Accelerating a Monte Carlo shielding calculation by learning the importance map

Michel Nowak
Eric Dumonteil\textsuperscript{1}, Jamal Atif\textsuperscript{2}, Davide Mancusi\textsuperscript{3}

\textsuperscript{1}IRSN

\textsuperscript{2}Paris Dauphine LAMSADE

\textsuperscript{3}CEA/DEN/DANS/DM2S/SERMA/LTSD
• Estimator of a response $\hat{\phi}_D$ in a detector
• variance on the result $\sigma^2$
• Low probability $\sim 10^{-8}$

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

$$\Sigma^*(P) = \Sigma(P) - \frac{\nabla \mathcal{I}(P)}{\| \nabla \mathcal{I}(P) \|} \cdot \Omega(P)$$

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

$$\nabla \mathcal{I}(P) \cdot \Omega(P) > 0 \implies \Sigma^* < \Sigma$$
Deterministic solver

**Inipond**
- Dijkstra algorithm
- Monitoring of energy groups

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

Michel Nowak
Accelerating a Monte Carlo shielding calculation by learning the importance map
Importance

Michel Nowak

Accelerating a Monte Carlo shielding calculation by learning the importance map
Average contribution of a point in the phase space ($\vec{r}$, $\Omega$, $E$) to a response (flux in a detector).
Average contribution of a point in the phase space ($\vec{r}$, $\vec{\Omega}$, $E$) to a response (flux in a detector).
Average contribution of a point in the phase space \((\vec{r}, \Omega, E)\) to a response (flux in a detector).
Importance

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

$$\mathcal{I}(x) = \mathbb{E}(c_D | x)$$

$$= \int c_D \frac{p(c_D, x)}{p(x)}$$

$$= \frac{1}{p(x)} \int c_D p(c_D, x)$$

Importance score on a mesh:

$$\hat{I}(\text{cell}) = \frac{\mathbb{E}\left(\frac{1}{N} \sum_{x \in \text{cell}} c_D\right)}{\mathbb{E}\left(\frac{1}{N} \sum_{x \in \text{cell}} \omega(x)\right)}$$
Score the adjoint flux

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

![Graph showing adjoint flux vs. radius](image)
- Homogeneous sphere
- Tripoli AMS multi-group 2 groups
- detector: $5cm \leq r \leq 5.2cm$
- score on mesh / volumes
- score COLL

![Graph showing adjoint flux vs radius](image)
Accelerating a Monte Carlo shielding calculation by learning the importance map

Michel Nowak
Accelerating a Monte Carlo shielding calculation by learning the importance map
3D Bunker: importance

Geometrical map

Inipond map

Michel Nowak
Accelerating a Monte Carlo shielding calculation by learning the importance map
Figure of merit

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

Michel Nowak

Accelerating a Monte Carlo shielding calculation by learning the importance map
**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
Learning

Michel Nowak

Accelerating a Monte Carlo shielding calculation by learning the importance map
Points

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

phase space:
$x, y, z, E, \Omega$

target: $contribution^{(i)}$
Learning

Points
N collisions points \( \text{coll}^{(i)} \)
phase space:
\( x, y, z, E, \Omega \)
target: \( \text{contribution}^{(i)} \)

Features
N observations \( x^{(i)} \)
6 features
target: \( y^{(i)} \)
Points
N collisions points $coll^{(i)}$
phase space : $x, y, z, E, \Omega$
target : contribution$^{(i)}$

Features
N observations $x^{(i)}$
6 features
target : $y^{(i)}$

Model
Model, parameters $\theta : f_\theta$
Observation loss :
$$L_\theta(x^{(i)}, y^{(i)}) = \left[ y^{(i)} - f_\theta \left( x^{(i)} \right) \right]^2$$

Optimisation target :
$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} L_\theta(x^{(i)}, y^{(i)})$$

Michel Nowak
Accelerating a Monte Carlo shielding calculation by learning the importance map
\[ \hat{L}(x) = \frac{f^\text{contributions}_\theta(x)}{\exp\left(f^\text{coll\ density}_\theta(x)\right)} \]

\[ f^\text{coll\ density}_\theta \text{ fitted on } \log(y) \]
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
Neural Networks

Activation
- Logistic (sigmoid)
- Hyperbolic tangent

Gradient descent
- learning rate: $\eta$
- $\theta \leftarrow \theta - \eta \nabla_{\theta} J(\theta)$

Optimisation
- AdaGrad
- AdaDelta
- AdaBoost
- RMSProp
- . . .
First define a set of **split candidates**

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

Optimise nodes according to:

\[ J^{(t)} \simeq \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{y}^{t-1} + f_t(x_i)) + \Omega(f_t) \]
Wafe importance on mesh

Michel Nowak

Accelerating a Monte Carlo shielding calculation by learning the importance map
Michel Nowak

Accelerating a Monte Carlo shielding calculation by learning the importance map
Conclusions & 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

Analog

Inipond

AMS

Accelerating a Monte Carlo shielding calculation by learning the importance map