DE LA RECHERCHE À L'INDUSTRIE

www.cea.fr

Accelerating a Monte Carlo shielding calculation by learning the importance map

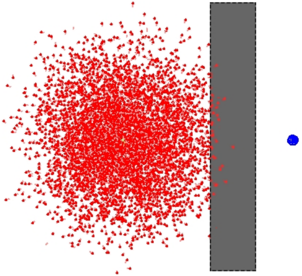
Michel Nowak
Eric Dumonteil[1], Jamal Atif[2],
Davide Mancusi[3]

[1]IRSN

[2]Paris Dauphine LAMSADE

[3]CEA/DEN/DANS/DM2S/SERMA/LTSD

Shielding Monte Carlo



- Estimator of a response $\hat{\phi}_D$ in a detector
- variance on the result σ^2
- Low probability $\simeq 10^{-8}$

$$FOM = \frac{1}{\sigma^2 \cdot T}$$

AMS

Adaptive Multi-level **Splitting**

- real collisions
- warranty of **reaching the detector**
- Function $\mathcal{I}(x)$ to rank the particles

AMS

Adaptive Multi-level **Splitting**

- real collisions
- warranty of **reaching the detector**
- Function $\mathcal{I}(x)$ to rank the particles

Exponential transform

Modified laws of physics

$$\Sigma^*(\mathbf{P}) = \Sigma(\mathbf{P}) - \frac{\vec{\nabla} I(\mathbf{P})}{\left\| \vec{\nabla} I(\mathbf{P}) \right\|} \cdot \Omega(\mathbf{P})$$

$$\vec{\nabla} I(\mathbf{P}) \cdot \Omega(\mathbf{P}) < 0 \implies \boxed{\Sigma^* > \Sigma}$$

$$\vec{\nabla} I(\mathbf{P}) \cdot \Omega(\mathbf{P}) > 0 \implies \boxed{\Sigma^* < \Sigma}$$

Deterministic solver

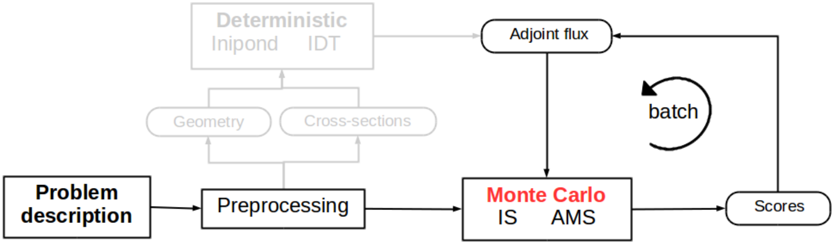
Inipond

- Dijkstra algorithm
- Monitoring of energy groups

IDT

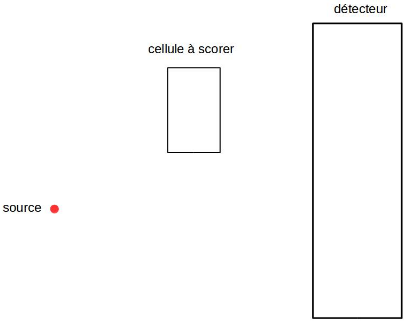
- 2D/3D SN code
- Geometry from mesh
- Multigroup XS

Importance


Deterministic
Inipond IDT
Adjoint flux
batch
Geometry
Cross-sections
Problem description
Preprocessing
Monte Carlo
IS AMS
Scores

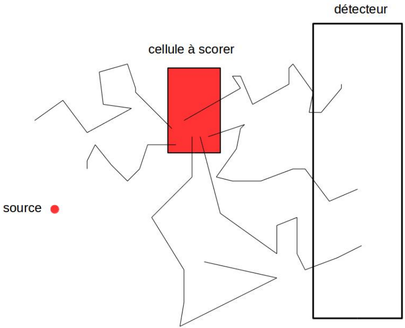
Importance : interpretation physique

Average contribution of a point in the phase space $(\overrightarrow{r}, \overrightarrow{\Omega}, E)$ to a response (flux in a detector).



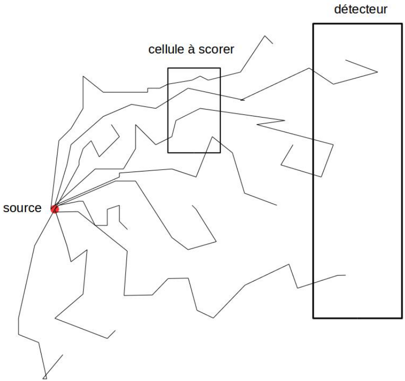
Importance : interpretation physique

Average contribution of a point in the phase space $(\overrightarrow{r}, \overrightarrow{\Omega}, E)$ to a response (flux in a detector).



Importance : interpretation physique

Average contribution of a point in the phase space $(\overrightarrow{r}, \overrightarrow{\Omega}, E)$ to a response (flux in a detector).



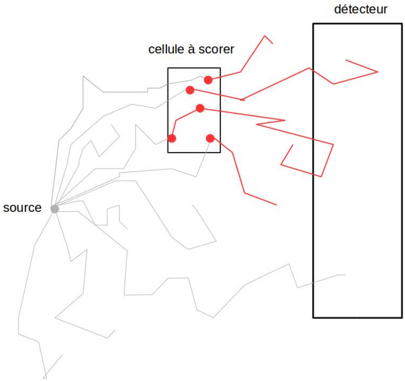
Importance

c_D is the contribution of a point in the phase space $x = \left(\overrightarrow{r}, E, \overrightarrow{\Omega}\right)$

$$\boxed{\mathcal{I}(x) = \mathbb{E}\left(c_D | x\right)}$$
$$= \int c_D \frac{p(c_D, x)}{p(x)}$$
$$= \frac{1}{p(x)} \int c_D p(c_D, x)$$

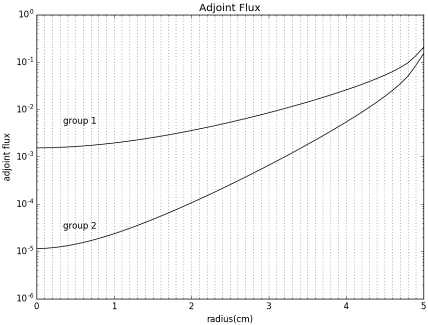
Importance score on a mesh :

$$\hat{\mathcal{I}}(cell) = \frac{\mathbb{E}\left(\frac{1}{N} \sum_{x \in cell} c_D\right)}{\mathbb{E}\left(\frac{1}{N} \sum_{x \in cell} \omega(x)\right)}$$



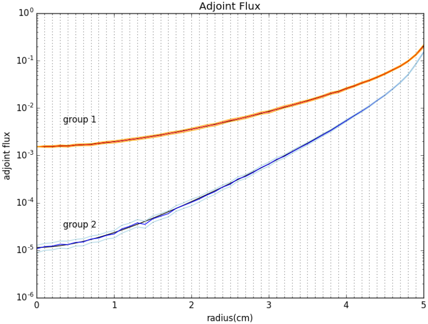
Score the adjoint flux

- Homogeneous sphere
- Tripoli AMS multi-group
  2 groups
- detector : $5cm \leq r \leq 5.2cm$
- score on mesh / volumes
- score COLL

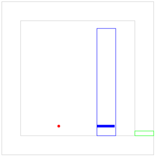


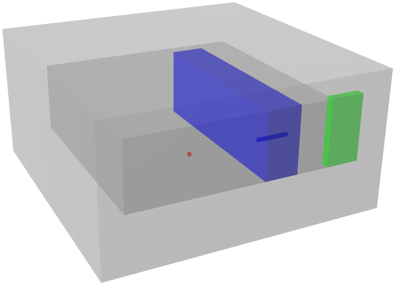
Score the adjoint flux

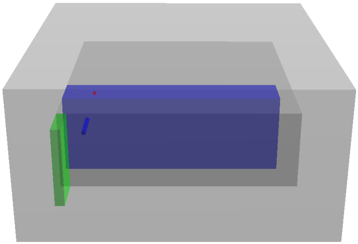
- Homogeneous sphere
- Tripoli AMS multi-group
  2 groups
- detector : $5cm \leq r \leq 5.2cm$
- score on mesh / volumes
- score COLL



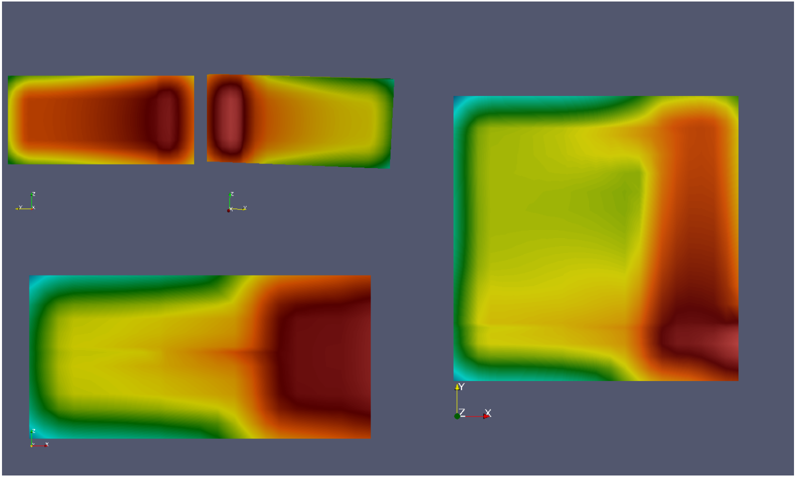
Bunker





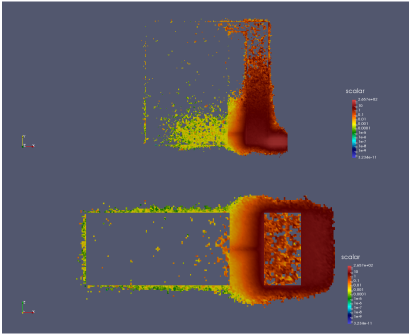


3D Bunker : Idt

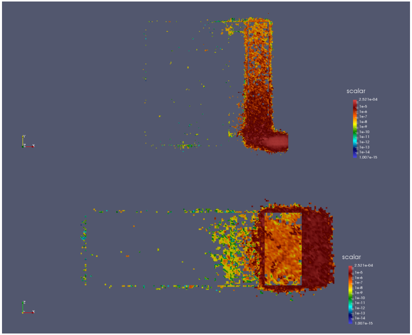


3D Bunker : importance

Geometrical map



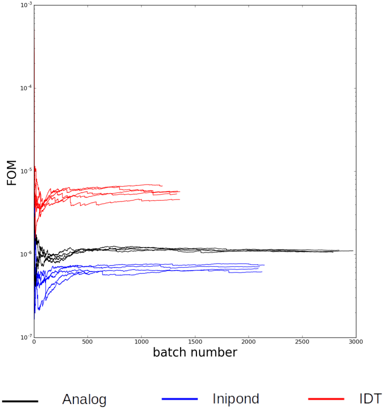
Inipond map



3D Bunker : FOM

Figure of merit

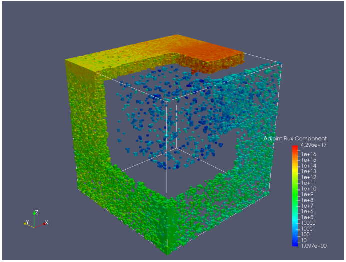
Time FOM : $\frac{1}{\sigma^2 T}$



Why learning?

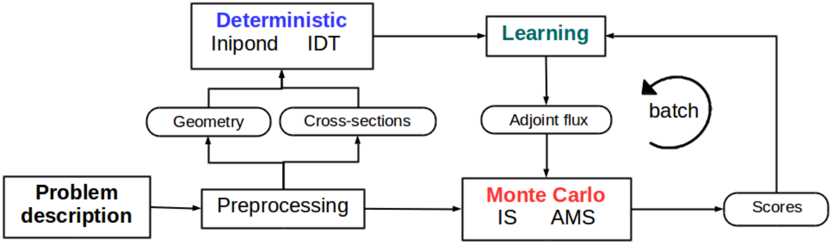
What could be improved

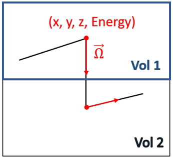
- Holes in the importance score
- Discretisation choices (direction, or energy)
- Difficult convergence of deterministic solvers for streaming problems



Motivations

- Improved FOM with IDT importance map


Deterministic
Inipond IDT
Learning
Geometry
Cross-sections
Adjoint flux
batch
Problem description
Preprocessing
Monte Carlo
IS AMS
Scores



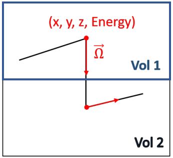
Points

N collisions points $coll^{(i)}$

phase space :

x, y, z , E, Ω

target : $contribution^{(i)}$



Points

N collisions points $coll^{(i)}$

phase space :

x, y, z , E, Ω

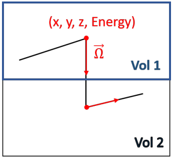
target : $contribution^{(i)}$

Features

N observations $x^{(i)}$

6 features

target : $y^{(i)}$



Points

N collisions points $coll^{(i)}$

phase space :

x, y, z , E, Ω

target : $contribution^{(i)}$

Features

N observations $x^{(i)}$

6 features

target : $y^{(i)}$

Model

Model, parameters θ : f_θ

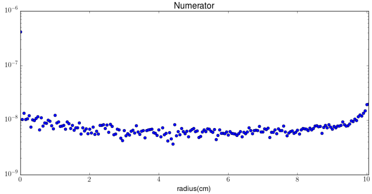
Observation loss :

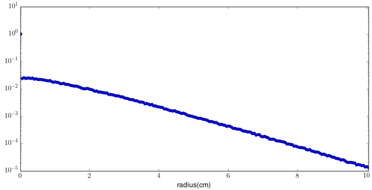
$$\mathcal{L}_\theta(x^{(i)}, y^{(i)}) = \left[y^{(i)} - f_\theta\left(x^{(i)}\right)\right]^2$$

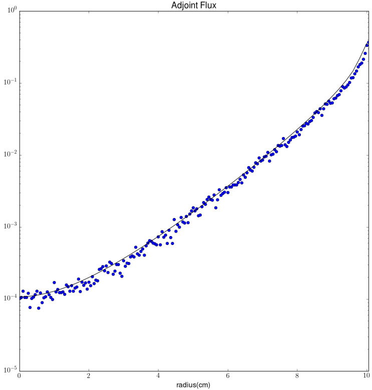
Optimisation target :

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_\theta(x^{(i)}, y^{(i)})$$

Learning Sphere







$$\hat{\mathcal{I}}(x) = \frac{f_\theta^{\text{contributions}}(x)}{exp\left(f_\theta^{\text{coll density}}(x)\right)}$$

$f_\theta^{\text{coll density}}$ fitted on $log(y)$

Choosing the learning algorithm

Nature of the problem :

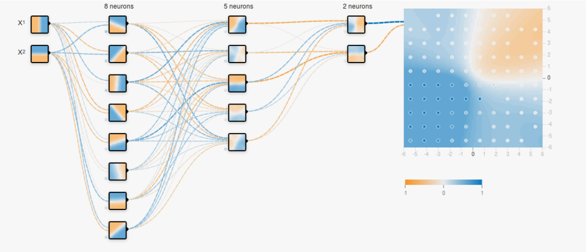
1. large dataset : $10k$ of observations per second
2. lots of noise

Requirements :

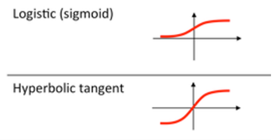
1. online algorithm
2. fast training
3. fast prediction of the importance at each collision

Candidates :

1. Neural Networks
2. Decision Trees



Activation



Gradient descent

learning rate : η

$$\theta \leftarrow \theta - \eta \nabla_\theta J(\theta)$$

Optimisation

- AdaGrad
- AdaDelta
- AdaBoost
- RMSProp
- ...

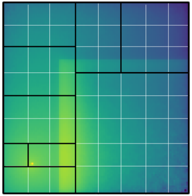
Decision Trees

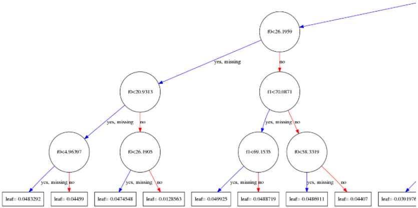
First define a set of **split candidates**

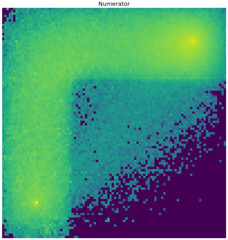
$$\hat{y}_i = \sum_{t=0}^{T} f_t(x_i)$$

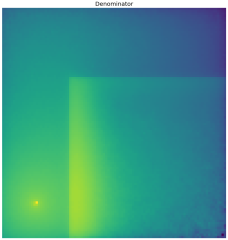
Optimise nodes according to :

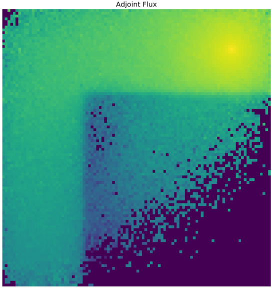
$$J^{(t)} \simeq \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{y}^{t-1} + f_t(x_i)) + \Omega(f_t)$$





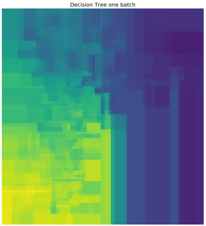

Numerator

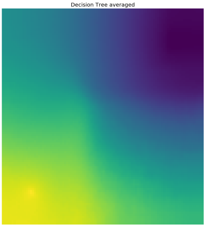

Denominator


Adjoint Flux

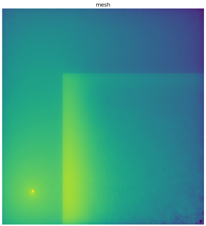
Wafe prediction

Decision Trees

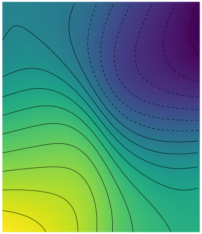


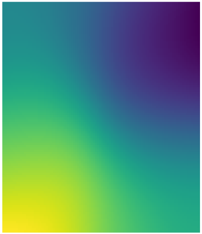


Mesh



Neural Networks





Conlusions & Perspectives

Conclusions

- Importance from **IDT**
- Importance from **Mesh**
- Importance from **Models (N. Nets, Decision Trees)**
- **Averaging** of non-online models on mesh

Perspectives

- TRACK estimator
- Hyperoptimization
- Initialisation of model with IDT map

Difficulties

- AMS stops when bad importance is predicted

Bunker tracks