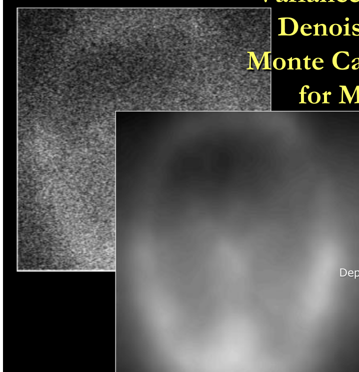
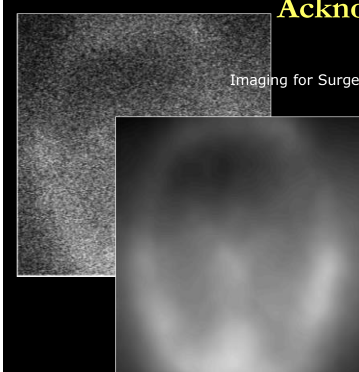
Variance Reduction and Denoising Methods in Monte Carlo Simulations for Medical Imaging



W. Zbijewski
Department of Biomedical Engineering
Johns Hopkins University

Acknowledgements

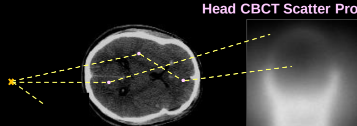


The I-STAR Laboratory
Imaging for Surgery, Therapy, and Radiology
http://istar.jhu.edu/

Funding Support
Carestream Health
NIH 2R01-CA-112163
NIH R21-AR-062293

Overview

MC-GPU
Head CBCT Scatter Projection



10^{10} photons
8.5 min / projection
(on nVidia GTX 780 Ti)

Analog Monte Carlo
- Signal-to-noise ratio (SNR) proportional to $\sqrt{\text{number of tracked particles}}$
- More particles = better SNR
- More particles = longer simulation time
- Achieving low noise MC estimates can be prohibitively long even on a GPU

Algorithmic acceleration strategies
- Variance Reduction (VR): modifies the tracking process (avoids bias)
- De-noising: typically post-processing (may introduce bias)
- Combination with analytical methods
- We will focus on MC scatter estimation in x-ray imaging
- Methods also applicable to dose scoring

Variance Reduction

Increase the number of events contributing to the signal

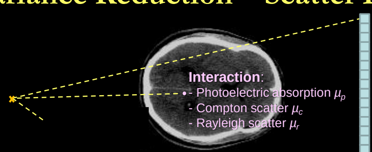
General approach:
- Modify the probability distribution
- Increase the fraction of histories leading to "detection"
- Include a weight to avoid biasing the signal

Analog MC*:
- *X*: space of photon histories
- *p(x)*: probability density on space *X*
- *S*: mean signal

$$\bar{S} = \int_x S(x)\,p(x)\,dx$$

*DR Haynor RL Harrison, *Variance Reduction Techniques* in *Monte Carlo Calculations in Nuclear Medicine*, 2nd edition, CRC Press 2013 (eds M Ljungberg, SE Strand, MA King)

Variance Reduction – Scatter Forcing



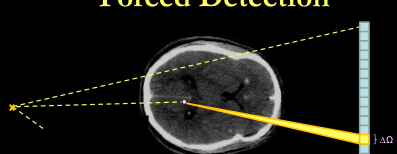
Interaction:
- Photoelectric absorption μ_p
- Compton scatter μ_c
- Rayleigh scatter μ_r

Analog MC probability distribution
- *p(x)*:
  - [absorption, scatter] = $[\mu_p / (\mu_p + \mu_c + \mu_r), (\mu_c + \mu_r) / (\mu_p + \mu_c + \mu_r)]$
- Photoelectric absorption does not contribute to the signal

Let's force the photon to scatter
- *q(x)*:
  - [absorption, scatter] = [0, 1]
- We need to weight the photons after forcing scatter:
  - $w = p(x) / q(x) = (\mu_c + \mu_r) / (\mu_p + \mu_c + \mu_r)$

Intuition about the weights
- Weight by the "analog" probability of the history

Forced Detection



Measure scatter at a particular detector cell
- Only small fraction of events contributes to the signal
- Even worse if seeking distribution at a point (point detectors)
- Also applies to dose scoring

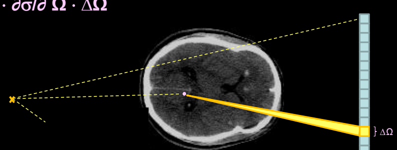
Let's force towards the detector cell
- Pick scatter angle within the solid angle of the cell ($\Delta\Omega$)
- Weight:
  - Ratio of events scattered towards the detector vs. events scattered into 4π
  - Scatter cross section over $\Delta\Omega$ vs. total scatter cross section
- Followed by sending the particle directly to the detector (ray-tracing)

Consider a VR technique where at each Compton interaction, a virtual scattered photon is created and forced towards a fixed detector cell. Let:
- σ be the total cross section for Compton interaction.
- $\partial\sigma/\partial\Omega$ be the differential cross section for the direction towards the center of the detector cell (assumed constant across the cell).
- $\Delta\Omega$ be the solid angle covered by the detector cell as seen from the point of interaction.

What weight needs to be assigned to the virtual photon to avoid bias in the resulting scatter distribution:

- 19% A. No weight.
- 13% B. $\partial\sigma/\partial\Omega \cdot \Delta\Omega$
- 6% C. Bias cannot be avoided by weighting
- 38% D. $\sigma / \Delta\Omega$
- 25% E. $\sigma^{-1} \cdot \partial\sigma/\partial\Omega \cdot \Delta\Omega$

Answer:
(e) $\sigma^{-1} \cdot \partial\sigma/\partial\Omega \cdot \Delta\Omega$



Forced detection*

Probability that a photon would scatter into $\Delta\Omega$ in analog simulation:

$$w_{FD} = \underbrace{\partial\sigma/\partial\Omega}_{\text{Scatter towards the detector cell}} \cdot \underbrace{\Delta\Omega}_{\text{Angular "coverage" of the detector cell}} \;/\; \underbrace{\sigma}_{\text{Scatter into } 4\pi}$$

(assumes differential cross section does not change much throughout the detector cell)

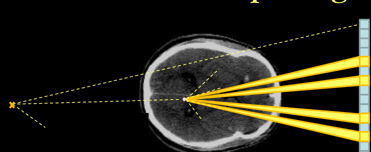
Next step – send the photon directly to the detector

Weight = attenuation along the path (ray tracing)

$$w = w_{FD} \cdot e^{-\int \mu(x)dx}$$

*JF Williamson, *Monte Carlo evaluation of kerma at a point for photon transport problems*, Med Phys 14 (1987)
CJ Leliveld, *A fast Monte Carlo simulator for scattering in X-ray Computerized Tomography*, PhD Thesis, TU Delft, Netherlands (1996).

Interaction Splitting



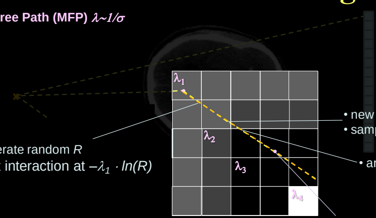
Create additional particles at each interaction
- Split into N_p photons
- Weight each by $1/N_p$
- Track all new particles

*AP Colijn and FJ Beekman, *Accelerated simulation of cone beam X-ray scatter projections.*, IEEE TMI 23 (2004)

Woodcock Tracking

*E Woodcock et al, *Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry*, Argonne Nat. Lab. Report (1965)
J Sempau et al *DPM, a fast, accurate Monte Carlo code optimized for photon and electron radiotherapy treatment planning dose calculations*, PMB 45 (2000)

Mean Free Path (MFP) $\lambda \sim 1/\sigma$



- generate random *R*
- next interaction at $-\lambda_1 \cdot ln(R)$

- new material (λ_2)
- sample again

- and again…

- interaction at $-\lambda_4 \cdot ln(R)$
- virtual interaction with no change in motion $P=1- \lambda_4/\lambda_3$

Analog Monte Carlo
- Resampling of the path length at each material boundary
- Inefficient in voxelized geometries

Woodcock Tracking (δ scattering)*
- Treat whole volume as made of the most attenuating material - shorter "steps"
- No boundary checking
- Virtual interaction accounts for difference between true MFP and the maximal MFP
- Implemented in MC-GPU
- Challenging when highly attenuating materials (implants) are present

Other VR Techniques

Mean Free Path Transformations*
- Increase number of interactions in a region
- Change the probability distribution function of number of MFP till next interaction
  - Stretch the paths – more interactions deeper
  - Change path length depending on travel direction (exponential transforms)**
  - Force interaction to happen within a volume**
- Weight the photon

Russian Roulette*
- To remove particles from tracking without bias
  - Combined with splitting – remove photons not pointing towards the detector
- Randomly decide if a particle should be terminated or survive
  - Probability of particle being terminated: *P*
  - Weight if particle survives: *1/(1-P)*
- Weight for surviving particles > 1!

*E Mainegra-Hing and I Kawrakow, *Variance reduction techniques for fast Monte Carlo CBCT scatter correction calculations*, PMB 55 (2010)
people.physics.carleton.ca/~drogers/pubs/papers/RB90.pdf
**DWO Rogers and AF Bielajew, *Monte Carlo Techniques of Electron and Photon Transport for Radiation Dosimetry* in *The Dosimetry of Ionizing Radiation*, Academic Press 1990 (eds KR Kase, BE Bjarngard FH Attix)

MC Simulation Efficiency

Computational cost of VR
- Additional arithmetic (weights)
- Potentially breaks parallelism (Interaction Splitting)
- Many events with low contribution to signal (Scatter Forcing)
- Widely fluctuating weights (introduces noise)

Simulation time vs. Signal-to-Noise Ratio (SNR)
- Efficiency*:

  $\varepsilon = 1 / (\sigma^2 \cdot T)$
  σ^2: variance
  T: simulation time

- Inverse variance per unit time
- Higher value – better efficiency
- To improve efficiency:
  - $\sigma^2\downarrow$ or $T\downarrow$, but it's the product that counts
- Other definitions (involving SNR) exist - similar meaning

*people.physics.carleton.ca/~drogers/pubs/papers/RB90.pdf
DWO Rogers and AF Bielajew, *Monte Carlo Techniques of Electron and Photon Transport for Radiation Dosimetry* in *The Dosimetry of Ionizing Radiation*, Academic Press 1990 (eds KR Kase, BE Bjarngard FH Attix)

Consider:
-Analog MC (aMC) simulation with a runtime of 300 sec.
-VR technique A that reduces the variance (for the same signal level) by a factor of 2 with a runtime of 600 sec
-VR technique B that reduces the variance by a factor of 3 with a runtime of 1000 sec.

Which of the statements is true:

- 28% A. Technique A improves efficiency compared to aMC
- 24% B. Technique B improves efficiency compared to aMC
- 14% C. Neither of the VR techniques improves efficiency over aMC
- 7% D. Efficiency of technique A is 2/3 of efficiency of technique B
- 17% E. Technique B has higher efficiency than technique A

Answer:
(c) Neither of the techniques improves efficiency over analog MC

Using $\varepsilon = 1 / (\sigma^2 \cdot T)$ defined previously*:

Analog simulation:
Variance σ_0^2
$T_0 = 300$ s
$\varepsilon_0 = 1 / (\sigma_0^2 \cdot 300 \text{ s})$

Variance reduction A:
$\sigma_A = \sigma_0^2/2$
$T_A = 600$ s
$\varepsilon_A = 1 / ((\sigma_0^2 / 2) \cdot 600 \text{ s}) = 1 / (\sigma_0^2 \cdot 300 \text{ s})$

Variance reduction B:
$\sigma_B = \sigma_0^2/3$
$T_B = 1000$ s
$\varepsilon_B = 1 / ((\sigma_0^2 / 3) \cdot 300 \text{ s}) = 1 / (\sigma_0^2 \cdot 333.3 \text{ s})$

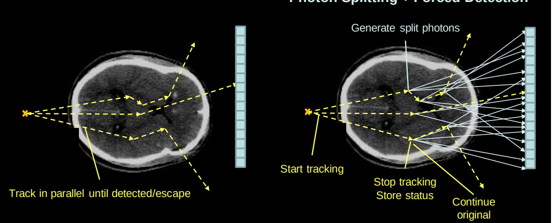
$\varepsilon_A = \varepsilon_0$
$\varepsilon_B < \varepsilon_0$
$\varepsilon_A > \varepsilon_B$

*people.physics.carleton.ca/~drogers/pubs/papers/RB90.pdf
DWO Rogers and AF Bielajew, *Monte Carlo Techniques of Electron and Photon Transport for Radiation Dosimetry* in *The Dosimetry of Ionizing Radiation*, Academic Press 1990 (eds KR Kase, BE Bjarngard FH Attix)

Variance Reduction on GPUs

Analog MC- GPU

Variance Reduction in MC – GPU
Photon Splitting + Forced Detection



Efficient GPU implementation
Simple parallelism
1 GPU thread per photon

Challenges for GPU implementation
Tracking split into stages
Breaks parallelism
Cost in context switch
Complex memory management

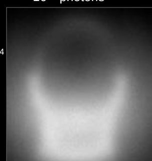
*A Sisniega et al, *Ultra-Fast Monte Carlo Simulation for Cone Beam CT Imaging of Brain Trauma*, AAPM 2014, TH-A-18C-9

Scatter Estimation with VR

Analog MC-GPU
10^{10} photons

MC-GPU + Variance Reduction
10^7 photons – 512 split

Time = 510 s
Noise = 4.8×10^{-4}
Efficiency = 0.9



Time = 80 s
Noise = 3.5×10^{-4}
Efficiency = 9.7

GPU-accelerated MC with VR*
- Optimized combination of photon splitting and variance reduction
- ~10 fold improvement in efficiency over analog MC-GPU (depends on object/geometry)

VR for MC executed on a CPU (Mainegra-Hing 2010)**
- Forced detection, Woodcock tracking, path length transformations
- Splitting and Russian Roulette
  - Region dependent importance splitting
- 2-orders of magnitude improvement in efficiency
- Relies heavily on splitting – may not be suited for GPU

*A Sisniega at al, *Monte Carlo study of the effects of system geometry and antiscatter grids on cone-beam CT scatter distributions*, Med Phys 40 (2013)
**E Mainegra-Hing and I Kawrakow, *Variance reduction techniques for fast Monte Carlo CBCT scatter correction calculations*, PMB 55 (2010)

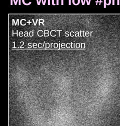
Beyond Variance Reduction

Assumptions
- Distribution of interest (scatter, dose) is "smooth"
- De-noising does not introduce significant bias

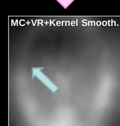
Examples (scatter estimation)
- Colijn et al (IEE TMI 2004)
  - Richardson-Lucy fit
  - Gaussian kernels
- Jarry et al (SPIE 2006)
  - Polynomial fitting
- Bootsma et al (Med Phys 2013)
  - Frequency domain filtering
  - Butterworth filter
- Zbijewski et al (SPIE 2013)
  - Kernel smoothing
  - Gaussian kernels
- Thing and Mainegra-Hing (Med Phys 2014)
  - Adaptive Savitzky-Golay filter
  - egs_cbct
- Xu et al (AAPM 2014)

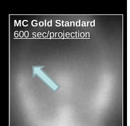
Challenges
- Noise reduction vs. bias

MC with low #photons – { **Fast**, **Noisy** }



De-noise





Which of these acceleration methods are NOT considered MC Variance Reduction Techniques:

4%	A.	Forced Detection
4%	B.	Projection De-noising
13%	C.	Interaction Splitting
13%	D.	Scatter Forcing
13%	E.	Woodcock Tracking

Answer:
(b) Projection De-noising

Variance reduction
- Weights are applied to avoid bias
- We discussed:
  - Forced Detection, Interaction Splitting, Scatter Forcing and Woodcock Tracking

De-noising
- Generally cannot guarantee that no bias will be introduced*
- Acceptable bias may depend on application
- Brute force optimization (parameter sweep)
- Theoretical considerations*

*RS Thing and E Mainegra-Hing, *Optimizing cone beam CT scatter estimation in egs_cbct for a clinical and virtual chest phantom*, Med Phys 41 (2014).
I Kawrakow *On the de-noising of Monte Carlo calculated dose Distributions*, PMB 47 (2002).

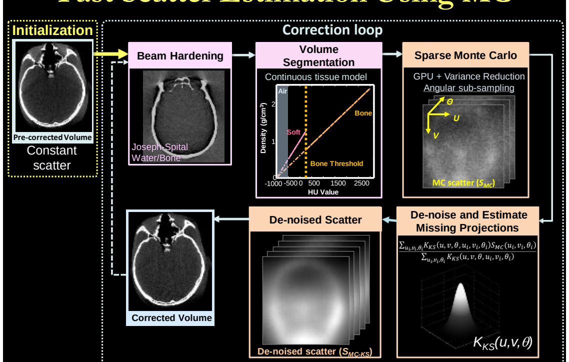
Other Acceleration Strategies

Sparse Detector Sampling / Fixed Forced Detection (Poludniowski 2009*)
- Detector as a sparse, fixed grid of nodes (8x8 nodes)
  - Each interaction → forced detection into all nodes
- Sparse sampling in projection angles
- Interpolation in-plane and between views
- Correction in ~2 min per scan for on-board CBCT

Combination with analytical simulation (Kyriakou 2006*)
- Analytical simulation (deterministic) for 1st order scatter
- Monte Carlo (with Forced Detection and optional smoothing) for higher order scatter
  - Higher order scatter is smooth, only about 30% of total scatter intensity
  - Low #photons in MC for higher order scatter
- ~0.5 min/projection for chest C-arm CBCT

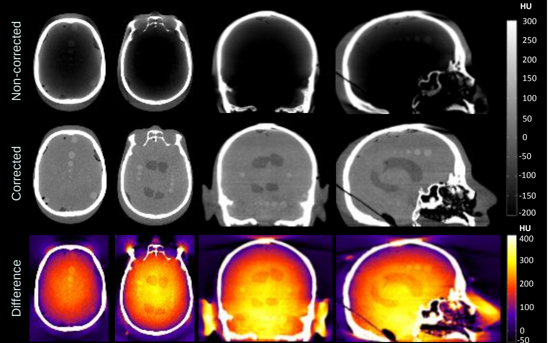
*G Poludniowski et al, *An efficient Monte Carlo-based algorithm for scatter correction in keV cone-beam CT*, PMB 54 (2009)
**Y Kyriakou et al, *Combining deterministic and Monte Carlo calculations for fast estimation of scatter intensities in CT*, PMB 51 (2006)

Fast Scatter Estimation Using MC



*A Sisniega et al, *Ultra-Fast Monte Carlo Simulation for Cone Beam CT Imaging of Brain Trauma*, AAPM 2014, TH-A-18C-9

MC-based Scatter Correction in CBCT



Running time: 3.5 min/iteration - 3 iterations

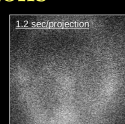
Summary and Conclusions

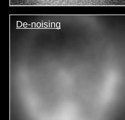
Algorithmic acceleration techniques for MC
- Improve SNR per unit time
- Essential for "real-time" MC applications
  - Scatter estimation and correction

Variance Reduction
- Alter the tracking to generate more events of interest
- Weighting to avoid bias
- Well-established methodology
- Implemented in some MC packages
- 1-2 orders of magnitude acceleration (gain in efficiency)

De-noising
- May introduce bias
- More efficient when quantity of interest is smooth
- Another ~1 order of magnitude acceleration

Applications
- Correction of scatter in 1-10 min/scan
- Real-time dose estimation





Fast MC scatter correction