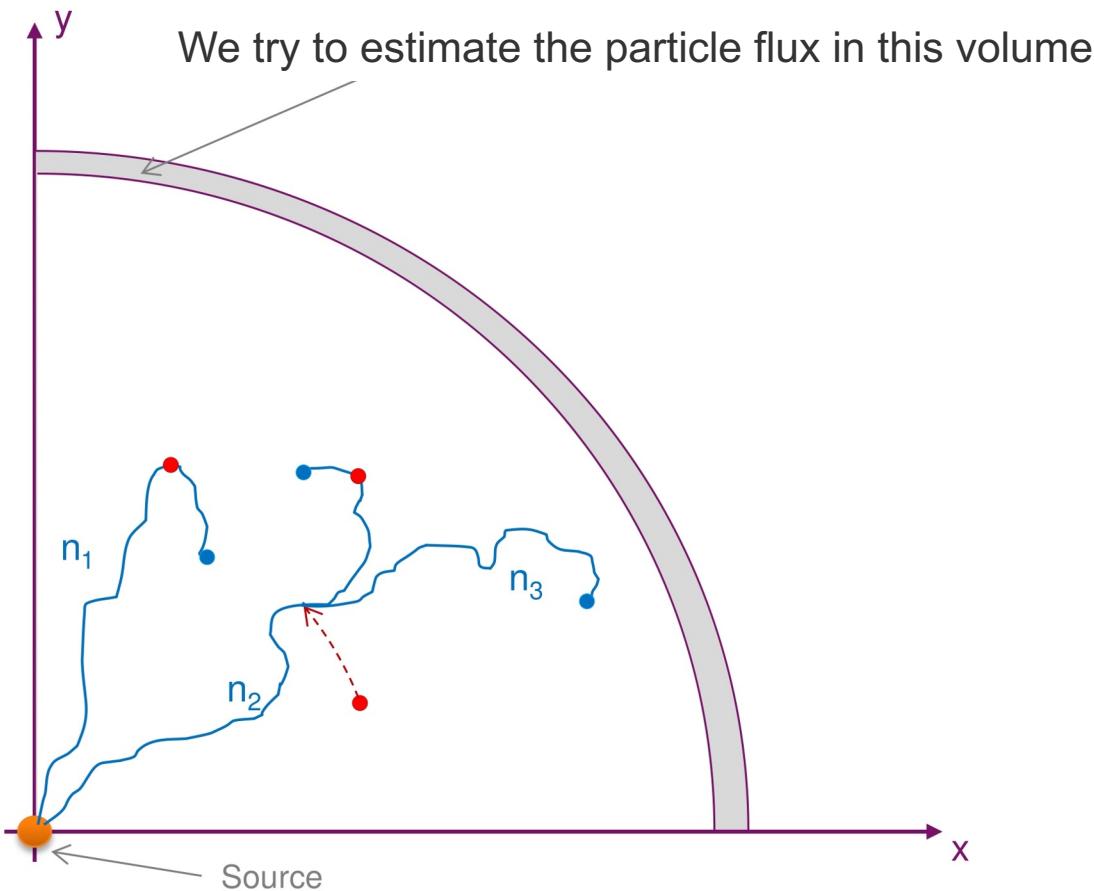
- $n=3$
- $k=1$
- ξ : distance to the source
- Target: spherical shell (purple)
- 3 particles simulated from the source to their absorption (blue points)



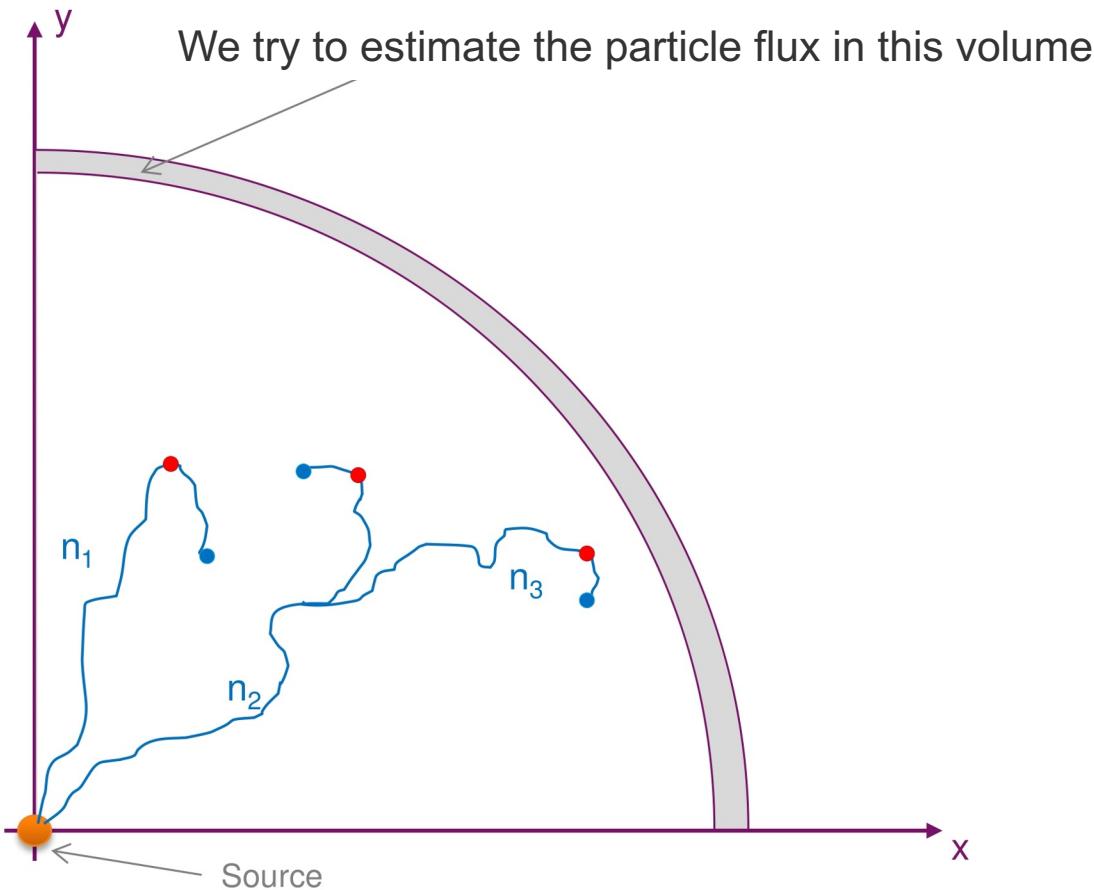
- The importance function is the maximal distance to the source reached by the particle (red points)
- In this case the neutron tracks with the lowest score is n_3



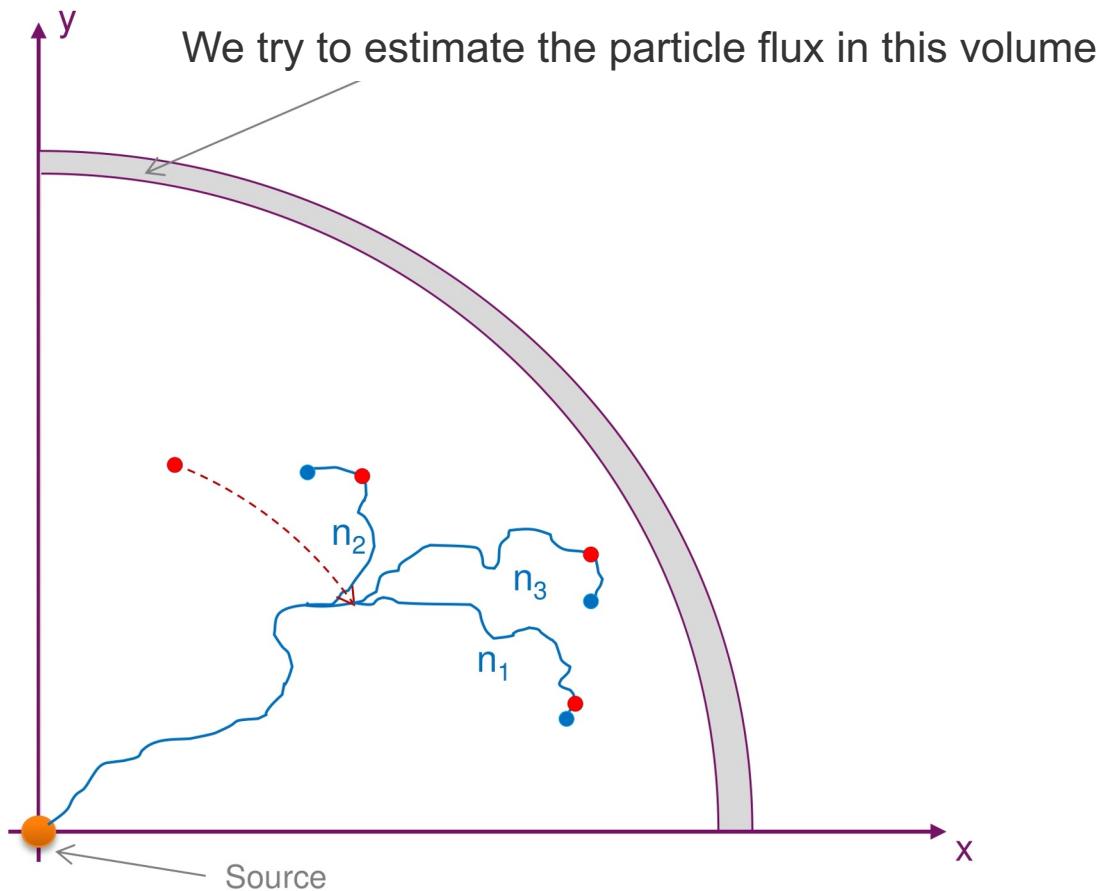
- The tracks associated to particle 3 is deleted
- The maximum score of this track is stored and defines the first splitting level



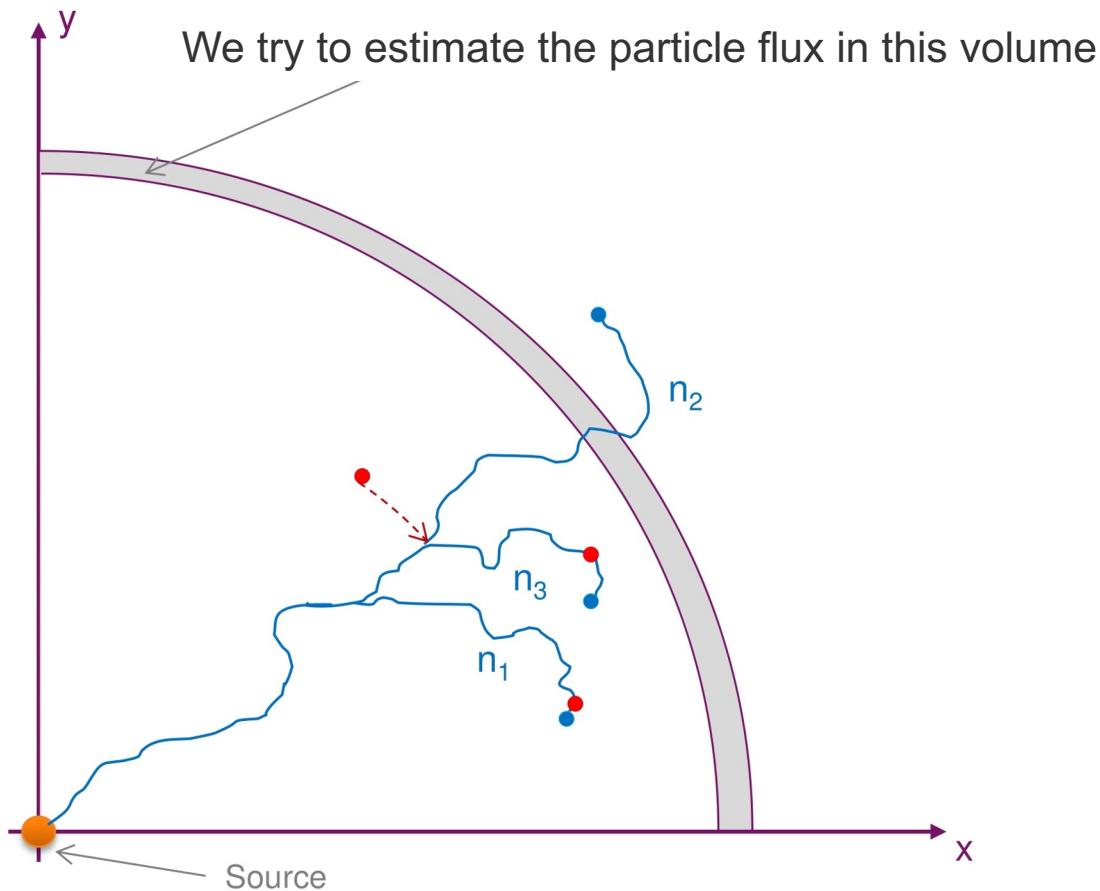
- **Track number 2 is randomly sampled for the splitting**
- **A new particle is simulated from this splitting point until its absorption**



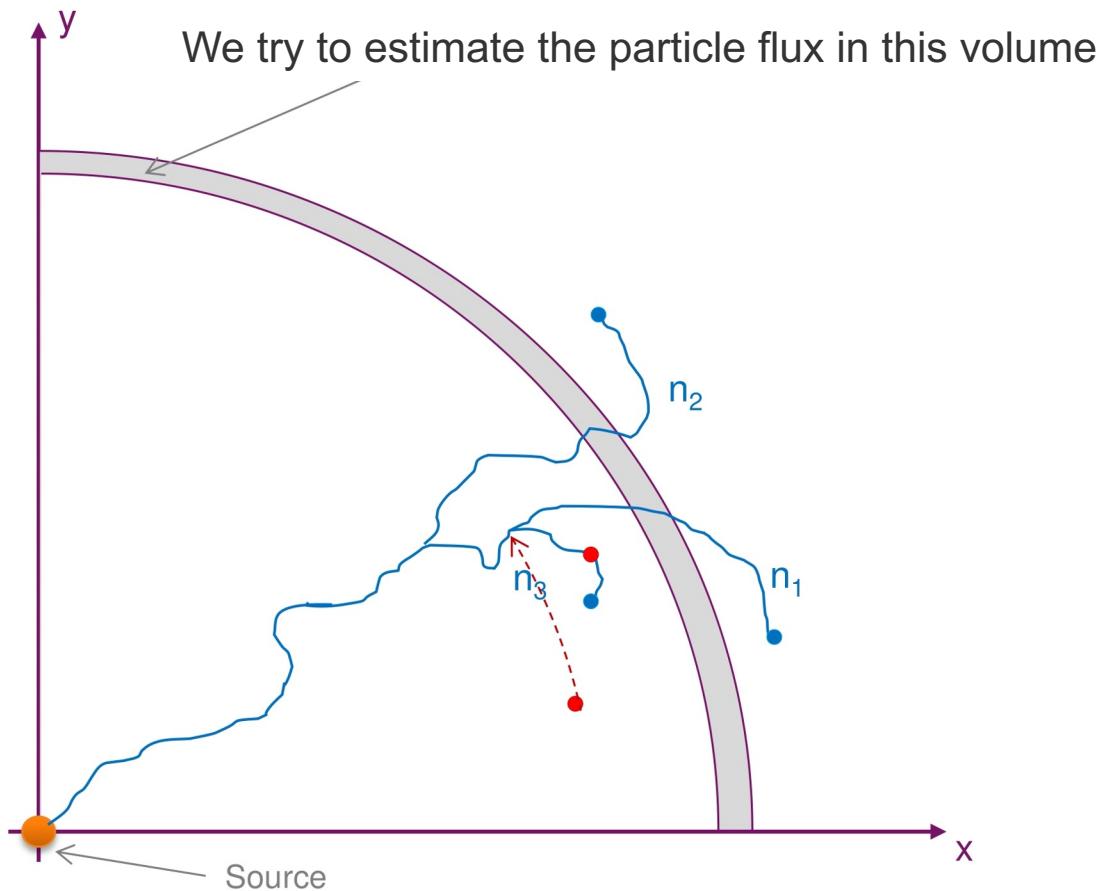
- The score of this new tracks n_3 is calculated
- The first iteration is over
- The stopping criterium is not meet: the iteration process goes on



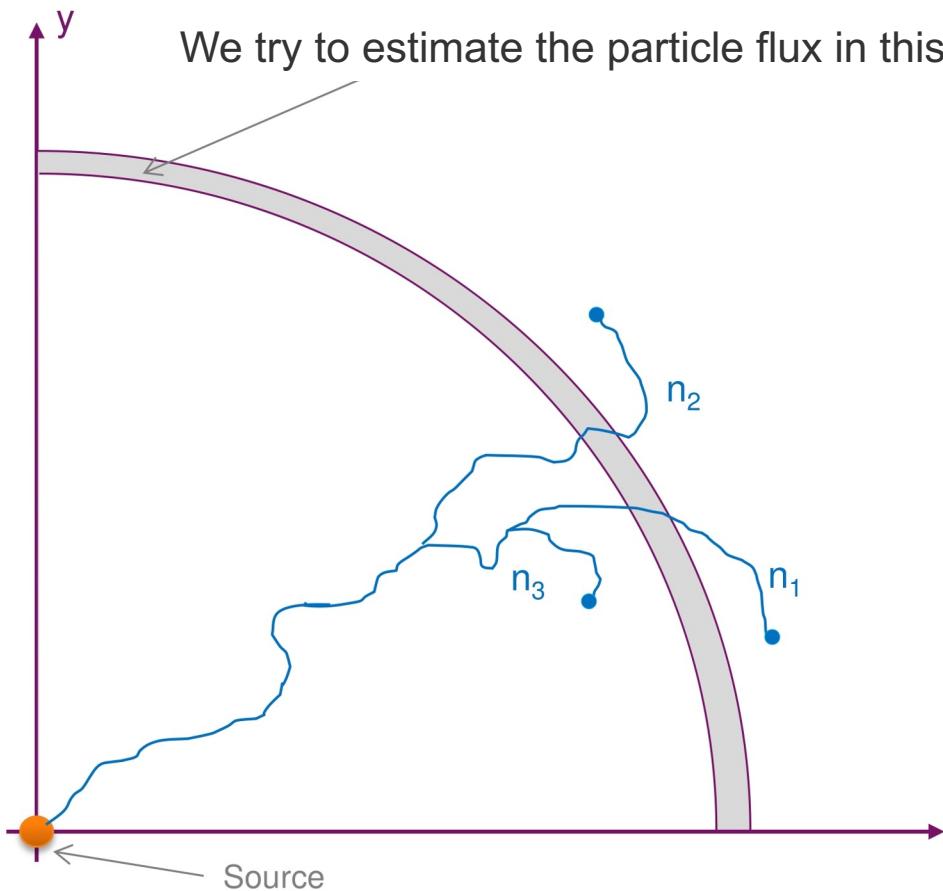
- **Iteration 2**



- **Iteration 3**

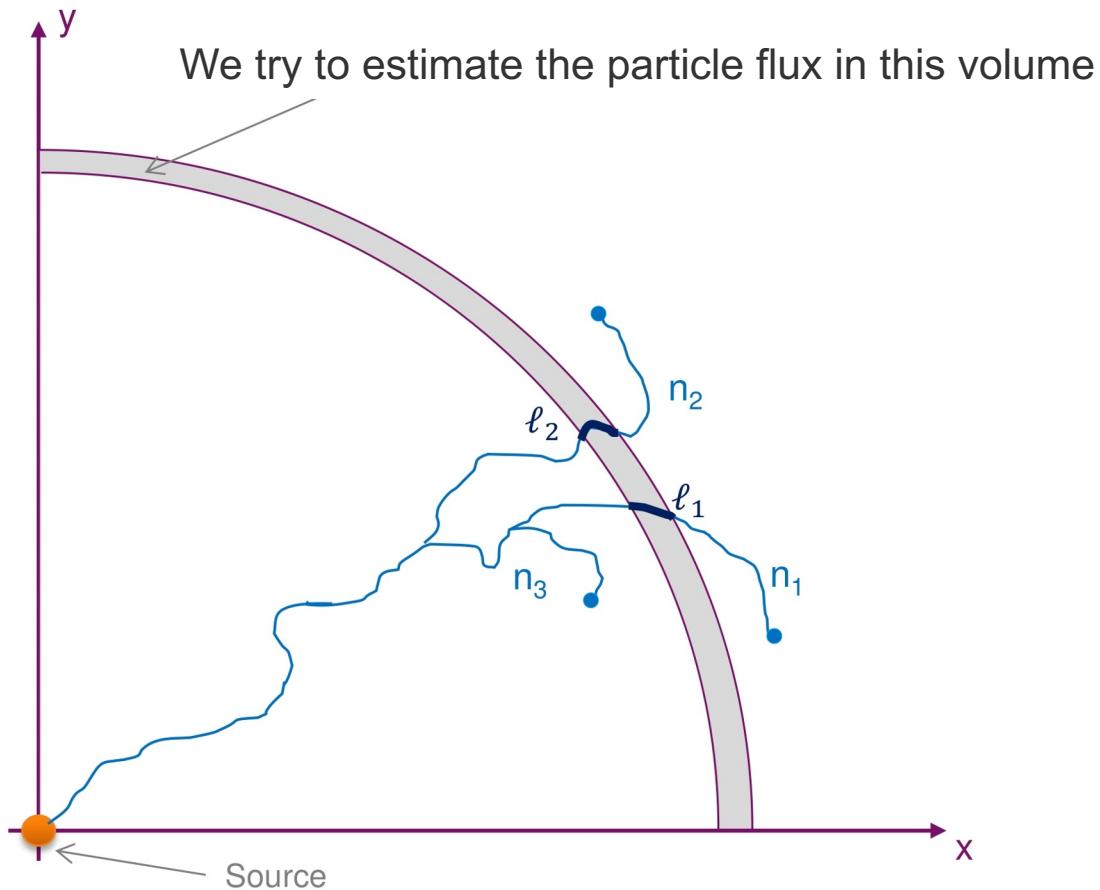


- **Iteration 4**



- **Iteration 4**
- **$n-k$ particles have reached the target, the algorithm stops**
- **The statistical weight of the particles is :**

$$\alpha = \left(1 - \frac{1}{3}\right)^4$$



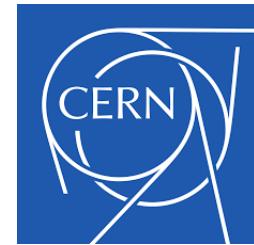
- The flux is calculated according to standard MC flux estimators. For example the travelled length in the spherical shell can be used to tally the flux:

$$\varphi = \frac{1}{3} \alpha(l_1 + l_2)$$



TRIPOLI 4®

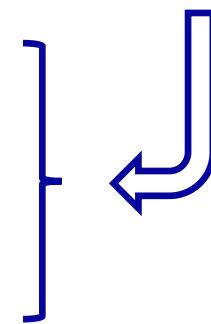
- TRIPOLI-4 @ CEA
- Distributed by OECD/NEA
- Neutron, gamma, e+, e-
- E < 20 MeV
- Evaluated cross-sections
- (Brun et al, 2015)
- www.cea.fr/energies/tripoli-4
- Nuclear industry



- Geant4 @ CERN
- Open-source
- All particles
- All energies
- Both evaluated cross-sections & models
- (Agostinelli et al, 2003)
- geant4.web.cern.ch
- Fundamental / medical / spatial

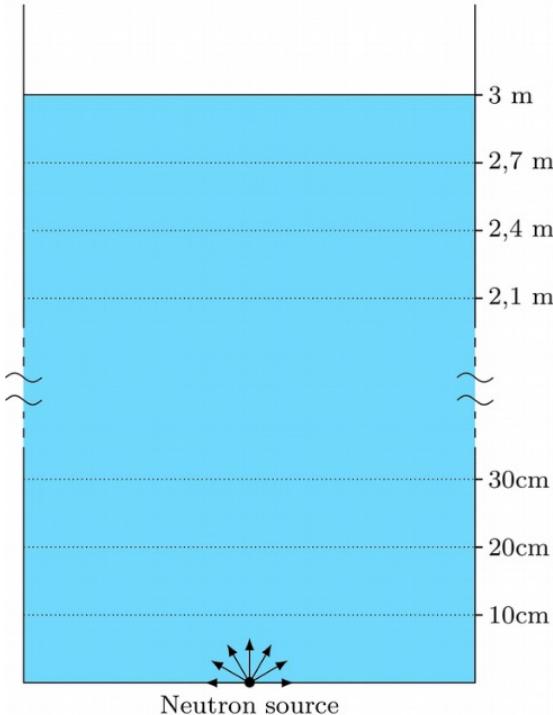
AMS has been implemented in a light (2 classes!) C++ framework
Should be released and distributed open-source in 2023
Small ‘user guide’ to link it to other transport codes

Provided with basic cost functions (distance to spatial detector, radial, ...)
Verified using analytical benchmarks



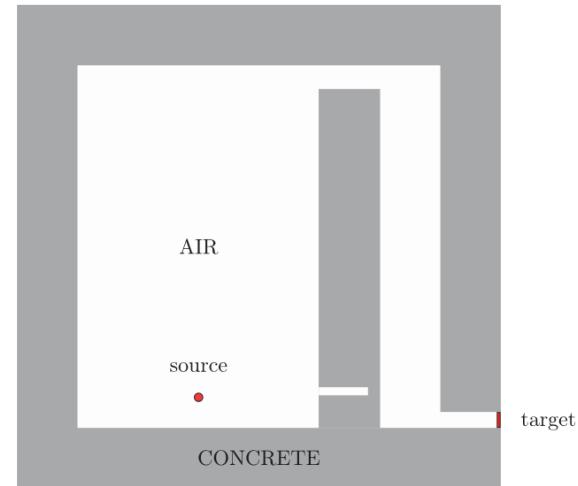
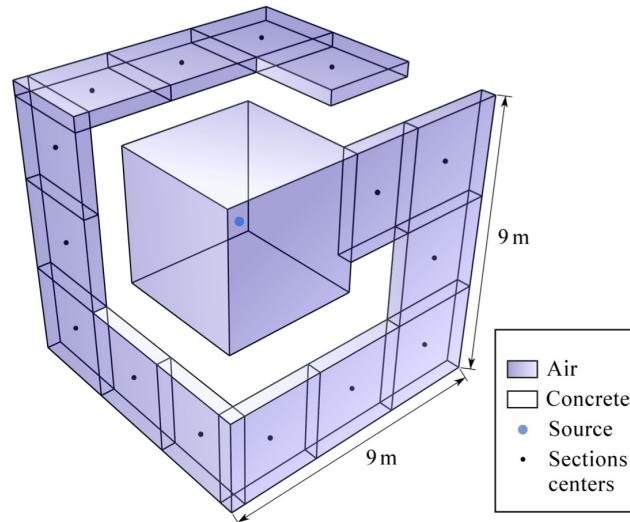
- AMS principle
 - ✓ Basic principle
 - ✓ Adaptation for particle transport
- AMS within Geant4
- Validation using TRIPOLI4

- Many radiation protection problems require to score quantities "everywhere"



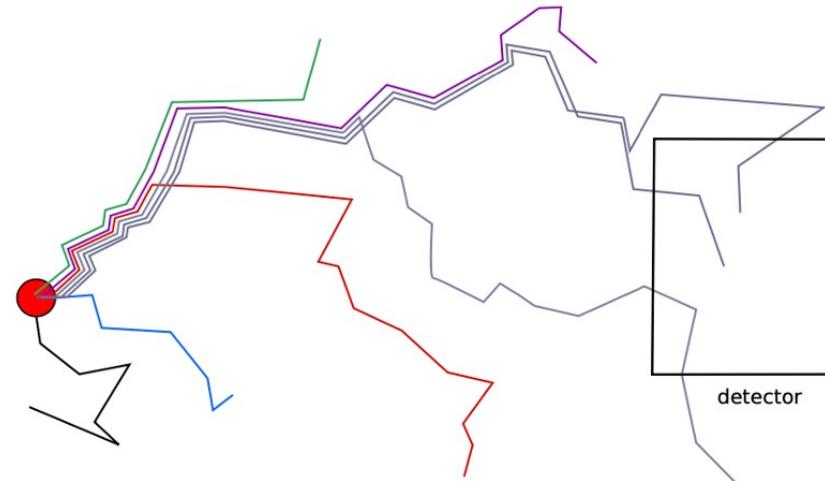
Flux calculation in the
cooling pool of a
reactor

Streaming problems as
can be met in nuclear
marine propulsion



Dose calculations
while dimensioning
shielding rooms

- Following (Brehier, 2016), the idea consists in building a particle genealogy
- Old tracks are kept in memory
- New tracks = copy of the track selected for duplication from 1st point up to splitting point



We consider the analog Monte Carlo unbiased estimate of a score ψ , where $(X_i)_{i \in [1, n]}$ is a set of analog tracks

$$\hat{\psi}_{MC} = \frac{1}{n} \sum_{i=1}^n \psi(X_i),$$

For any iteration q of the scoring process we define the following unbiased estimate:

$$\hat{\psi}_q = w_q \hat{\psi}_q^{\text{on}} + \sum_{j=0}^q w_j \hat{\psi}_j^{\text{off}},$$

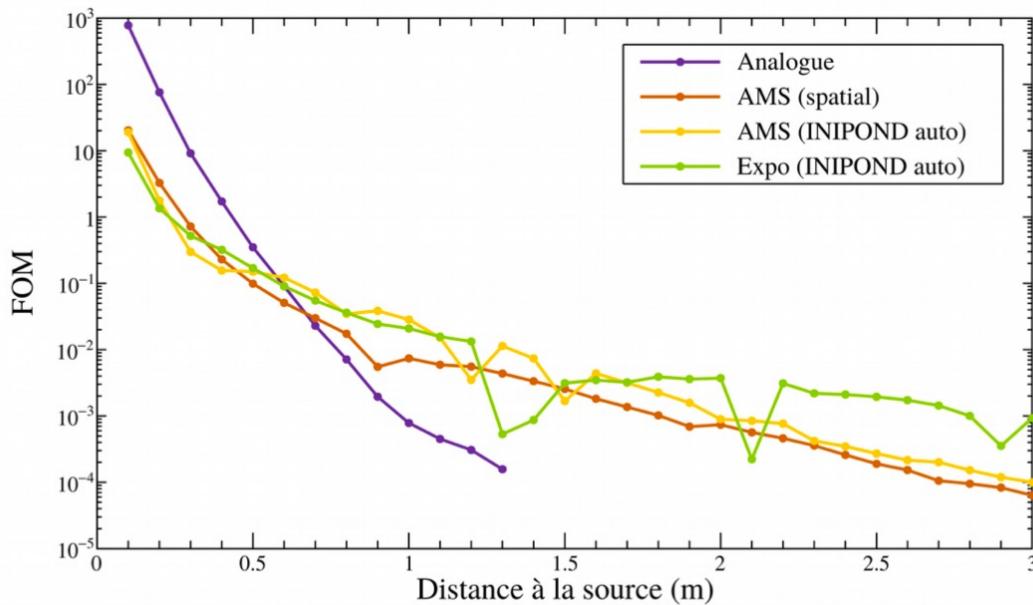
with $w_j = \begin{cases} \frac{1}{n} & \text{if } j = 0 \\ \frac{1}{n} \prod_{i=0}^{j-1} \left(1 - \frac{K_i}{n}\right) & \text{if } j > 0 \end{cases}$

$$\hat{\psi}_q^{\text{on}} = \sum_{X \in T_q^{\text{on}}} \psi(X) \quad \text{and} \quad \hat{\psi}_q^{\text{off}} = \sum_{X \in T_q^{\text{off}}} \psi(X).$$

$\rightarrow T / \xi > z_q$

$K_q = \# \text{ tracks} / \xi < z_q$
 $z_q = \text{splitting level}$

Cooling water pool

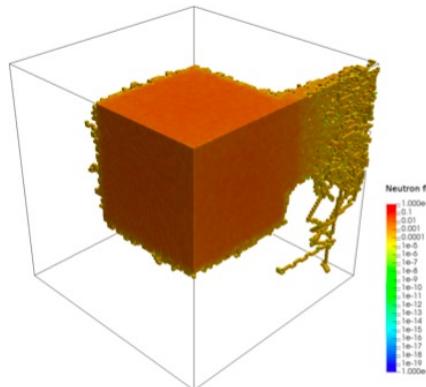


TRIPOLI-4 embedded
deterministic solver for
~adjoint flux calculation

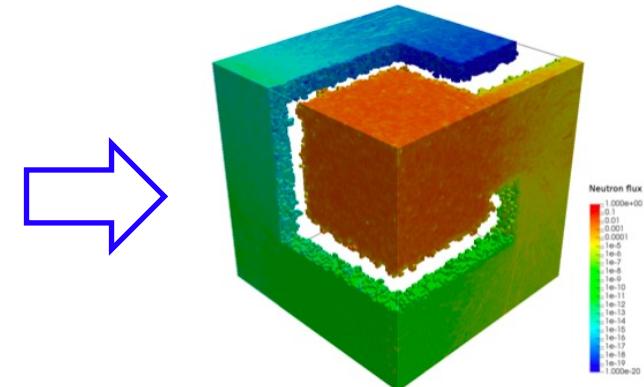
- ξ either spatial or INIPOND
- AMS FOM close to ET ...
- ... even with naïve ξ !

Labyrinth

- 2 MeV isotropic neutron source
- Air surrounded by concrete
- ET fails due to air concentration
- AMS $\xi = 0$ in concrete, growing following path to exit



(a) Analog neutron flux



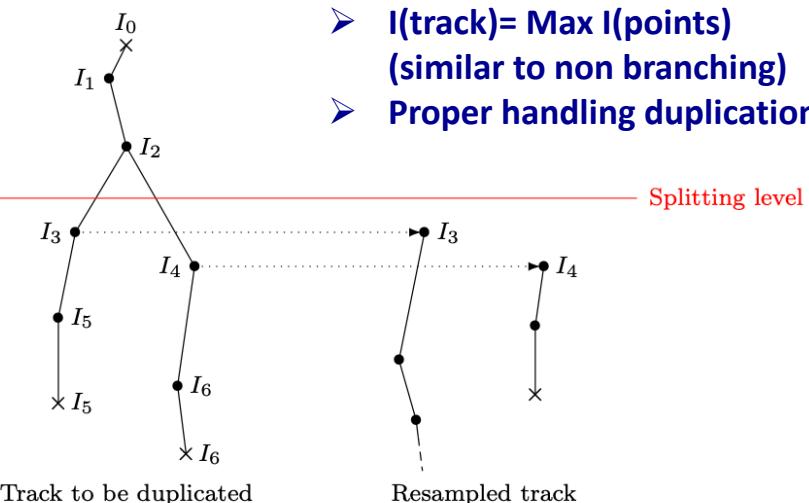
(b) AMS neutron flux

- Many examples of branching structures in particle transport:

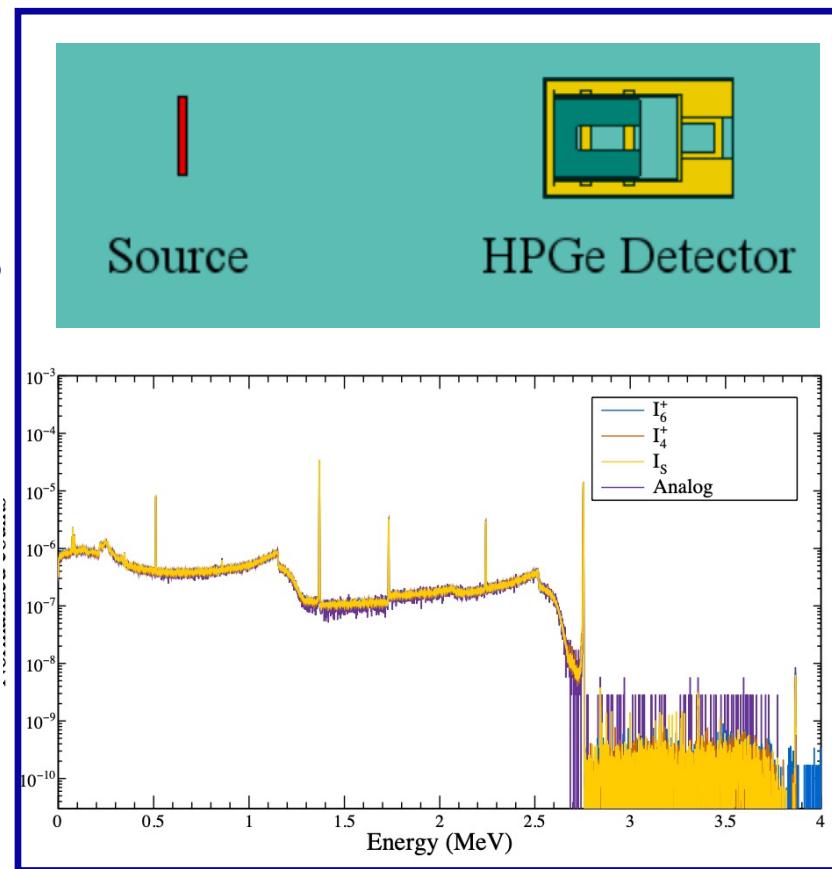
- (n,2n) ➤ electromagnetic cascade
- fissions ➤ Intranuclear cascade

- Weights/particles could be introduced but correlations would be lost

- Handled by an appropriate algorithm within AMS so as to preserve correlations



HPGe detector: g, e⁺, e⁻



➤ FOM_{AMS}/FOM_{analog} ~ 10²

- AMS principle
 - ✓ Basic principle
 - ✓ Adaptation for particle transport
- AMS within Geant4
- Validation using TRIPOLI4

C++ Geant4 interface Adaptation to CANS & Nucleus

NEEDS TO BE DISCUSSED...

... WITH WHOEVER MIGHT GIVE US ADVICES
ON THE BEST WAY TO IMPLEMENT IT !

Step 0: Copy and paste:

- > AMS.cc / ImportanceFunctions.cc in ./Sources/
- > AMS.hh / ImportanceFunctions.hh in ./Headers/

Step 1: In main.cc:

- > Add the headers:

```
#include "AMS.hh"
#include "ImportanceFunctions.hh"
AMS_Batch* amsBatch=NULL; ///AMS_Batch definition
ImportanceFunction* importanceFunction=NULL; ///Importance function for AMS
```

- > Add the add_to_main.cc

Step 2: In SteppingAction.cc

- > Add the private member in SteppingAction.hh: bool isAMS_Flag_On_

- > In SteppingAction.cc add the headers:

```
#include "ImportanceFunctions.hh"
#include "AMS.hh"
```

- > In SteppingAction::SteppingAction {

```
    isAMS_Flag_On_ = 0;
    if(/something/) isAMS_Flag_On_ = 1;
```

```
}
```

- > In NUSteppingAction::UserSteppingAction(const G4Step* aStep){

```
    ///Add the block defined in ./AMS_package/add_to_steppingAction.cc
```

```
}
```

Step 3: In PrimaryAction.cc

-> Add the method:

```
//-----  
The header: #include "AMS.hh"
```

Add in the member:

```
bool isAMS_Flag_On_;
```

Add in the constructor:

```
isAMS_Flag_On_ = 0;  
if(something) isAMS_Flag_On_ = inputs->getBool("Geant4/Geometry/AMS");
```

```
void PrimaryGeneratorAction::GeneratePrimaries_AMS(G4Event* anEvent){  
    //Manage by AMS  
    AMS_Point* amsPoint = amsBatch->GetPointToRestart();  
    int pdgEncoding = amsBatch->GetCurrentTrack()->GetCurrentBranch()->GetPDGEncoding();  
    particle_ = G4ParticleTable::GetParticleTable()->FindParticle( pdgEncoding );  
    particleGun_->SetParticleDefinition(particle_);  
  
    double* position = amsPoint->GetPosition();  
    double* direction = amsPoint->GetDirection();  
    particleGun_->SetParticlePosition(G4ThreeVector(position[0], position[1], position[2])); ///mm by default  
    particleGun_->SetParticleMomentumDirection(G4ThreeVector(direction[0], direction[1], direction[2]));  
    particleGun_->SetParticleEnergy(amspoint->GetEnergy()); ///MeV by default  
    particleGun_->SetParticleTime(amspoint->GetTime()); ///ns by default  
    particleGun_->GeneratePrimaryVertex(anEvent);  
    G4PrimaryVertex* primaryVertex = anEvent->GetPrimaryVertex();  
    primaryVertex->SetWeight(amspoint->GetWeight());  
}
```

Step 4:

In the class RunAction.cc:

- > Add the method: TFile*& getROOTFile(){return rootFileOut_;}
The ROOT file is passed to the AMS::Scoring method

Step 5:

In the class DetectorConstruction.cc:

- > Add the method:

```
vector<string> getAllPhysicalVolumeNames() {
    vector<string> names;
    for(auto it = physicalVolume_.begin(); it != physicalVolume_.end(); ++it) names.push_back(it->first);
    return names;
}
```

Step 6:

Please implement the method AMS_Batch::SaveResults() in AMS.cc

You need to customize it according to your needs, i.e. the output that you want

- AMS principle
 - ✓ Basic principle
 - ✓ Adaptation for particle transport
- AMS within Geant4
- Validation using TRIPOLI4

- Unidirectional 1D transport with capture in an infinite homogeneous medium
- Total cross section $\Sigma = 1/cm$, Scattering probability p , Capture probability $1-p$
- Score: flux (mean chord length) in spherical concentric shells

$$p(x, 1) = \Sigma_t e^{-\Sigma_t x}$$

Probability to scatter in x

$$\mathcal{L}_{p(x,1)}(-t) = \frac{\Sigma_t}{\Sigma_t - t}$$

Moment generating function

$$\mathcal{L}_{p(x,n)}(-t) = \left(\frac{\Sigma_t}{\Sigma_t - t} \right)^n$$

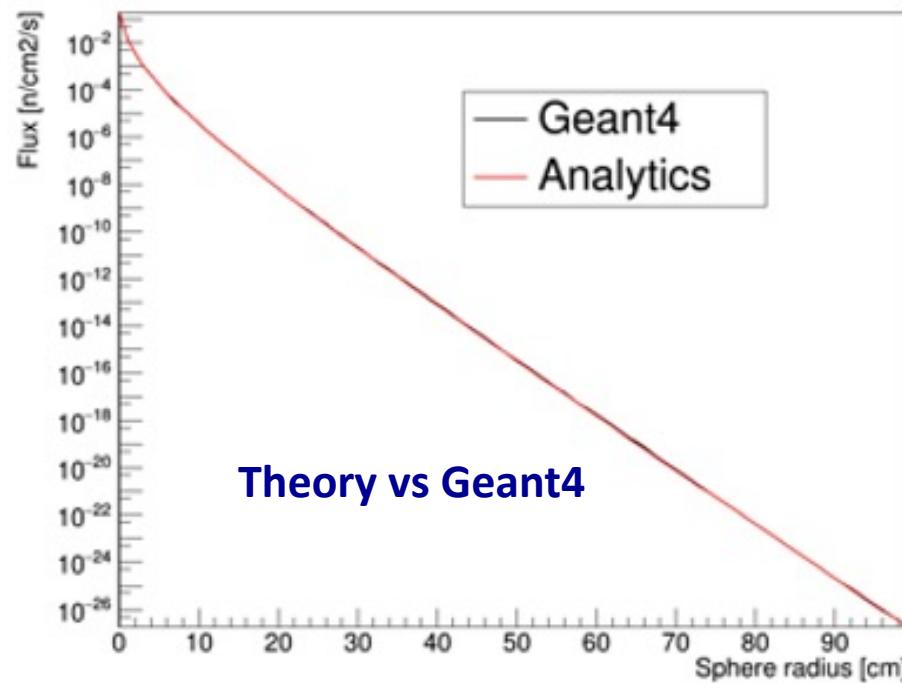
n-th convolution

Probability that the n-th collision reaches x :

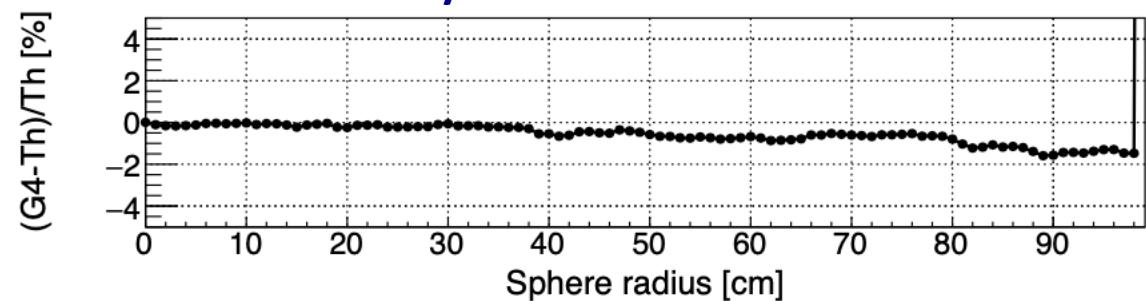
$$p(x, n) = \mathcal{L}_{p(x,n)}^{-1}(-t) = \frac{x^{n-1} e^{-\Sigma_t x} (p\Sigma_t)^n}{p\Gamma(n)}$$

Collision density in x :

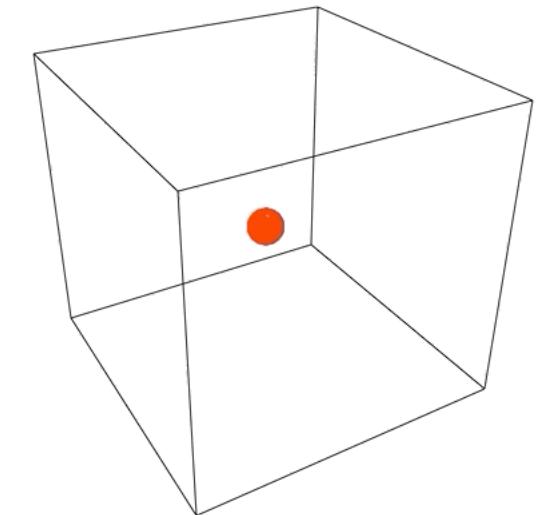
$$\psi(x) = \sum_{n=1}^{\infty} \frac{x^{n-1} e^{-\Sigma_t x} (p\Sigma_t)^n}{p\Gamma(n)} = \Sigma_t e^{-(p-1)\Sigma_t x}$$



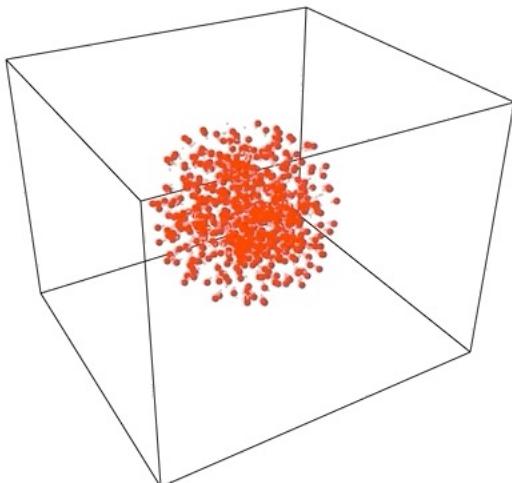
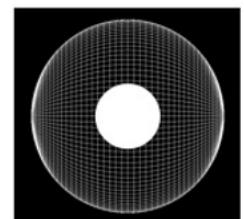
**Relative differences
between
Theory and Geant4**



$$\phi(x) = \frac{3}{4\pi(r_2^3 - r_1^3)} \int_{r_1}^{r_2} \psi(x) = \frac{3}{4\pi(r_2^3 - r_1^3)(p-1)} (e^{-(p-1)\Sigma_t r_2} - e^{-(p-1)\Sigma_t r_1})$$

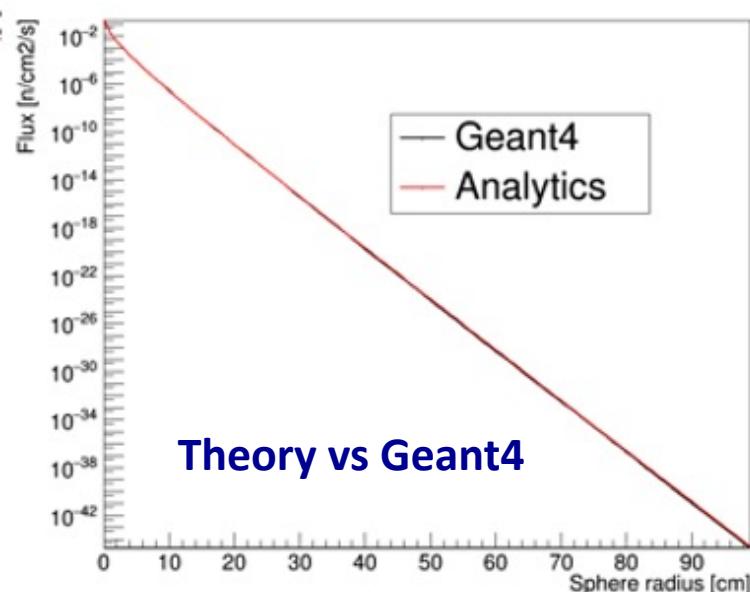


- ❑ 3D isotropic exponential flights & diffusion in an infinite homogeneous medium
- ❑ Total cross section $\Sigma = 1/cm$, Scattering probability p , Capture probability $1-p$
- ❑ Score: flux (mean chord length) in spherical concentric shells
- ❑ Once again, using random walk theory to calculate the flux leads to:

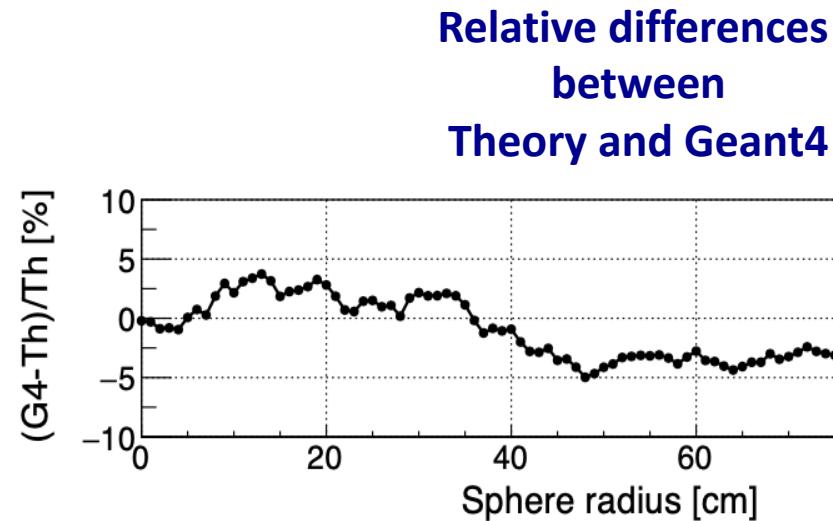


$$\phi(r) = \int_0^{\infty} r^{\frac{3}{2}} k^{-\frac{1}{2}} J_{\frac{1}{2}} \psi(k) dk \longrightarrow \text{Scalar flux}$$

$$\phi = \frac{3}{4\pi(r_2^3 - r_1^3)} \int_{r_1}^{r_2} \phi(r) dr \longrightarrow \text{Integrated flux}$$



Theory vs Geant4



Relative differences
between
Theory and Geant4

- Excellent agreement...
- ... up to very large attenuations (10^{-42})
- Extremely large FOM (10^{36}) at 1m

- AMS developments within Geant4 have been verified and validated
 - on the flight scorers
 - branching processes
- Might be useful in particle physics, and more generally in context where one is interested in :
 - Simulation of rare events / processes (multiparticles cascades)
 - Simulation of correlated background noise
 - Simulation of events resulting from the correlation of different particles detected in different detectors
- Particularly efficient when the “importance function” is unknown as it is robust

- F. Cérou and A. Guyader, "Adaptive Multilevel Splitting for Rare Event Analysis", Stochastic Anal.Appl. 25,2, 417 (2007).
 - F. Cérou, et al. "A multiple replica approach to simulate reactive trajectories", Journal of Chemical Physics, 134, 054108, (2011).
 - D. Aristoff, T. Lelièvre, C.G. Mayne and I. Teo. "Adaptive multilevel splitting in molecular dynamics simulations" ESAIM Proc., 48:215-225, (2015).
-
- H. Louvin et al., "Adaptive Multilevel Splitting for MonteCarlo Particle Transport", Eur. Phys. J. Nucl. Sci. Technol. 3,29 (2017).
 - H. Louvin, "Development of an Adaptive VarianceReduction Technique for Monte Carlo Particle Transport", PhD Thesis, Université Paris-Saclay (2017).
 - M. Nowak et al, "Accelerating Monte Carlo Shielding Calculations inTRIPOLI-4 with a Deterministic Adjoint Flux », Nucl. Sci. Eng., 193 (2019).
 - Mancusi et al., « Evaluating importance maps for using deterministic or on-line methods », ANS RPSD Proceedings (2018)
-
- E. Brun et al., "Tripoli-4®, CEA, EDF and AREVAReference Monte Carlo Code", Ann. Nucl. Energy 82,151 (2015).



Questions ?

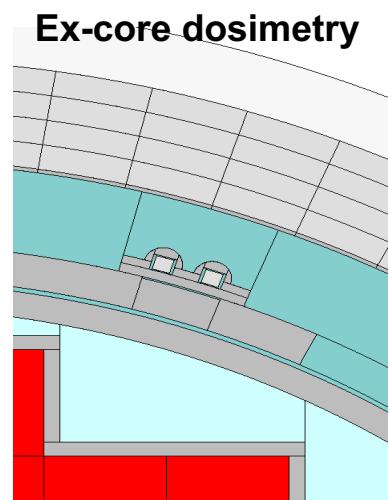


SPARE SLIDES

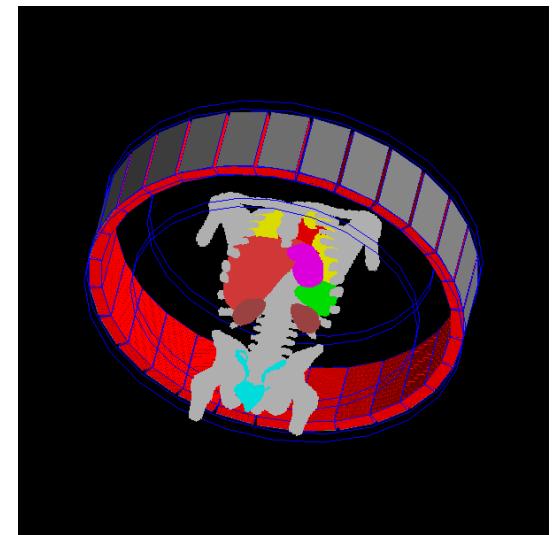
- Static Boltzmann equation in non-reproductive media and fixed sources
- Neutrons, photons between 0 and 20 MeV
- Main challenge: variance reduction w.r.t. x and E parameters



Radioactive waste management

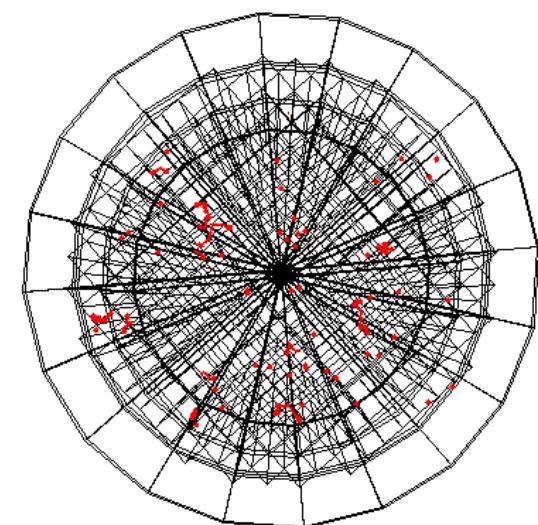
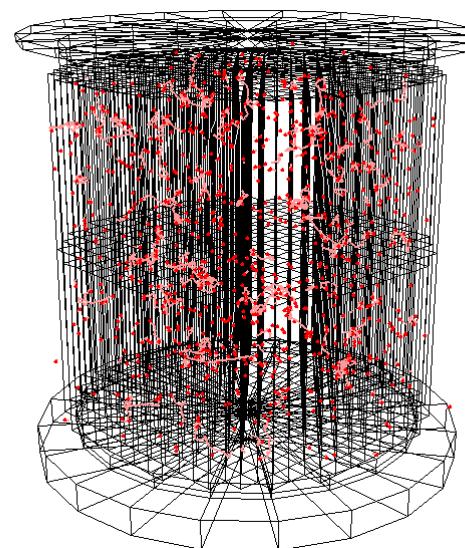
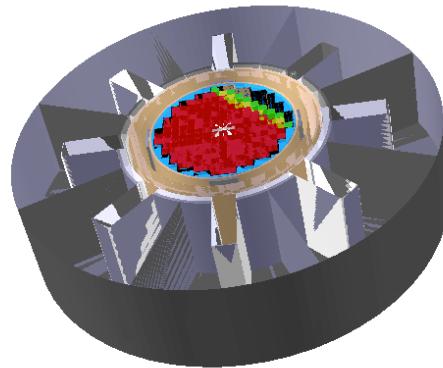


Ex-core dosimetry

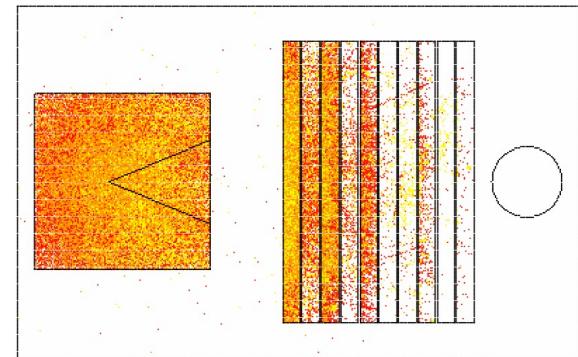


Radiotherapy
(opengatecollaboration.org)

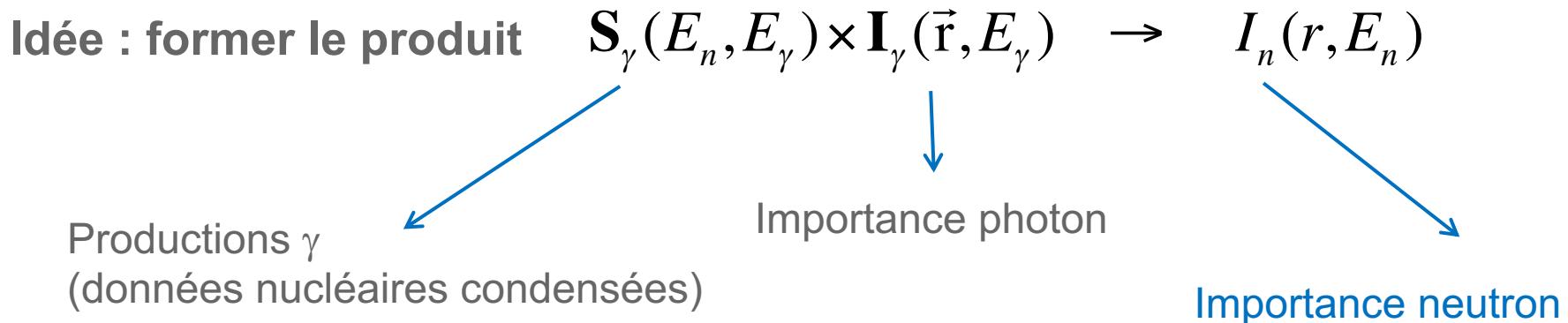
- Static linear Boltzmann equation in fissile media (no fixed sources)
- Neutrons between 0 and 20 MeV
- Main challenge: finding eigenvalues and eigenvectors while tackling correlations



Biaisage en couplé neutron-gamma

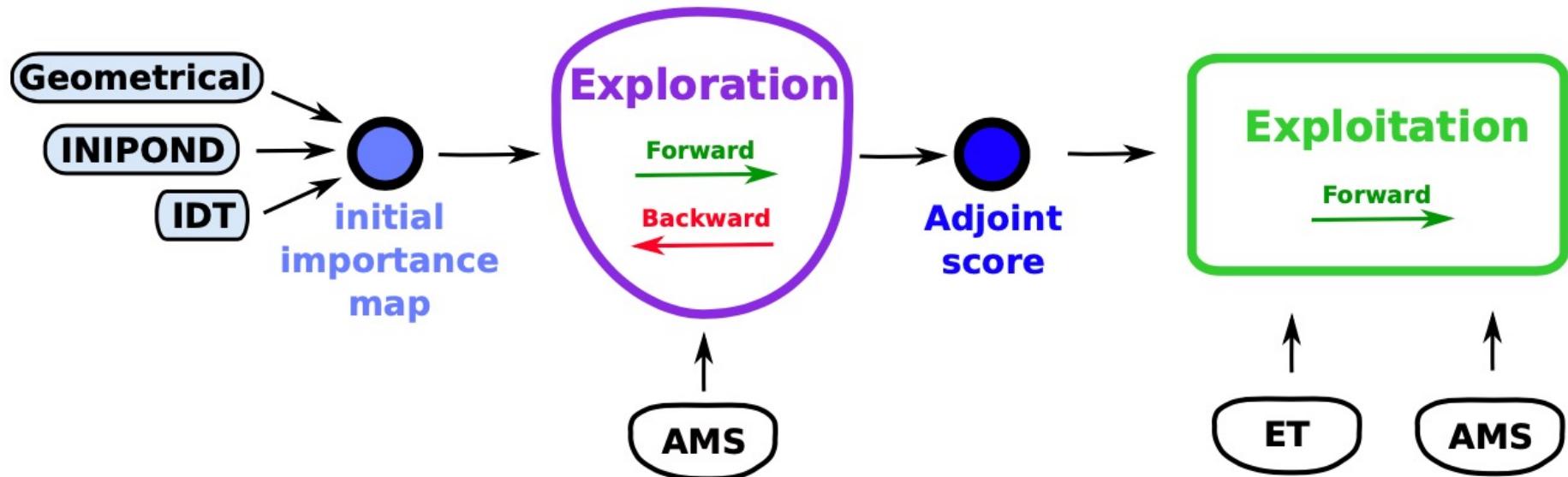


- calculs couplés n, γ (avec scores γ)
- où attirer les **neutrons** pour produire des **photons importants** ?



Calculer les ratios $I_n(r, E_n) / I_n(r, E_n)_{\max}$ → diagnostic
→ β_n neutron, ou utiliser une surface attractrice neutron,
ou favoriser la production γ localement.

- Investigating the use of machine learning to improve the estimation of the adjoint flux
- (Nowak et al, 2018)
- Numerical 2-step scheme using AMS to adaptively improve the cost function / adjoint flux :



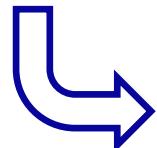
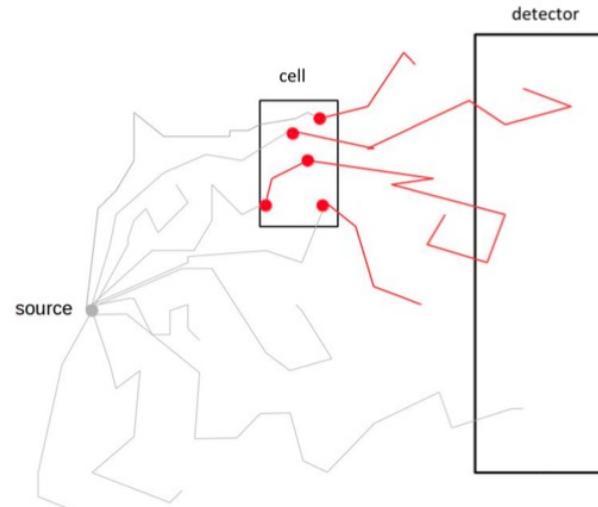
□ Standard MC estimator of the adjoint flux

Expected contribution c of a point
in the phase space

$$x = (\vec{r}, \vec{\Omega}, E)$$

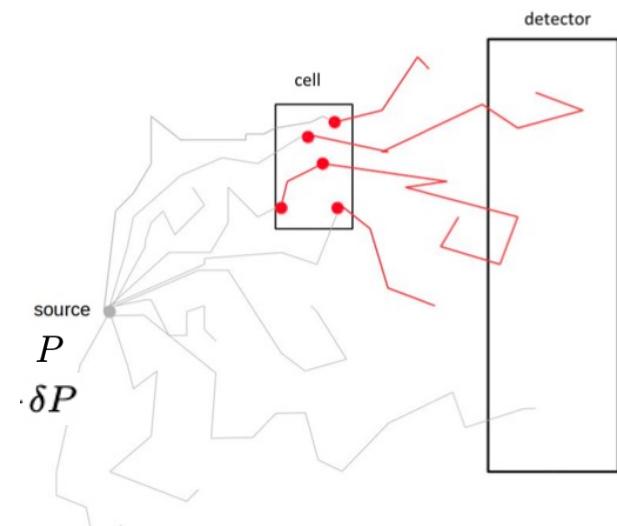
to a response (flux in a detector).

$$\psi^\dagger(P) = \mathbb{E}(c|P)$$



- N histories $\{T_0, \dots, T_N\}$
- N_i points $\{P_0^{(i)}, \dots, P_{N_i}^{(i)}\}$

$$\psi^\dagger(P) = \frac{\sum_{P_i \in \mathcal{T}} c_i \mathbf{1}_{\delta P}(\mathbf{r}_i, E_i, \boldsymbol{\Omega}_i)}{\sum_{P_i \in \mathcal{T}} \mathbf{1}_{\delta P}(\mathbf{r}_i, E_i, \boldsymbol{\Omega}_i)}$$



□ AMS estimator of the adjoint flux => the same with weights given by

$$\psi^\dagger(P) = \frac{\sum_{P_i \in \mathcal{T}} c_i \mathbb{1}_{\delta P}(\mathbf{r}_i, E_i, \Omega_i)}{\sum_{P_i \in \mathcal{T}} \mathbb{1}_{\delta P}(\mathbf{r}_i, E_i, \Omega_i)}$$

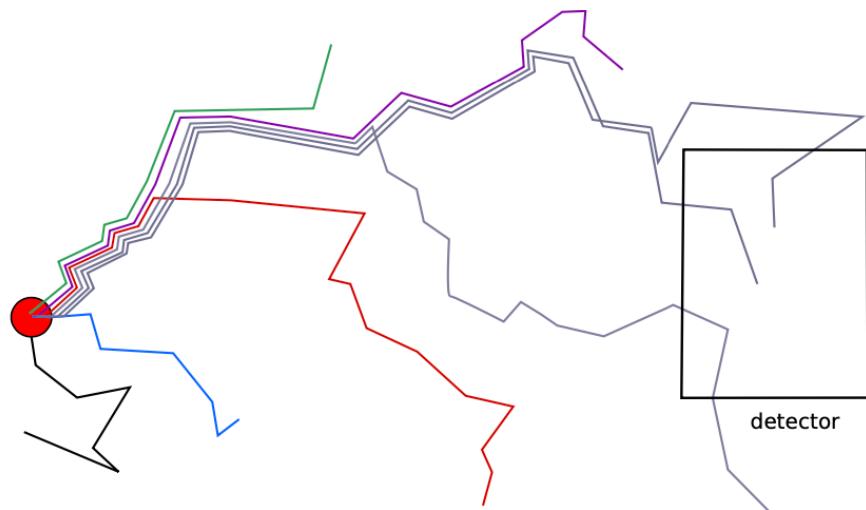
becomes

$$\frac{\sum_{\mathcal{T} \in T^{\text{AMS}}} w_{\text{AMS}}(\mathcal{T}) \sum_{P_i \in \mathcal{T}} c_i \mathbb{1}_{\delta P}(\mathbf{r}_i, E_i, \Omega_i)}{\sum_{\mathcal{T} \in T^{\text{AMS}}} w_{\text{AMS}}(\mathcal{T}) \sum_{P_i \in \mathcal{T}} \mathbb{1}_{\delta P}(\mathbf{r}_i, E_i, \Omega_i)}$$

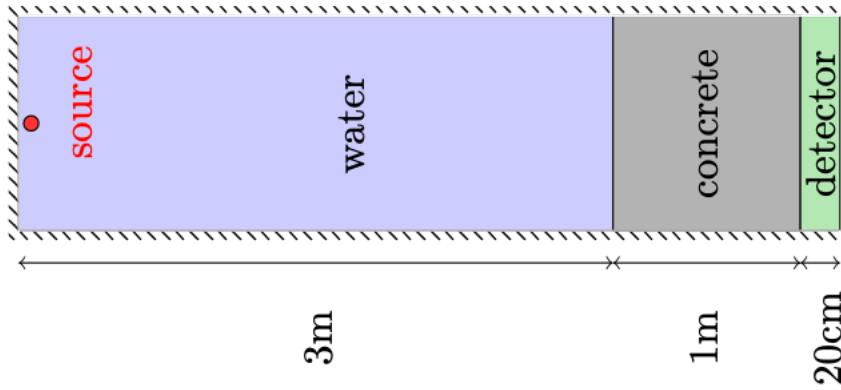
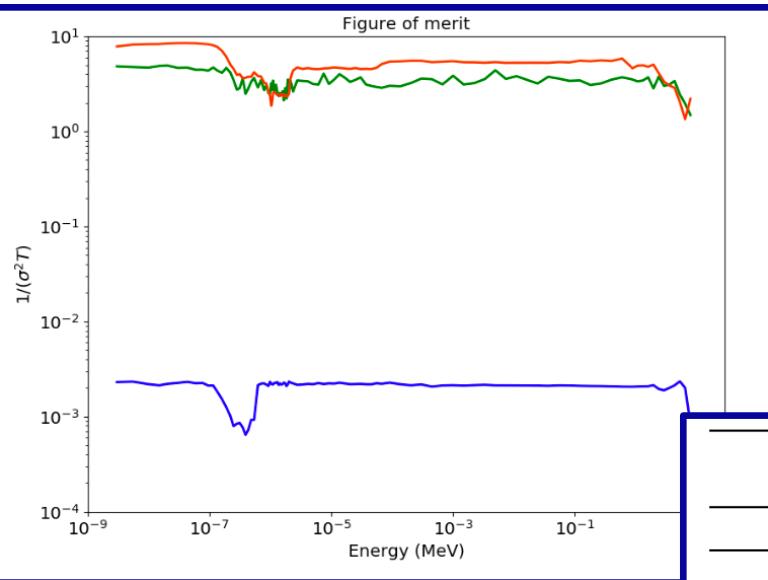
with AMS weights given by the genealogy formula

$$w_{\text{AMS}}(\mathcal{T}) = \frac{1}{N} \prod_{i=0}^j \left(1 - \frac{K_i}{N}\right)$$

$$\text{for } \mathcal{T} \in T^{\text{AMS}} = \left(\bigcup_{q=0}^Q S_q^{\text{off}} \right) \bigcup S_Q^{\text{on}}$$



AMS level →
importance →



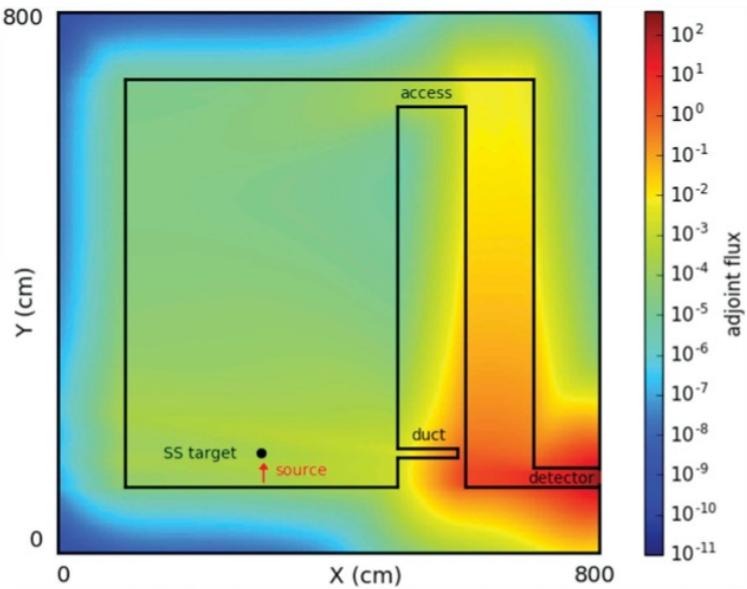
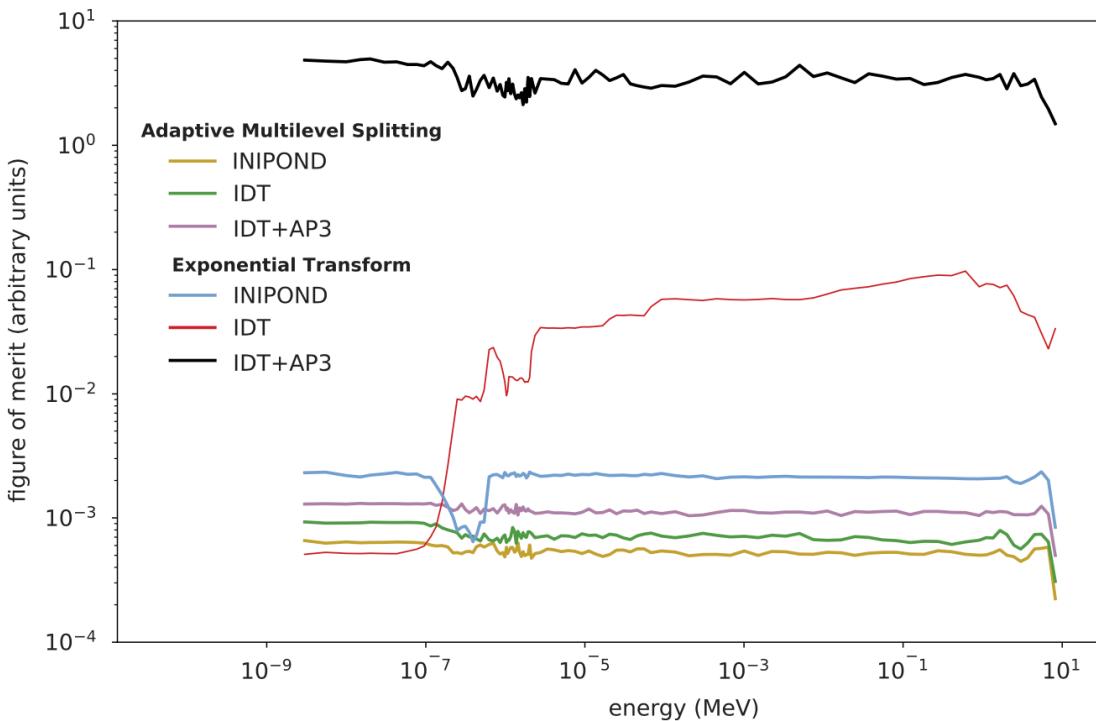
Importance map (properties)	Mean($n/cm^3/s$)	error(%)	time(s)	FOM (arbitrary units)
Adaptive Multilevel Splitting				
INIPOND (Manual mode)	2.58×10^{-15}	9.90	1.67×10^5	9
IDT (cross sections from TRIPOLI4)	2.61×10^{-15}	9.83	1.20×10^5	13
IDT (cross sections from APOLLO3)	2.78×10^{-15}	7.11	1.59×10^5	16
Adjoint score (scored with AMS)	2.66×10^{-15}	9.98	1.08×10^5	13
Exponential Transform				
INIPOND (Manual mode)	2.55×10^{-15}	6.51	9.41×10^4	39
IDT (cross sections from TRIPOLI4)	2.04×10^{-15}	6.60	2.39×10^5	23
IDT (cross sections from APOLLO3)	2.81×10^{-15}	0.82	3.27×10^3	576
Adjoint score (scored with AMS)	2.77×10^{-15}	0.52	4.33×10^3	1313

- **AMS with AMS ξ has a FOM equivalent to AMS with deterministic ξ**
- **ET with AMS ξ has FOM 2 times than ET with deterministic ξ**



- Bunker benchmark**
- (Mancusi et al, 2018)**
- First coupling between CADIS* & AMS**
- Comparison coarsest->finest: INIPOND/IDT/IDT+AP3**

ξ = adjoint flux from IDT deterministic solver



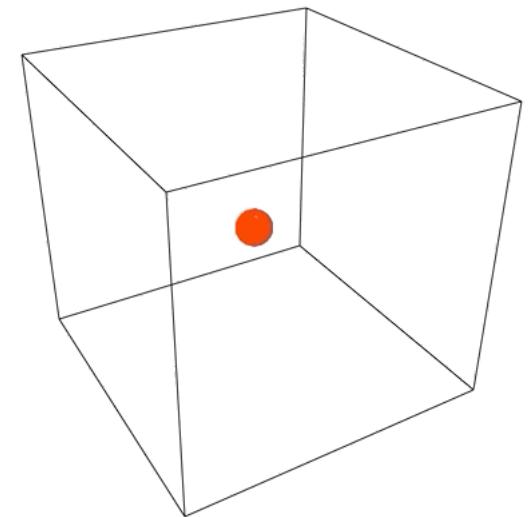
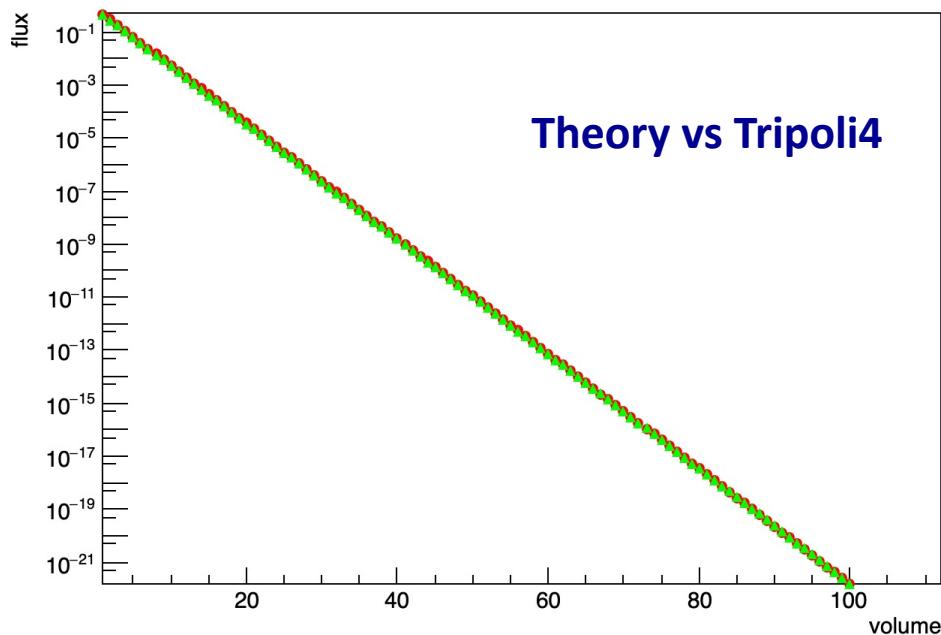
- ET : larger FOM if ξ well-known
- AMS & ET exhibit similar FOM for 'intermediate' ξ

CADIS* = Consistent Adjoint Driven Importance Sampling (Haghhighat, 2003)

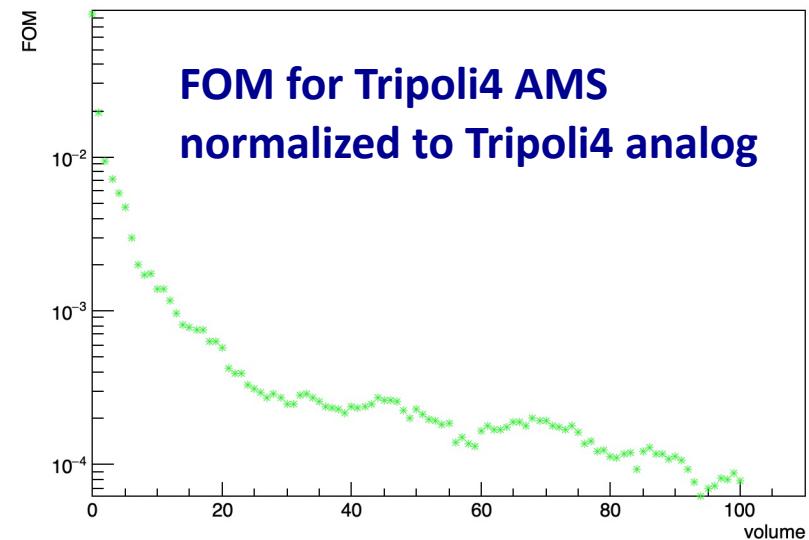
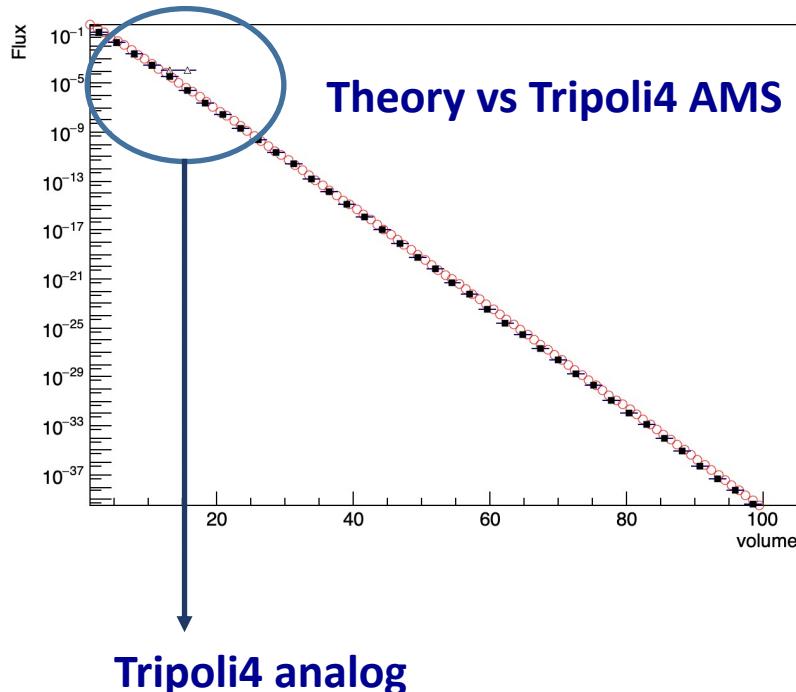
IDT = 3D Cartesian deterministic solver for the multigroup time-independent transport equation (Zmijarevic, 2001)

AP3 = improved adjoint flux wrt anisotropy, upscattering, and energy during cross section condensation (Schneider, 2016)

$$\phi(x) = \frac{3}{4\pi(r_2^3 - r_1^3)} \int_{r_1}^{r_2} \psi(x) = \frac{3}{4\pi(r_2^3 - r_1^3)(p-1)} (e^{-(p-1)\Sigma_t r_2} - e^{-(p-1)\Sigma_t r_1})$$



- Excellent agreement...
- ... up to 10^{-21} !

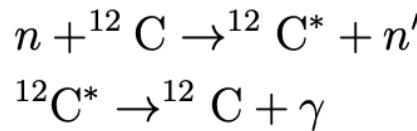


- Excellent agreement...
- ... up to very large attenuations (10^{-37})
- Extremely large FOM (10^{36}) at 1m

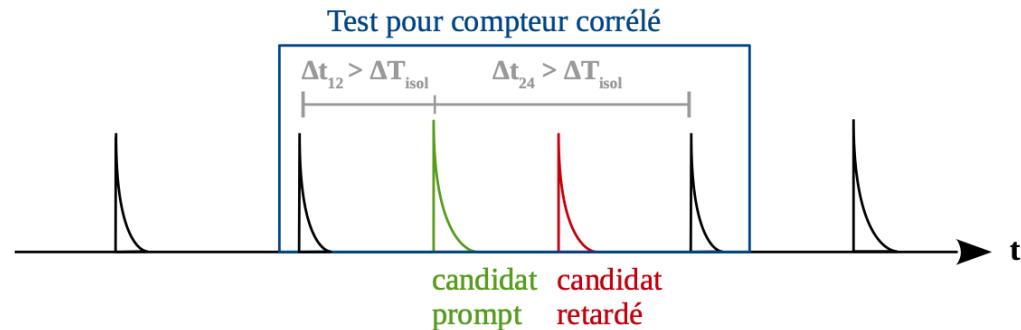
- Multi-particle time-dependant non-linear Boltzmann equation
- Potentially all particles at all energies
- Time correlation detectors => need to preserve correlations
- Main challenge: rare events simulations for (correlated) signal-to-noise studies



Rare backgrounds events
e.g. "C" channel



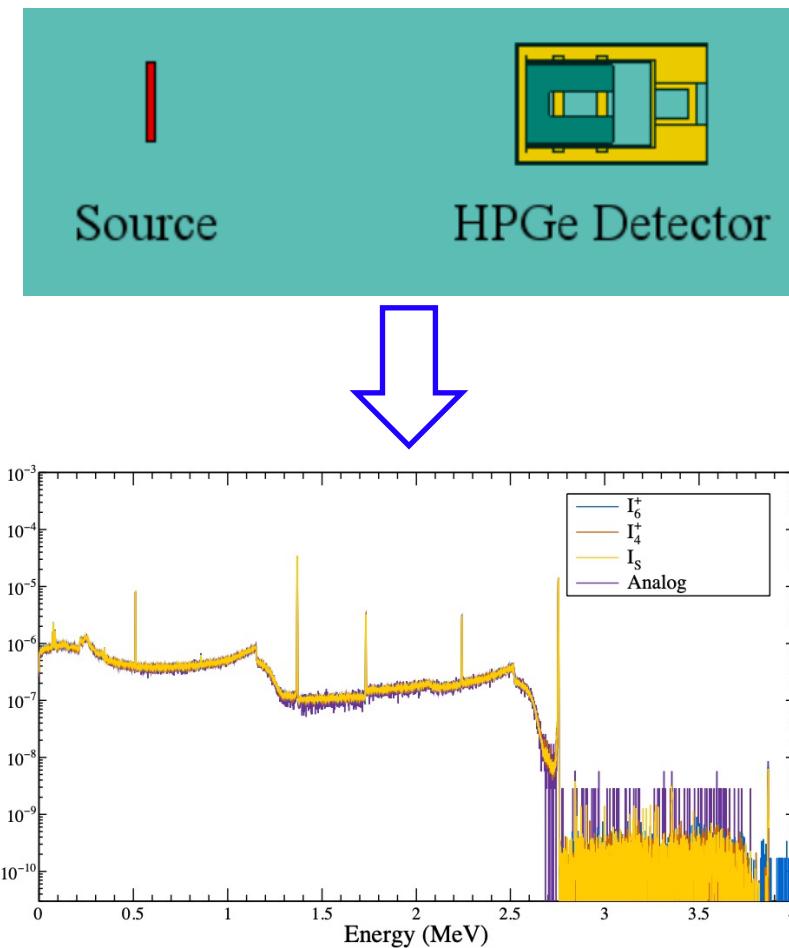
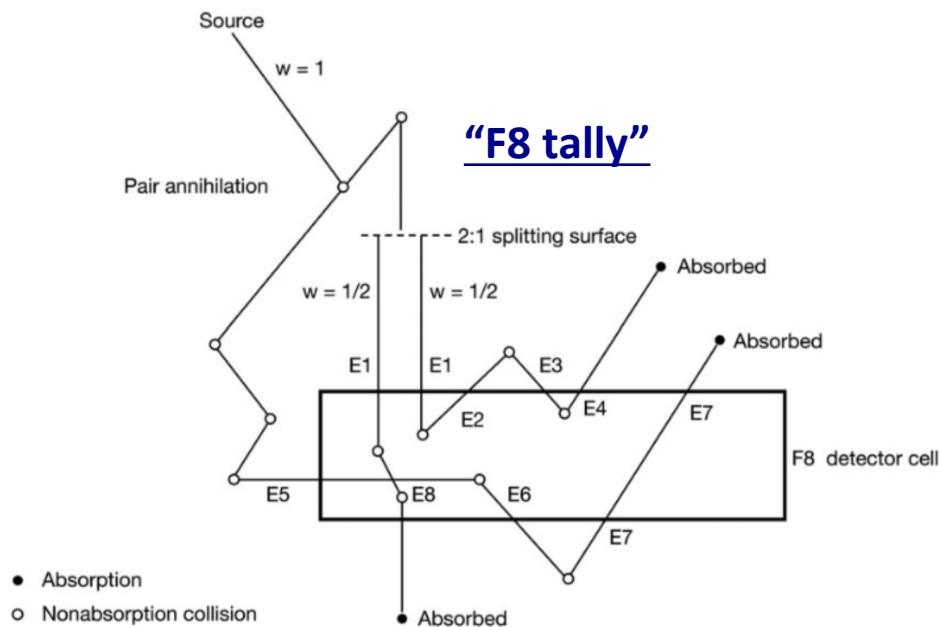
Time detector



(Bonhomme, 2018)

□ Difficulties similar to (n,g) variance reduction in TRIPOLI4

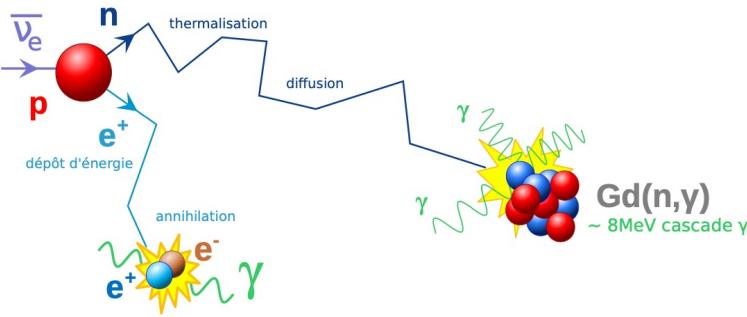
- Particle “conversion”
- Cascades
- Correlations



(Louvin, 2018)

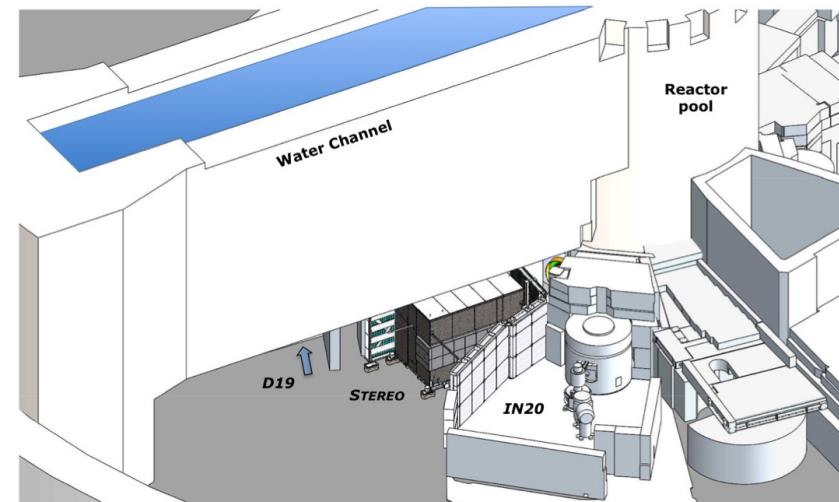
□ Important questions to answer are:

- Where do we loose simulation time ? (simulation of non-contributing channels)
- What are the rare events that contribute the most to signal and/or background ?



Typical STEREO background events :

- $\mu^+ \Rightarrow$ high energy neutron (via spallation)
- ...



(Bonhomme, 2018)

- Adapting AMS Classes to STEREO simulation framework
- Defining a benchmark
 - Muons distributions above ILL as initial conditions
 - Defining detectors
- Working on the "cost function" = "importance function"
 - Trying to figure out the most important processes that contribute to the background
 - Build an cost function => ordering all the contributing processes
- The problem is challenging but from the feedback of (n,g) variance reduction within Tripoli4 it is worth the try

Challenging problem
⇒ Preliminary developments
⇒ Find appropriate cost functions