GPU-BASED PARTICLE TRANSPORT FOR PET RECONSTRUCTION

A dissertation submitted to
the Budapest University of Technology and Economics
in fulfillment of the requirements
for the degree of Doctor of Philosophy (Ph.D)

by

Magdics Milán

Department of Control Engineering and Information Technology
Budapest University of Technology and Economics

advisor
Szirmay-Kalos László

2014

Acknowledgements

This research work has been carried out at the Department of Control Engineering and Information Technology of the Budapest University of Technology and Economics in close collaboration with Mediso Medical Imaging Systems and was supported by OTKA K-104476 (Hungary). This work could not have been completed without the help of several great people to whom I wish to express my sincere thanks.

I would like to express my deepest gratitude to my supervisor, Prof. László Szirmay-Kalos, for his excellent guidance, endless support and admirable patience. He taught me how to think as a researcher and how to work effectively as an engineer. I have learned many invaluable lessons from him, which I will always try to follow during my future career.

I would like to express my sincere thanks to the people at Mediso Medical Imaging Systems for the fruitful and friendly collaboration within the Tera-Tomo project and for their numberless inspiring comments. GATE simulations and PET measurements used by this thesis work were carried out at Mediso Mediso Imaging Systems by Tamás Bükki, Balázs Domonkos, Judit Lantos, Győző Egri, Gergő Patay, Dávid Völgyes and Gábor Jakab; their kind aid is gratefully acknowledged. I am also thankful for the great sandwiches they brought for the project meetings.

I am much obliged to my fellow labmates in the Computer Graphics Group at the Budapest University of Technology and Economics, László Szécsi, Tamás Umenhoffer, Balázs Tóth and Balázs Csébfalvi, who provided an excellent atmosphere in the lab, it has been a great pleasure to work with them. It felt like having additional advisors, as they always had some time to share their expertise with me. A special thanks to Balázs Tóth for having been the "volunteer" who managed and maintained the whole research infrastructure of the lab (and also for the infinite amount of hilarious videos he had for cheering us up).

I am grateful to Ádám Csendesi, former MSc student at the Budapest University of Technology and Economics, who helped a lot in implementing the first prototype of the scattering simulation method.

I would like to thank Mateu Sbert from the Graphics & Imaging Laboratory of the University of Girona for freeing me from my other tasks while I was concentrating on writing this dissertation.

I would also like to thank Sándor Fridli from the Numerical Analysis Department of the Eötvös Loránd University for supporting me during the first semester of my Ph.D studies.

I wish to express my loving thanks to my parents for their continuous support and care during all these years. They have always believed in me much more than I have ever believed in myself.

Finally but not lastly, I would like to thank my friends, who have kindly understood that I could not host them in my temporary home near the wonderful Costa Brava while I was focusing on this dissertation.

Contents

Chapter 1

Introduction

During Positron Emission Tomography measurement, particles are emitted, all bouncing along different paths until they are finally detected by one of the numerous detectors of the scanner. Consequently, the reconstruction process of finding the spatial distribution of particle emissions is formalized by hundreds of millions of equations having high-dimensional integrals as coefficients. In order to work in clinical practice, traditional methods reduce the resulting computational burden by restricting themselves to an idealized case, simplifying or completely neglecting important physical phenomena. With the rapid evolution of PET scanners, this simplified model has become the main limitation of spatial resolution.

Fortunately, computational throughput of processors is evolving even faster making physically plausible models more and more feasible. In the past few years, the graphics hardware (GPU) has proven to be the most effective option for such computation intensive problems. The graphics hardware has, however, a massively parallel architecture very different from the traditional Neumann type processors, which has to be taken into account not only for algorithm design, but even on the level of mathematical modeling.

With a GPU implementation in mind, this dissertation proposes techniques to model and to simulate the main physical phenomena of PET from the emission of positrons until the absorption of γ-photons inside detectors as well as error reduction techniques to decrease integral estimation errors with negligible additional cost. We have to emphasize that the contributions of this dissertation fall into the category of new models and numerical methods to address the PET problem and not just their implementation in the massively parallel GPU architecture. However, as noted, the final implementation environment affects our development of numerical approaches even from the early design of models and solutions.

The proposed methods along with experimental studies justify that PET reconstruction using a physically accurate model can be evaluated in reasonable time on a single graphics hardware, and thus suitable for everyday clinical use.

1.1 Problem statement

The goal of *Positron Emission Tomography* (PET) reconstruction is to find out the *spatial density of the radioactive tracer* that was injected into the subject before the examination. The tracer typically consists materials essential for metabolism (e.g. oxygen or glucose) and thus, it is transported by the blood flow to regions with high cell activity enabling in vivo examination of organic functions. The numerous fields of application of PET including neurology, oncology, cardiology and pharmacology are out of scope of this dissertation, the interested reader is referred to [Che01, Kip02, KCC+09, MRPG12, DJS12].

As radioisotopes of PET undergo *positron emission decay* (a.k.a. β^+ decay), the tracer density is given by the number of emitted positrons. Formally, we are interested in the *3-*

dimensional density of positron emissions $x(\vec{v})$, in a finite function series form:

$$x(\vec{v}) = \sum_{V=1}^{N_{\text{voxel}}} x_V b_V(\vec{v}), \tag{1.1}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_{N_{\text{voxel}}})$ are unknown coefficients and $b_V(\vec{v})$ $(V = 1, \ldots, N_{\text{voxel}})$ are *basis functions* [CR12], which are typically defined on a *voxel grid*. As only non-negative tracer density makes sense, we impose positivity requirement $x(\vec{v}) \geq 0$ on the solution. If basis functions $b_V(\vec{v})$ are non-negative, the positivity requirement can be formulated for the coefficients as well: $x_V \geq 0$.

As in every type of tomography, the unknown function is reconstructed from its observed projections, which is the *inverse problem* of particle transport in scattering and absorbing media.

In the remainder of this section we discuss the complete sequence of the physical phenomena from the radioactive decay up to the detection of the corresponding particles inside the detectors (Figure 1.1), i.e. the *particle transport problem* for PET. We provide an introduction to PET scanners and the process through which the final input of the reconstruction algorithm is formed. Finally, we define the reconstruction problem.

1.1.1 PET physics: from decay to photon-hits

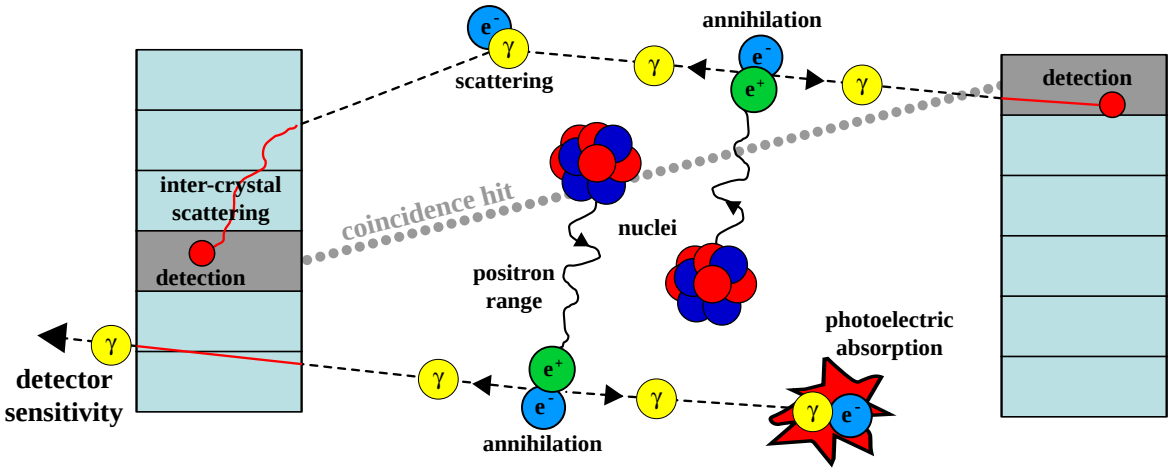


Figure 1.1: Physical process during a PET measurement starts with positron emission decay. Positrons travel through the tissue following a chaotic path, terminated by positron–electron annihilation. As a result of annihilation two anti-parallel γ-photons are born. These photons may interact with electrons of the surrounding medium, which results in either the photoelectric absorption or scattering of the photons. Inside the crystals, photons are detected by tracking their energy loss due to photoelectric absorption and Compton-scattering events, discarding those events that are outside a given energy range. However, photons may transfer their energy after arbitrary number of bounces, possibly occurring in several crystals away from their incident location known as inter-crystal scattering, or simply leave the system unnoticed, described by detector sensitivity. When two photons are detected in the given time window and energy range, the system registers a coincidence hit.

Positron range and annihilation

The physical process starts with *positron emission decay*. As a positron travels through the tissue it gives up its kinetic energy by Coulomb interactions with electrons, which results in a chaotic path terminated by *positron–electron annihilation*. The statistical properties of these

paths heavily depend on the initial kinetic energy, i.e. the type of the isotope and on the electron density of the particular tissue, i.e. the type of the material (e.g. bone, flesh, air, etc.). The *positron range*, i.e. the translation between the isotope decay and the positron annihilation results in positional inaccuracies in tomography reconstruction. As the mean free-path length of positrons is typically in a range of up to a few millimeters in tissues, positron range is one of the most important limiting factors of the resolution in small animal PETs [LH99, RZ07].

The spatial density on Cartesian axis X of the annihilation of a positron born in the origin can be approximated by [Der86, PB92, LH99]

$$p_X(X) = ae^{-\alpha X} + be^{-\beta X}. \tag{1.2}$$

Parameters a, α, b, β depend on the actual radiotracer and the material of the object, and can be determined by fitting this function onto data measured or simulated e.g. with GATE [Jea04].

As a result of positron–electron annihilation two anti-parallel γ-*photons* are born, each at an energy level of 511 keV since the energy must be preserved. Because of the conservation of momentum, the initial directions of the photons have an angular uncertainty of approximately 0.25 degrees FWHM [Bur09], known as *acollinearity*. We note that acollinearity is the only phenomenon completely neglected in our particle transport model which introduces a 2-3 mm and a 0.3-0.4 mm positional inaccuracy to human and small animal PET imaging, respectively. Ignoring acollinearity also has the important consequence that the photon-pair together initially travels a linear path.

Photon–matter interaction

A significant portion of the photons passes through host tissues directly without any interactions — hence the name *direct component* often used in the literature — and either leave the system or hit one of the surrounding detectors to finally get absorbed and thus detected (see the next section for more details). Simultaneous detection of photon pairs occurring within a few nanoseconds is registered as a valid *coincidence event* (a.k.a. *coincident hit* or *coincidence*) by the scanner. As the photon pair in this case follows a linear path, the number of coincidence events detected by a specific pair of detectors becomes proportional to the total number of positron-electron annihilations occurring in the pipe-like volume between the two detectors, called *volume of response* or *VOR* (see Figure 1.2). For infinitesimally small surfaces of detectors, this concept is reduced to a *line of response* (*LOR*), however, as a line uniquely joins two detectors the term LOR is also used to denote pairs of detector crystals, i.e. the conceptual detectors of PET.

Photons on the other hand, may interact with electrons of the surrounding medium, which results in either the *absorption* or *scattering* of the photons. The former case causes an *attenuation* of the detected beam of photons or reversely, when this attenuation is not considered it introduces a spatially varying underestimation of the reconstructed tracer density. During scattering, the photon looses a portion of its energy and more importantly, changes its direction, turning the path of the photon pair into a polyline consisting of arbitrary number of segments with arbitrary directions. This means that detected scattered photons, known as *scattered coincidence* (Figure 1.2), may originate from annihilations that occurred potentially anywhere in the measured object, even outside of the VOR subtended by the two detectors. Both the probability of the photon-electron interaction and the distribution of scattering direction depend on the material distribution, which varies from measurement to measurement, and the photon energy that is changing with scattering. Since the mean free-path length of γ-photons inside tissues is comparable to the diameter of the human chest (it is 10 cm in water for a 511 keV photon), accurate attenuation and scattering models become crucial especially for human PET: a roughly 30–50% of the photons get scattered before reaching the detectors [ZM07], depending on the scanner geometry and the size of the subject. For small animal PET systems the probability of photon-electron interactions is significantly smaller.

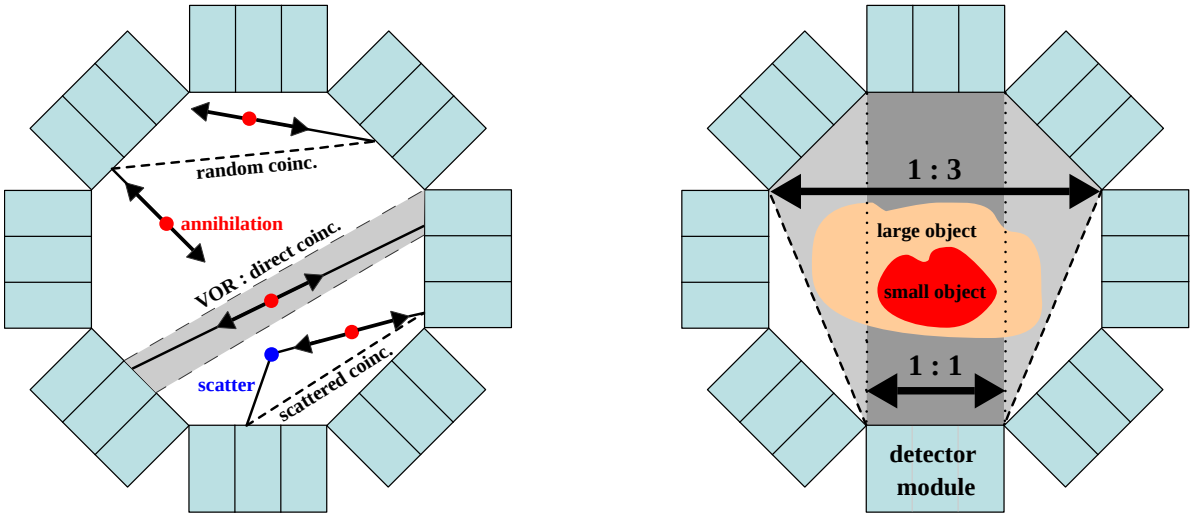


Figure 1.2: Coincidence types (left) and coincidence modes (right), depicted on an axial slice. Direct coincidences, i.e. when none of the photons scatters can only result from annihilations in the volume of response of the detector pair. When at least one of the photons changes its direction due to scattering, it may cause a scattered coincidence in potentially any of the detectors. The system may also register a coincident photon-hit originated from two different annihilation events, called random coincidence. Coincidence mode determines the field of view of the scanner: higher coincidence modes can measure larger objects at the expense of increased size of the produced data.

To describe photon–volume interaction, we consider how the photons go through participating media (Figure 1.3).

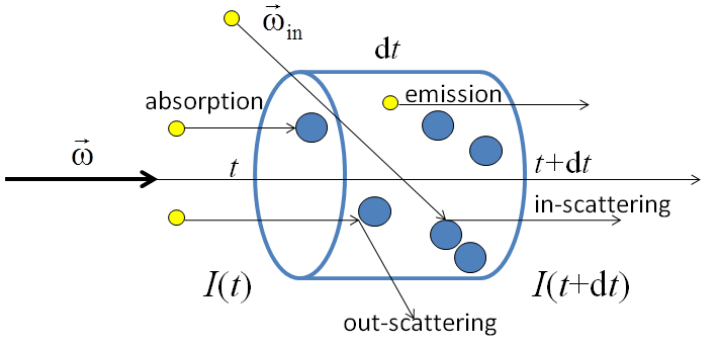


Figure 1.3: Modification of the intensity of a ray in participating media.

Let us consider the radiant intensity I on a linear path of equation $\vec{l}(t) = \vec{l}_0 + \vec{\omega}t$. The change of radiant intensity I on differential length dt and of direction $\vec{\omega}$ depends on different phenomena:

Absorption: the intensity is decreased when photons collide with the electrons or atomic cores and are absorbed due to the *photoelectric effect*. This effect is proportional to the number of photons entering the path, i.e. the intensity and the probability of this type of collision. If the probability of such collision in a unit distance is σ_a, called *absorption cross section*, then the probability of collision along distance dt is $\sigma_a(\vec{l})$dt. Thus, the total intensity change due to absorption is $-I(\vec{l})\sigma_a(\vec{l})dt$.

Out-scattering: the radiation is scattered out from its path when photons collide with the material and are reflected after collision. This effect is proportional to the number of photons entering the path, and the probability of such type of collisions in a unit distance, which is described by the *scattering cross section* σ_s. The total out-scattering term is $-I(\vec{l})\sigma_s(\vec{l})\mathrm{d}t$.

Emission: the intensity may be increased by the photons emitted by the medium. This increase in a unit distance is expressed by the emission density $I^e(\vec{l})$. We assume that the emission is isotropic, i.e. it is independent of the direction.

In-scattering: photons originally flying in a different direction may be scattered into the considered direction. The expected number of scattered photons from differential solid angle $\mathrm{d}\omega_{\mathrm{in}}$ equals to the product of the number of incoming photons and the probability that a photon is scattered in distance $\mathrm{d}t$, and the conditional probability density that the photon changes its direction from solid angle $\mathrm{d}\omega_{\mathrm{in}}$ to $\vec{\omega}$ provided that scattering happens. The conditional probability density is called the *phase function* $P(\vec{\omega}_{\mathrm{in}}, \vec{\omega})$, which depends on the angle θ between the incident and scattered directions:

$$\frac{\mathrm{d}\sigma_s}{\mathrm{d}\omega} = \sigma_s(\vec{x}) \cdot P(\vec{\omega}_{\mathrm{in}}, \vec{\omega}), \quad P(\vec{\omega}_{\mathrm{in}}, \vec{\omega}) = P(\vec{\omega}_{\mathrm{in}} \cdot \vec{\omega}) = P(\cos\theta).$$

Taking into account all incoming directions Ω of a sphere, the radiance increase due to in-scattering is:

$$\sigma_s(\vec{l})\mathrm{d}t \left(\int_\Omega I^{in}(\vec{l}, \vec{\omega}_{\mathrm{in}}) P(\vec{\omega}_{\mathrm{in}} \cdot \vec{\omega}) \mathrm{d}\omega_{\mathrm{in}} \right).$$

Cross sections of a material depend on frequency ν of the photons, or equivalently on their energy $E = h\nu$ where h is the Planck constant. In PET reconstruction, we are interested in the 100–600 keV range, where it is convenient to describe the frequency by the photon energy relative to the energy of the resting electron, i.e. by $\epsilon_0 = E/(m_e c^2)$ where m_e is the rest mass of the electron, c is the speed of light, and $m_e c^2 = 511$ keV is the energy of the resting electron.

The probability of the absorption due to the photoelectric effect depends on the material (grows rapidly with the atomic number) and is inversely proportional to the cube of the photon energy:

$$\sigma_a(\epsilon_0) \approx \frac{\sigma_a(1)}{\epsilon_0^3}.$$

If the photon energy does not change during collision, which happens when the photon collides with an atomic core or a base state, not excited electron, then the scattering is said to be *coherent* or *Rayleigh scattering* (RS). Coherent scattering can be described by the Rayleigh phase function

$$P_{\mathrm{RS}}(\cos\theta) = \frac{3}{16\pi}(1 + \cos^2\theta)$$

if the particle size is much smaller (at least 10 times smaller) than the wavelength of the radiation wave, which is the case of electrons and photons less than 1 MeV energy.

If energy is exchanged between the photon and the electron during scattering, the scattering is said to be *incoherent* or *Compton scattering* (CS). The energy change is defined by the Compton law:

$$\epsilon = \frac{1}{1 + \epsilon_0(1 - \cos\theta)},$$

where $\epsilon = E_1/E_0$ expresses the ratio of the scattered energy E_1 and the incident energy E_0, and $\epsilon_0 = E_0/(m_e c^2)$ is the incident photon energy relative to the energy of the electron. The

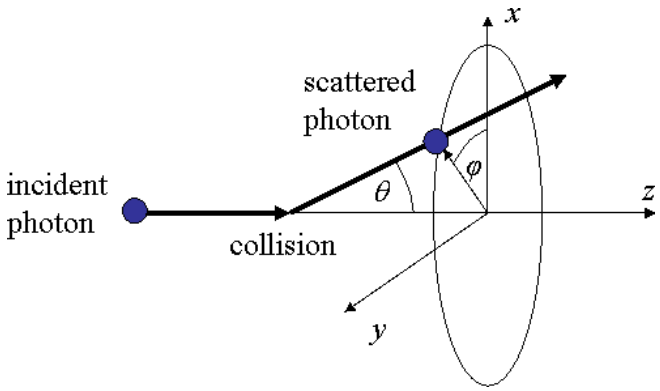


Figure 1.4: Geometry of photon scattering.

differential of the *scattering cross section*, i.e. the probability density that the photon is scattered from direction $\vec{\omega}$ to $\vec{\omega}_{\text{in}}$, is given by the *Klein-Nishina formula* [Yan08]:

$$\frac{\mathrm{d}\sigma_s^{\text{CS}}}{\mathrm{d}\omega} \propto \epsilon + \epsilon^3 - \epsilon^2 \sin^2\theta$$

where the proportionality ratio includes the classical electron radius and the electron density of the material. Instead of using these physical parameters explicitly, we may use the measured cross section of Compton scattering on energy level 511 keV, i.e. $\epsilon_0 = 1$ for the representation of the material. From this, the phase function that is supposed to be normalized can be found as:

$$P_{\text{KN}}(\cos\theta) = \frac{\epsilon + \epsilon^3 - \epsilon^2 \sin^2\theta}{\int_\Omega \epsilon + \epsilon^3 - \epsilon^2 \sin^2\theta \mathrm{d}\omega}.$$

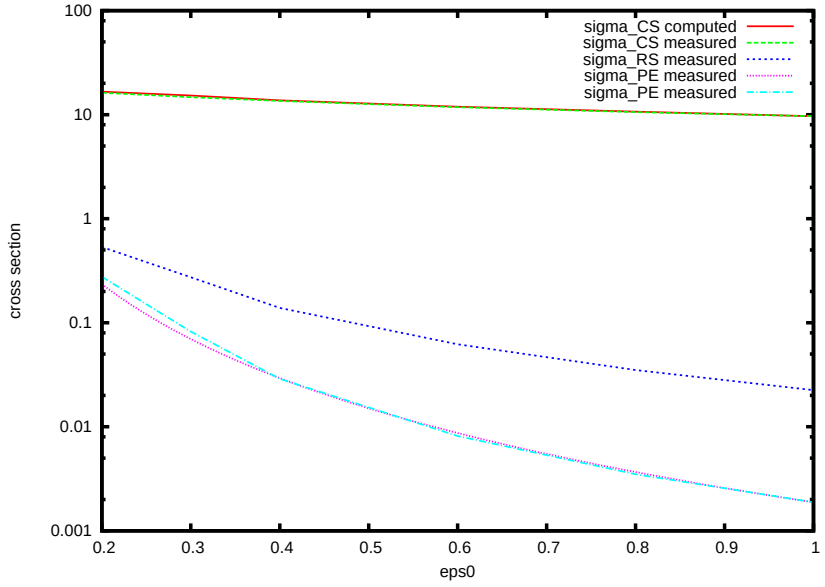


Figure 1.5: Cross sections σ_s^{CS}, σ_s^{RS}, σ_a [m^{-1}] for water in the 100 keV ($\epsilon_0 = 0.2$) and 511 keV ($\epsilon_0 = 1$) range. For Compton scattering and photoelectric absorption, we depicted both the calculated and the measured [BHS+98] energy dependence. Note that we used logarithmic scale as the absorption cross section and the Rayleigh cross section are almost two orders of magnitude smaller than the Compton cross section in this energy range.

The energy dependence of the Compton scattering cross section can be computed from the scaling factor in the Klein-Nishina formula:

$$\sigma_s^{\mathrm{CS}}(\epsilon_0) = \sigma_s^{\mathrm{CS}}(1) \cdot \frac{\int_\Omega \epsilon(\epsilon_0) + \epsilon^3(\epsilon_0) - \epsilon^2(\epsilon_0)\sin^2\theta \mathrm{d}\omega}{\int_\Omega \epsilon(1) + \epsilon^3(1) - \epsilon^2(1)\sin^2\theta \mathrm{d}\omega}.$$

The ratio between $\sigma_s^{\mathrm{CS}}(\epsilon_0)$ and $\sigma_s^{\mathrm{CS}}(1)$ is depicted as a function of relative energy ϵ_0 in Figure 1.5. These graphs apply to water, which is the most important constituent of living bodies, which are typically examined in PET.

Taking into account all contributions, intensity $I(\vec{l}, \vec{\omega}, \epsilon)$ of a particle flow at energy level ϵ satisfies an integro-differential equation:

$$\vec{\omega} \cdot \vec{\nabla} I(\vec{l}, \vec{\omega}, \epsilon) = \frac{\mathrm{d}I}{\mathrm{d}t} = -\sigma_t(\vec{l}, \epsilon) I(\vec{l}, \vec{\omega}, \epsilon) + I^e(\vec{l}, \epsilon) + \int_\Omega I(\vec{l}, \vec{\omega}_{\mathrm{in}}, \epsilon_{\mathrm{in}}) \frac{\mathrm{d}\sigma_s(\vec{l}, \vec{\omega}_{\mathrm{in}} \cdot \vec{\omega}, \epsilon_{\mathrm{in}})}{\mathrm{d}\omega_{\mathrm{in}}} \mathrm{d}\omega_{\mathrm{in}}, \quad (1.3)$$

where $\sigma_t(\vec{l}, \epsilon) = \sigma_a(\vec{l}, \epsilon) + \sigma_s(\vec{l}, \epsilon)$ is the *extinction parameter* that is the sum of the absorption cross section and the scattering cross section, $I^e(\vec{l}, \epsilon)$ is the source intensity, Ω is the directional sphere, ϵ_{in} and ϵ are the incident and scattered photon energies, respectively. Scattered photon energy ϵ is equal to incident photon energy ϵ_{in} for coherent scattering. For incoherent scattering, the scattered and incident photon energies are related via scattering angle $\cos\theta = \vec{\omega} \cdot \vec{\omega}_{\mathrm{in}}$ as stated by the Compton law.

In PET [RZ07], source intensity is non zero only at $\epsilon = 511$ keV. Photon energy may drop due to incoherent scattering. As typical detectors are sensitive in the 100–600 keV range, we can ignore photons outside this range. In this energy range and typical materials like water, bone and air, incoherent scattering is far more likely than coherent scattering, thus we can ignore Rayleigh scattering. However, we note that the inclusion of Rayleigh scattering into the model would be straightforward.

According to Equation 1.3, the intensity along a ray is decreased due to absorption and out-scattering. However, photons scattered out show up as a positive contribution in the in-scattering term in other directions, where they represent a positive contribution. While absorption decreases the intensity along the ray and also the radiation energy globally, out-scattering is a local loss for this ray, but also a positive contribution for other directions, so globally, the number of relevant photons is preserved while their energies may decrease due to the Compton effect.

If the in-scattering integral is ignored, Equation 1.3 becomes a pure linear differential equation

$$\frac{\mathrm{d}I}{\mathrm{d}t} = -\sigma_t(\vec{l}, \epsilon) I(\vec{l}, \vec{\omega}, \epsilon) + I^e(\vec{l}, \epsilon), \quad (1.4)$$

which can be solved analytically resulting in

$$I(\vec{l}(t), \vec{\omega}, \epsilon) = A_\epsilon(t_0, t) I(\vec{l}(t_0), \vec{\omega}, \epsilon) + \int_{t_0}^{t} A_\epsilon(\tau, t) I^e(\vec{l}(\tau), \epsilon) \mathrm{d}\tau, \quad (1.5)$$

where

$$A_\epsilon(\tau, t) = e^{-\int_\tau^t \sigma_t(\vec{l}(u), \epsilon) \mathrm{d}u} \quad (1.6)$$

is the *attenuation* for photons of energy ϵ between points $\vec{l}(\tau)$ and $\vec{l}(t)$. When the line is explicitly given by its endpoints $\vec{v}_1$, $\vec{v}_2$, we use the notation $A_\epsilon(\vec{v}_1, \vec{v}_2)$. Having extinction parameter $\sigma_t(\vec{l}, \epsilon) = \sigma_a(\vec{l}, \epsilon) + \sigma_s(\vec{l}, \epsilon)$, $A_\epsilon(\tau, t)$ is the product of the *attenuation due to out-scattering* $T_\epsilon(\tau, t)$, and the *attenuation due to photoelectric absorption* $B_\epsilon(\tau, t)$:

$$T_\epsilon(\tau, t) = e^{-\int_\tau^t \sigma_s(\vec{l}(u), \epsilon) \mathrm{d}u}, \qquad B_\epsilon(\tau, t) = e^{-\int_\tau^t \sigma_a(\vec{l}(u), \epsilon) \mathrm{d}u}.$$

When only unscattered contribution, i.e. 511 keV photons are considered, we shall omit photon energy from the notation.

Photon detection

The majority of PET scanners use scintillation detector systems (Figure 1.6). Scintillator crystals convert the energy of photoelectric absorption and Compton-scattering events of γ-photons to light photons, from which a *photomultiplier tube* (PMT) generates electric signals. The analysis of these signals tells us information about the time and location of the interaction events between the incident γ-photon and the electrons of the crystal. However, the mean free-path length of γ-photons inside the crystals may be much larger than the crystal size (especially for small animal PET, where this factor might be 5 or even higher), which means that photons may transfer their energy after arbitrary number of bounces, possibly occurring in several crystals away from their incident location known as *inter-crystal scattering* (Figure 1.6), or simply leave the system unnoticed. Additionally, due to manufacturing errors, the sensitivity of detection varies from crystal to crystal (often referred to as *crystal efficiency*). These effects altogether form the so-called *detector model* of PET that can be described by *transport function* $E_t(\vec{z}, \vec{\omega}, \epsilon_0 \to \mathbf{d})$ that gives the expected number of hits reported in crystal $\mathbf{d}$ provided that a photon entered the detector module at point $\vec{z}$ from direction $\vec{\omega}$ and with energy ϵ_0. We should use conditional expected value instead of conditional probability since the measuring system consisting of photon multipliers and electronics can result in values larger than 1 as a response to a single photon incident. The transport function E_t may be obtained through empirical measurements [SPC+00, TQ09, AST+10], Monte Carlo simulations [MLCH96, MDB+08, LCLC10, C8] or approximated analytically [YHO+05, MDB+08, RLT+08, C8].

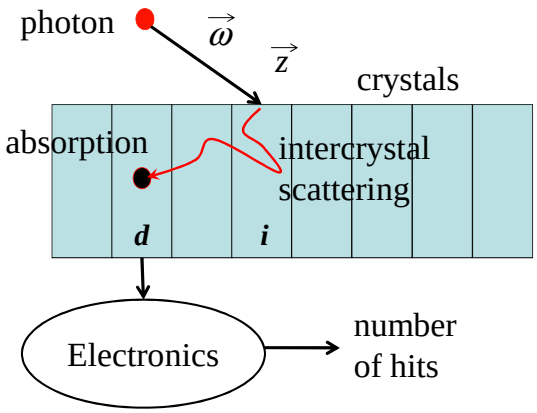


Figure 1.6: Inter-crystal scattering

1.1.2 The scanner system

Scanner geometry and acquisition mode

The measured object is surrounded by detectors (see Figure 1.2), which absorb the γ-photons emitted during the examination. There is a large number of PET scanners available in the market with different geometric properties; in the devices on which the results of this dissertation were tested detector crystals are packed together into panels forming 2D grids, called *detector modules*. For a given pair of modules, therefore, the set of LORs is 4D data. Modules are placed in a cylindrical shape around the object, the axial direction of the cylinder is consequently named as the *axial* direction of the system, denoted by the $\mathbf{z}$ axis. The perpendicular direction is called *transaxial* and referred to as the $\mathbf{x}$-$\mathbf{y}$ plane.

Early PET scanners used septa between the axial slices of the detectors to limit incoming photon directions to nearly perpendicular to the $\mathbf{z}$ axis resulting in a so-called *2D imaging*,

considering only LORs approximately lying within the axial planes. This could greatly decrease the generated data allowing faster reconstruction [AK06]. Furthermore, as staying in or getting within axial slices after scattering happens with a very low probability, scattered events are almost completely eliminated making scatter correction unnecessary [AK06]. PET imaging, however, has always been struggling with low signal-to-noise ratio which is even further reduced by considering only a small portion of the data in the 2D case. As a consequence, *fully 3D imaging* is used nowadays without restricting the axial direction of the LORs. A typical fully 3D acquisition system consists of hundreds of millions of LORs, each of which may capture a significant portion of scattered events [Tho88]. For fully 3D imaging, thus, algorithms must be implemented on high performance hardware and include an accurate scatter model.

Coincidence types and modes

When two incident photons are detected in a given time window and energy range, the device registers it as a coincidence for the corresponding LOR. However, it is possible that two crystals detect a photon hit each within the detection time window, but the two photons were not born from the same annihilation, which is called a *random event* or *random coincidence* (Figure 1.2). Such events are generated by photons whose pair was lost due to attenuation of the object, limited field of view (FOV), miss in the detector, etc. Random events can also be caused by the self emission of crystals. Finally, random events can be the consequence of the bad pairing of photon pairs. For example, if a coincidence pair and a random photon or another coincidence pair cause events within the detection time window, the electronics may identify more than one coincidence pair. Random events are omitted throughout the dissertation; nevertheless, it is worth mentioning that there exist methods either to apply random correction on the input of the reconstruction algorithm [HHPK81], or to include it into the factorized model [BKL$^+$05, DJS12] as shown in Section 1.2.5.

As a large portion of the detected coincidences are unscattered events and even scattering is more likely to happen forward [ZM07, C9], it is worth restricting the accepted coincident events to those that have their endpoints at the opposite sides of the scanner, to reduce the number of random events and the size of the captured data. This restriction is defined in terms of detector modules, $1 : N$ *coincidence mode* means that a module can form LORs with the N opposite modules, coincident photon hits arriving outside of this range are rejected by the system.

Detector dead-time

During data acquisition, a number of coincident hits may be missed when the system is "busy" either with processing a previous event or due to transferring buffered data, which is known as *dead-time*. Dead-time is not considered in this thesis work. However, we note that the percentage of loss by dead-time can be characterized with a single constant for each pair of detector modules which may be included in the reconstruction with relative ease [CGN96, BM99].

List-mode and binning

If a *list-mode* reconstruction algorithm is used, each detected LOR-hit, optionally extended with the estimated *Time of Flight* (*ToF*) of the two photons, is immediately sent to the reconstruction process. This enables reconstruction during the data acquisition using only a small but continuously growing amount of information. List-mode reconstructions had their significance when the time required for performing the reconstruction was large, compared to that of the data acquisition. With the evolution of hardware and algorithm, reconstruction has become faster causing list-mode to be used less frequently. *Binned reconstructions* first create the histogram of coincidences $\mathbf{y} = (y_1, y_2, \ldots, y_{N_{LOR}})$ in a pre-processing step, with y_L denoting the number of coincidences detected in LOR L during the acquisition, and thus utilize all the available information from the beginning of their execution. Furthermore, the histogram can be spatially

ordered which allows more efficient access than list-mode data. Recent results show that ToF information may be incorporated to binned reconstruction in an efficient way [SSKEP13].

Multi-modal imaging

Modern PET scanners are coupled with other modalities such as Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) and thus are capable of producing different types of data simultaneously. This is especially important for PET since positron range, scattering and absorption depend on the material properties of the measured object. Thus, we assume that a *registered and segmented material map* is available for the reconstruction algorithm providing a unique material index $m(\vec{v})$ in every point $\vec{v}$ of the volume of interest, along with the required data such as the absorption cross section σ_a or the scattering cross section σ_s for each material type. Although segmentation may introduce error to the transmission data, in fact, it was shown by Ollinger [Oll97] that adaptive segmentation greatly increases the accuracy of analytic scatter correction methods (e.g. the single scatter model presented in Chapter 5) for short CT scans.

Real scanners

In this dissertation we assumed two real scanners, a preclinical one and a human one.

Preclinical PET scanner: Mediso's nanoScan-PET/CT

Small animal PET tests were carried out with Mediso's *nanoScan-PET/CT* [Med10b], which has 12 detector modules consisting of 81×39 crystals of size $1.12^2 \times 13$ mm^3. It supports 1:3 and 1:5 coincidence modes, the number of LORs is $N_{\text{LOR}} \approx 1.8 \cdot 10^8$ and $N_{\text{LOR}} \approx 3 \cdot 10^8$, respectively.

Human PET scanner: Mediso's AnyScan human PET/CT

For human PET tests, we used Mediso's *AnyScan human PET/CT* [Med10a]. AnyScan has 24 detector modules consisting of 27×38 crystals of $3.9^2 \times 20$ mm^3. In 1:13 coincidence mode, the number of LORs is $N_{\text{LOR}} = 1.6 \cdot 10^8$.

1.1.3 The PET reconstruction problem

The objective of PET reconstruction is to determine the voxel intensities $\mathbf{x} = (x_1, x_2, \ldots, x_{N_{\text{voxel}}})$ from the set of observations $\mathbf{y} = (y_1, y_2, \ldots, y_{N_{LOR}})$, the measured hits in detector pairs. The correspondence between the positron density $x(\vec{v})$ and the expected number of hits $\tilde{y}_L$ in LOR L is described by *scanner sensitivity* $\mathcal{T}(\vec{v} \to L)$ that expresses the probability of generating an event in the two detectors of LOR L given that a positron is emitted in point $\vec{v}$:

$$\tilde{y}_L = \int_{\mathcal{V}} x(\vec{v}) \mathcal{T}(\vec{v} \to L) \mathrm{d}v \tag{1.7}$$

where $\mathcal{V}$ is the volume where the positron density is to be reconstructed. This scanner sensitivity is usually a high-dimensional integral of variables unambiguously defining the path of particles from positron emission point $\vec{v}$ to the detector electronics. Considering the finite series form approximation of $x(\vec{v})$ (Equation 1.1) we obtain:

$$\tilde{y}_L = \int_{\mathcal{V}} x(\vec{v}) \mathcal{T}(\vec{v} \to L) \mathrm{d}v = \sum_{V=1}^{N_{voxel}} A_{LV} x_V \tag{1.8}$$

where

$$A_{LV} = \int_{\mathcal{V}} b_V(\vec{v}) \mathcal{T}(\vec{v} \to L) \mathrm{d}v \tag{1.9}$$

is the *System Matrix* (*SM*) [JSC+97]. This equation can also be written in a matrix form:

$$\tilde{\mathbf{y}} = \mathbf{A} \cdot \mathbf{x}.$$

Data values of the PET measurement are intrinsically stochastic due to the underlying physics and detection process such as the positron decay or scattering, thus the *model of PET reconstruction* includes a statistical noise component $\mathbf{n}$:

$$\mathbf{y} = \mathbf{A} \cdot \mathbf{x} + \mathbf{n} \tag{1.10}$$

1.2 Reconstruction framework

PET reconstruction methods consist of three basic components [AK06], surveyed in this section. Subsection 1.2.1 discusses different approaches for modeling the positron density function $x(\vec{v})$. A brief overview of the algorithms solving Equation 1.10, i.e. finding the positron density for a given measurement is presented in Subsection 1.2.2, whereas Subsection 1.2.3 describes the specific algorithm used in the dissertation. Subsection 1.2.4 collects techniques for estimating the SM. Subsection 1.2.5 discusses the basic idea of factoring the SM into phases and finally, Subsection 1.2.6 presents the decomposition according to physical phenomena, that is widely used in the literature and also adopted by our work.

1.2.1 Models of the unknown function

Due to its efficiency and simplicity, the most popular choice of basis functions in Equation 1.1 — also favored by the dissertation — are piece-wise constant and tri-linear approximations defined on a regular 3D grid. Tri-linear interpolation is especially preferred in GPU-based applications since the hardware provides it with no additional cost. Regular 3D grids are widely used in many fields that have to deal with 3D data, such as virtual colonoscopy [KJY03, CK04, KSKTJ06, Kol08], flow simulation [Klá08], volume visualization [J4] or volumetric effects for computer games [TU09]. Other popular grid structures are the BCC [Csé05, Csé10] and FCC grids, which would be worth considering in tomography. Smooth basis functions, such as blobs [Lew92, ML96, CGR12] or clusters of voxels [RSC+06] have been proposed to include prior information on image smoothness to the model, but the complexity of these approaches is prohibitive for clinical use [AK06]. Treating the coefficients of the discretized positron density function as random variables leads to Bayesian reconstruction algorithms [Gre90b, GLRZ93, RLCC98], however, these are less often used due to their sensitivity to parameter settings and increased complexity in implementation [AK06].

1.2.2 Reconstruction algorithms

The standard reconstruction algorithm of PET is the *Filtered Back Projection* (*FBP*) [BR67, Lak75], which is based on direct inversion of the Radon Transform [Rad17, Rad86] and belongs to the family of analytic algorithms. In FBP it is assumed that observed LOR values y_L are (1) noise-free and (2) can be expressed as line integrals through the measured object i.e. contain only the direct component; with these assumptions the reconstruction can be solved analytically. *Iterative algebraic methods* [VDdW+01], such as the Algebraic Reconstruction Technique (ART) [GBH70, Gor74], the Simultaneous Iterative Reconstruction Technique (SIRT) [Gil72] or the Iterative Least-Squares Technique (ILST) [Goi72], try to solve Equation 1.10 directly with assuming no noise i.e. $\mathbf{n} = 0$, by minimizing the L_2 norm of the left and right sides, i.e. $||\mathbf{y} - \mathbf{A} \cdot \mathbf{x}||_2$. These approaches have no restriction on the SM which means that the entire physical model can be incorporated, often resulting in a much higher image quality. The iterative *Maximum Likelihood Expectation Maximization* (*ML-EM*) method by Shepp and Vardi [SV82] goes one step further, and incorporates the Poisson nature of the acquired data into the model.

1.2.3 Maximum likelihood expectation maximization

The goal of the ML-EM algorithm is to find the discretized tracer density $\mathbf{x}$ which has the highest probability to have generated the measured projection data $\mathbf{y}$ [VDdW$^+$01]. Assuming that photon incidents in different LORs are independent random variables with Poisson distribution, the algorithm should maximize the following likelihood function under the positivity constraint of the solution:

$$\log \mathcal{L} = \sum_{L=1}^{N_{LOR}} \left(y_L \log \tilde{y}_L - \tilde{y}_L \right). \tag{1.11}$$

The iterative optimization [QL06] alternates *forward projection*

$$\tilde{y}_L = \sum_{V=1}^{N_{voxel}} \mathbf{A}_{LV} x_V^{(n)}, \tag{1.12}$$

then *back projection*:

$$x_V^{(n+1)} = \frac{x_V^{(n)}}{\sum_{L=1}^{N_{LOR}} \mathbf{A}_{LV}} \cdot \sum_{L=1}^{N_{LOR}} \mathbf{A}_{LV} \frac{y_L}{\tilde{y}_L} \tag{1.13}$$

in each of the $n = 1, 2, \ldots$ iteration step. This can be written in matrix form, where vector division is defined in an element-wise basis:

$$\text{Forward: } \tilde{\mathbf{y}} = \mathbf{A} \cdot \mathbf{x}^{(n)}, \qquad \text{Back: } \frac{\mathbf{x}^{(n+1)}}{\mathbf{x}^{(n)}} = \frac{\mathbf{A}^T \cdot \frac{\mathbf{y}}{\tilde{\mathbf{y}}}}{\mathbf{A}^T \cdot \mathbf{1}} \tag{1.14}$$

There are two main drawbacks of the ML-EM algorithm [VDdW$^+$01]. First, the algorithm is known to be ill-conditioned, which means that enforcing the maximization of the likelihood function may result in a solution with drastic oscillations and noisy behavior. This problem can be attacked in several different ways, such as the inclusion of additional information, e.g. as a penalty term, leading to *regularization methods* [Gre90a]. An appropriate penalty term is the *total variation* of the solution [PBE01, B3, D6] which forces the reconstructed function to contain less oscillations while preserving sharp features. Other approaches can be, for example, the inclusion of stopping rules into the algorithm to terminate the iteration process before noise deterioration could degrade image quality [VL87, BMM08, Gai10] or simply to post-filter the output of the ML-EM algorithm [DJS12].

The second disadvantage of the ML-EM algorithm is its computational complexity, especially for the fully 3D case, as the iteration should work with very large matrices. Additionally, as this section will show later, it is beneficial to re-compute matrix elements as high-dimensional integrals in every iteration step. This needs enormous computation power if we wish to obtain the reconstruction results in reasonable time (i.e. at most in a few minutes).

Ordered subset expectation maximization

A slight modification to the ML-EM algorithm was introduced by Hudson and Larkin to reduce its computation time. *Ordered Subsets Expectation Maximization* (*OSEM*) [HL94] updates the current estimate of the tracer density $x^{(n)}$ using only a subset S_b ($b = 1 \ldots B$) of the entire data [AK06]:

$$x_V^{(n+1)} = \frac{x_V^{(n)}}{\sum_{L \in S_b} \mathbf{A}_{LV}} \cdot \sum_{L \in S_b} \mathbf{A}_{LV} \frac{y_L}{\tilde{y}_L}.$$

Subsets cover the complete set of LORs and are alternated in each iteration periodically, therefore compared to ML-EM, the tracer density is updated with the entire measurement data over B

subiterations. For $B = 1$, OSEM is equivalent to the conventional ML-EM. Several strategies exist for grouping the data into subsets, however, theoretical comparison is yet to be given. In practical terms, it has been demonstrated that increasing the number of subsets can increase convergence speed at the expense of greater image noise [LT00]. As a common experience, N OSEM iterations using B subsets deliver the same level of convergence (in terms of ML) as $N \times B$ ML-EM iterations, practically meaning a B-times faster execution [HTC+05, Ser06, AK06].

1.2.4 System matrix estimations

The foundations of ML-EM have been well established for three decades, the basic equations of this iterative scheme are fairly simple and their sequential implementation is straightforward. The crucial element of PET reconstruction in these days is the accurate estimation of the SM, which has huge effect on image quality. Several approaches try to obtain and store the SM either by measuring [LKF+03, PKMC06] or pre-computing it [RMD+04, HEV+06, YYO+08, MDB+08, SSG12]. There are three major problems with these types of methods. First, the use of pre-computed or measured data prohibits the consideration of patient or object specific positron range, absorption or scattering. Second, in high-performance computing, especially for GPUs, most of the input data has to fit into the main memory of the target hardware in order to avoid a huge decrease in performance (see Section 1.3). However, the SM is huge, its size is typically in the order of magnitude of $10^8 \times 10^7$ (e.g. assuming $N_{\text{voxel}} = 256^3 \approx 1.6 \cdot 10^7$ voxels and the data dimensions N_{LOR} of the real scanners of Section 1.1.2). For high resolution scanners, thus, storing the SM is mostly hopeless even if it is factored [QLC+98] (Section 1.2.5) and its symmetry is exploited [MR06, HCK+07, HEG+09]. And finally, the stored SM never equals to the true value. Using a fixed estimation in every iteration introduces the same error and thus modifies the fixed point. As demonstrated in Chapter 2, re-computing the matrix on-the-fly is usually a better choice.

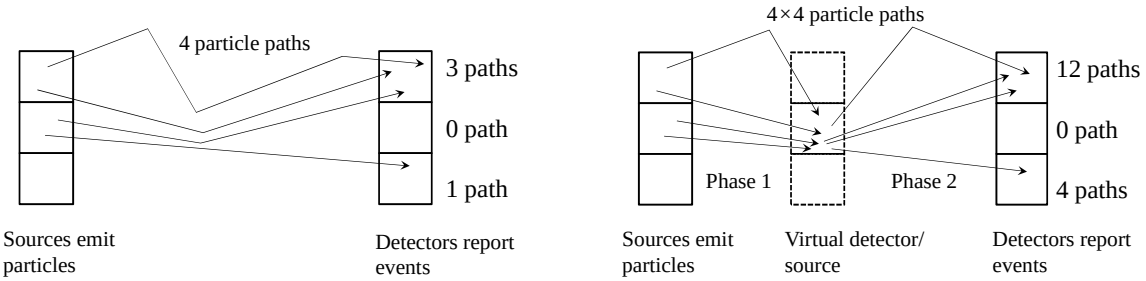
Emitted particles may end up in detectors after traveling in space including possible scattering and type changes. As the source and the detectors have 3D domain, and scattering can happen anywhere in the 3D space, the contribution of sources to detectors is a high-dimensional integral in the domain of source points, detector points and arbitrary number of scattering points. Such high-dimensional integrals are calculated by numerical integration where sampling corresponds to tracing paths. The more paths are computed, the higher precision reconstruction is obtained. Classical quadratures, such as the rectangle rule, suffer from the so-called "*curse of dimensionality*": the number of samples required to achieve a certain level of approximation grows exponentially with the dimension. The traditional method to handle high-dimensional integrals is the Monte Carlo integration, described in Section 2.1.

1.2.5 Factorized model

Factoring [QLC+98] the SM according to physical phenomena can help not only to reduce data storage but also to speed up calculations if matrix elements are computed on-the-fly. The idea of factoring is that the transport process is decomposed to phases with the introduction of virtual detectors (Figure 1.7 shows the decomposition into two phases, but more than two phases are also possible). First the expected values in the first layer of virtual detectors are computed from the source. Then, the first layer of these detectors become sources and a similar algorithm is executed until we arrive in the real detectors. The advantages of this approach are the following:

- The calculation of a single phase can be much simpler than the complete transport process, thus we can eliminate all conditional statements that would reduce GPU efficiency.
- As a computed sample path ended in a virtual detector is continued by all paths started here in the next phase, we shall have much higher number of sample paths to compute high dimensional integrals, thus the result is more accurate (see Figure 1.7 for an example).

- Each phase is computed in parallel on the GPU where threads do not communicate. However, the next phase can reuse the results of all threads of the previous phase, so redundant computations can be eliminated.
- The reconstruction algorithm becomes modular. Models of the physical phenomena can be integrated or improved seamlessly, with minimal modification of the rest of the code.



(a) Traditional model (b) Factored model

Figure 1.7: Subfigure (a) shows the traditional model of emission tomography where the high-dimensional integrals describing the particle transport are calculated by tracing sample paths. The more paths are computed, the higher precision reconstruction is obtained. Subfigure (b) depicts the conceptual model of factoring: the particle transport process is decomposed to phases by introducing virtual detectors. The simulation of all particles is first executed to the virtual detectors, then virtual detectors become virtual sources and the second phase simulates transport from here to the real detectors. We also indicated the number of computed sample paths associated with each detector. Note that the number of sample paths increased from 4 to 16.

The disadvantage of factoring is that virtual detectors discretize the continuous space into finite number of bins, so if their number is small, discretization error occurs.

1.2.6 Decomposing the system matrix

The expectations of LOR values $\tilde{y}_L$ can be expressed as a sum of three terms [RZ07], the direct contribution, the scattered contribution, and the random contribution:

$$\tilde{\mathbf{y}} = \tilde{\mathbf{y}}^{\text{direct}} + \tilde{\mathbf{y}}^{\text{scatter}} + \tilde{\mathbf{y}}^{\text{random}}.$$

Hereafter, we ignore the random contribution from the model. We note that it is an additive term in the SM and thus can be included independently of the methods presented in the dissertation. The direct and scattered contributions are calculated by approximating the corresponding SM elements:

$$\tilde{\mathbf{y}}^{\text{direct}} = \mathbf{A}^{\text{direct}} \cdot \mathbf{x}, \quad \tilde{\mathbf{y}}^{\text{scatter}} = \mathbf{A}^{\text{scatter}} \cdot \mathbf{x}.$$

The SMs describing the direct and scattered contributions are factored, i.e. they are approximately expressed as products of smaller matrices according to the main physical effects:

$$\mathbf{A} \approx \mathbf{A}^{\text{direct}} + \mathbf{A}^{\text{scatter}}, \quad \mathbf{A}^{\text{direct}} = \mathbf{L} \cdot \mathbf{D} \cdot \mathbf{P}, \quad \mathbf{A}^{\text{scatter}} = \mathbf{L} \cdot \mathbf{S} \cdot \mathbf{P}, \quad \mathbf{D} = \hat{\mathbf{T}} \cdot \mathbf{G}, \tag{1.15}$$

where $\mathbf{P}$ is the voxel space blurring matrix of *positron range*, $\mathbf{G}$ is the non-square matrix of *geometric projection*, $\mathbf{S}$ is the non-square matrix of *scattered projection* considering also attenuation, $\hat{\mathbf{T}}$ is the diagonal matrix of *phantom attenuation* factors, $\mathbf{D}$ is the direct contribution including attenuation and $\mathbf{L}$ is the LOR space blurring matrix representing the *detector model*. Based on this factorization, we use the following notations in later chapters:

$$\mathbf{x}^{\text{a}} = \mathbf{P} \cdot \mathbf{x}, \quad \tilde{\mathbf{y}}^{\text{geom}} = \mathbf{D} \cdot \mathbf{x}^{\text{a}}, \quad \tilde{\mathbf{y}}^{\text{detmod}} = \mathbf{L} \cdot \tilde{\mathbf{y}}^{\text{geom}}.$$

where $\mathbf{x}^{\mathrm{a}}$ is the annihilation density, $\tilde{\mathbf{y}}^{\mathrm{geom}}$ is the expected number of direct hits on the surfaces of the detectors, and $\tilde{\mathbf{y}}^{\mathrm{detmod}}$ is the expected number of detected direct hits.

In theory, the exact back projector would be the transpose of the SM. However, as the back projector is computationally more expensive, in most cases a simplified model is used in this phase. Section 2.2 shows that it is beneficial to exclude voxel-space blurring effects from the back projection assuming a high dose measurement.

1.3 Key aspects of efficient GPU programming

The *Graphics Processing Unit* (*GPU*) or *graphics card* was originally designed to accelerate the highly parallel and arithmetic-intensive computations of computer graphics, such as vertex transformation or pixel shading. Due to their massively parallel architecture, GPUs have become much more efficient than traditional CPUs in terms of arithmetic throughput and bandwidth. Thus, in the past decade, a high effort was spent to utilize the computational power of graphics processors for general purpose tasks [DJ09] other than the conventional graphics pipeline, such as procedural geometry [C2, B2], ray-tracing [D3], global illumination [SKSS08], non-photorealistic visualization [J3, C15], CT reconstruction [JDB08, JRM$^+$09, JSK13] or augmented reality [J7]. Among the high-performance computing possibilities, like FPGAs [LCM$^+$02, ZSD$^+$08], multi-CPU systems [SRA$^+$02], the CELL processor [KKB07], and GPUs [XM07], the massively parallel GPU has proven to be the most cost-effective platform for tomography reconstruction [GMDH08]. The critical issue of GPU programming, and parallel programming in general, is *thread mapping*, i.e. the decomposition of the algorithm to parallel threads that can run efficiently. For example, while simulating particle transport, it is intuitive to mimic how nature works in parallel, and assign parallel computational threads, for example, to randomly generated photon paths [WCK$^+$09]. However, while significant speedups can be obtained with respect to a CPU implementation, this "natural" thread mapping cannot exploit the full potential of GPUs. Efficient GPU code requires the consideration of the GPU features even from the very first steps of problem definition and algorithm development. More specifically, the following issues must be considered:

- *Thread divergence*: A GPU is a collection of multiprocessors, where each multiprocessor has several *Single Instruction Multiple Data* (*SIMD*) scalar processors that share the instruction unit and thus always execute the same machine instruction. Thus, during algorithm development we should minimize the dependence of flow control on input data.

- *Coherent memory access and non-colliding writes*: Generally, memory access is slow on the GPU compared to the register and local memory access and to the computational performance of processors (e.g. on NVIDIA GPUs the difference is two orders of magnitude [NVI13]), especially when atomic writes are needed to resolve thread collisions. In addition to avoiding atomic writes, a significant speed up can be achieved by so-called *coalesced memory accesses*. If threads of the same scalar processor access neighboring data elements, then the transfer is executed in a single step amortizing the access time. This means we should design neighboring threads to access neighboring data elements. In iterative EM reconstruction, forward projection computing the expected detector hits from the actual positron density estimate and back projection correcting the positron density based on the measured and expected detector responses alternate. Equations of forward projection and back projection are similar in the way that they take many input values (voxel intensities and LORs, respectively) and compute many output values (again, LORs and voxel intensities, respectively). This kind of "many to many" computation can be organized in two different ways. We can take input values one-by-one, obtain the contribution of a single input value to all of the outputs, and accumulate the contributions as different input values are visited. We call this scheme *input driven* or *scattering*. The orthogonal approach would take output values (i.e. equations) one-by-one, and obtain the

contribution of all input values to this particular output value. This approach is called *output driven* or *gathering*. Generally, if possible, gathering type algorithms must be preferred since they can completely remove write collisions and may increase the coherence of memory access [PX11]. The forward projection of the ML-EM computes LOR values from voxels, whereas the back projection maps these LORs back to voxels. An efficient, gather-style GPU implementation of the forward and back projectors must be LOR centric and voxel centric, respectively [PX11], i.e. the forward projector should read the contributing voxel values and likewise, the back projector should accumulate the correction of each LOR.

- *Reuse*: Running independent threads on different processors, we cannot reuse temporary results obtained by different processors, which is obviously possible and worthwhile in a single thread implementation. To reuse partial results of other threads, the algorithm should be broken to phases, e.g. to voxel-processing, voxel-to-LOR transformation, and LOR-processing. When threads implementing a phase are terminated, they write out the results to the global memory. The threads of the next phase can input these data in parallel without explicit communication.

- *On-the-fly computation*: Current GPUs offer over one teraflops computational performance but their storage capacity is small and the CPU to GPU communication is relatively slow. Thus, GPU implementations have a different trade off between pre-computation with storage and on-the-fly re-computation whenever a data element is needed. For example, in GPU based PET reconstruction the SM cannot be stored but elements should be re-computed each time when they are needed.

1.4 Verification and validation methodology

Clinical acceptance of reconstruction methods assumes that a sufficient image quality is provided, i.e. in the case of this dissertation the physical effects are accurately modeled, in reasonable time (typically a few minutes). The National Electrical Manufacturers Association (NEMA) defines performance evaluation protocols in terms of image quality for both pre-clinical [Ass08] and human [Ass07] PET scanners, which have become a standard in the past few years. However, as it was pointed out recently by Goertzen et al. [GBB+12], the NEMA protocol prescribes a FBP algorithm to reconstruct the image and thus it is not designed to evaluate reconstruction methods. To the best knowledge of the author, no such standard protocol exists. Thus, we have developed our own methodology to evaluate the proposed reconstruction methods.

1.4.1 Scenarios

We use both real and hypothetical scanners. As the reconstruction scheme we used the ML-EM iteration algorithm, test cases differed in the dimensions of the scanner and the scanner sensitivity. We defined the following scenarios:

1. In the *analytical case* both the ground truth SM and the source are assumed to be explicitly known (which is never the case in practice), allowing us to compare the convergence of different sampling strategies in the ML-EM scheme with the theoretically best case, i.e. when iterating with the real SM.

2. Next, we leave the assumption of the explicitly known SM. The "measured" data $\mathbf{y}$ is computed with *off-line Monte Carlo simulation* using *GATE* [Jea04], taking into account the physical effects selectively (as few as possible at a time), which is ideal for evaluating the model of these phenomena independently. Tracer density — being the input of the off-line

simulation — is still known providing us information about convergence in terms of distance metrics between volumetric data. Scanner geometry and detector crystal properties accurately modeled the real scanners presented in Section 1.1.2.

3. The methods were also applied to *real scanner* data, including prepared physical phantoms like the Derenzo phantom or point phantoms, for which the activity distribution is approximately known. For preclinical PET, positron range and inter-crystal scattering are the dominant image degrading effects making it an ideal platform for testing these phenomena. Contrarily, photon absorption and scattering in the measured object play a major role in human PET and the effect of positron range and the detector model are almost negligible.

In the following we describe the system parameters of the different scenarios.

Analytic SM

1D case

SM of dimensions $N_{LOR} = 1000$ and $N_{voxel} = 500$ is defined as the sum of two Gaussian density functions of $d = v/N_{voxel} - L/N_{LOR}$ with standard deviations 0.0005 and 0.01, respectively. One Gaussian is significantly wider than the other, which may be interpreted as the scattered contribution, while the other Gaussian may refer to the direct contribution. The reference activity is another simple function of Figure 1.8. The measured values are obtained by sampling Poisson distributed random variables setting their means to the product of the SM and the reference activity (left of Figure 1.8). We use two reference activities, where the second is equal to the first one multiplied by 10. The first represents a low-dose case where Poisson sampling introduces significant amount of noise, while the second a high-dose case, where the Poisson-noise is moderate (Figure 1.8).

2D case

In the 2D case, the SM of dimensions $N_{\text{LOR}} = 2115$ and $N_{\text{voxel}} = 1024$ is defined similarly to the 1D case: a weighted sum of two Gaussian density functions of the distance between the LOR and the voxel with FWHMs equal to the detector size and to five times the detector size, respectively (Figure 1.9). Again, the wider Gaussian may be interpreted as the scattered contribution, while the narrower Gaussian may refer to the direct contribution. The Gaussian of the direct contribution is given 60% weight and the scattered contribution 40%, which reflects the typical ratios in human PETs.

The reference activity is a simple function defined by two hot rectangles of Figure 1.9. Similarly to the 1D case, the measured values are generated by sampling the product of the reference activity and the SM.

Simulated measurements

Positron range, geometric projection, direct component, scattering in the measured object and detector model were evaluated using the SM approximations $\mathbf{D} \cdot \mathbf{P}$, $\mathbf{G}$, $\mathbf{D}$, $(\mathbf{D} + \mathbf{S}) \cdot \mathbf{P}$, $\mathbf{L} \cdot \mathbf{G}$ respectively (see Equation 1.15 in Section 1.2.6 for the notations). GATE has built-in detector model, we set LYSO crystal according to the real scanners of Section 1.1.2. Ideally black detectors were modeled by a hypothetical crystal material with practically zero mean free path length. Other image degrading effects, such as random coincidences and detector dead-time were turned off.

We used the following numerical phantoms:

- an **Off-axis point** source of 0.1 MBq activity, placed 2 mm North and 1 mm East from the axis,

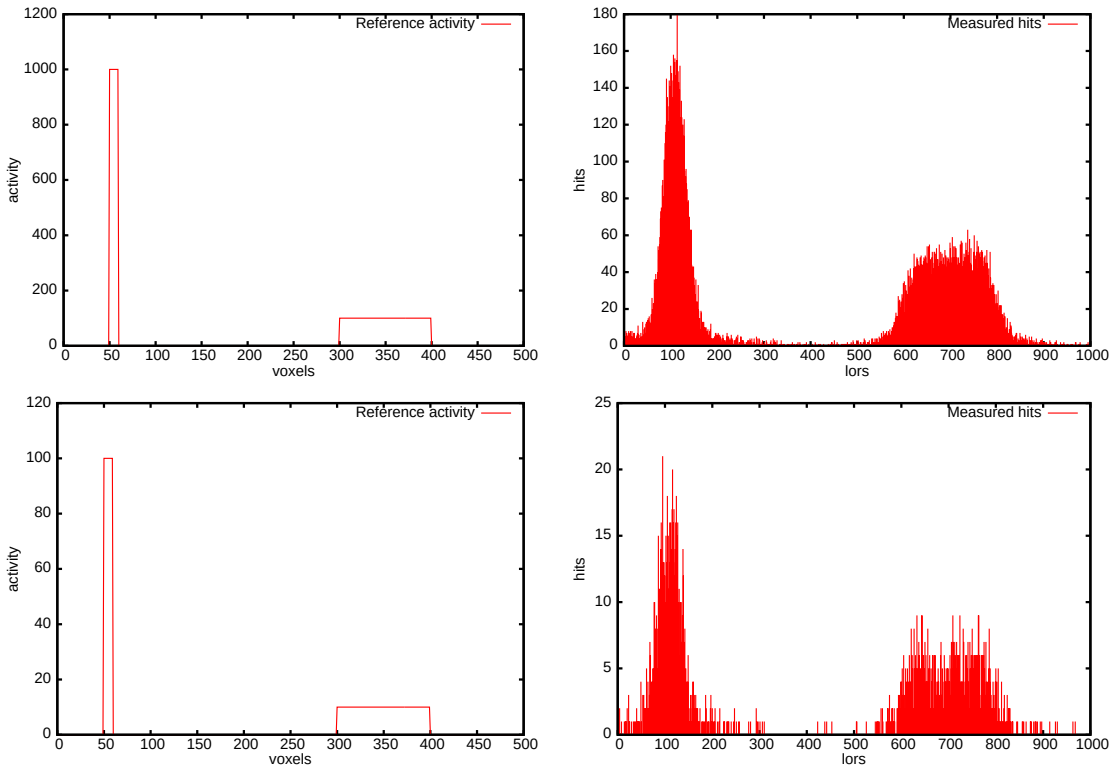


Figure 1.8: The measured phantom (left) and the distribution of the hits in different LORs (right). The upper and lower rows show a high-dose and a low-dose case, respectively.

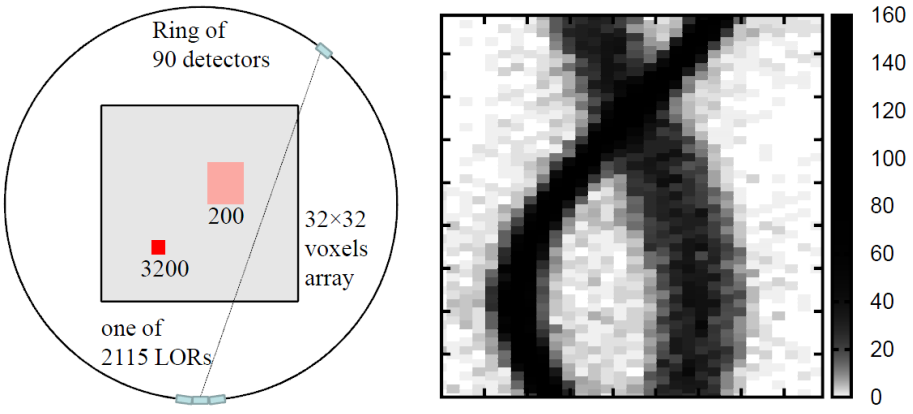


Figure 1.9: A simple 2D tomograph model used in our experiments (left). The detector ring contains 90 detector crystals and each of them is of size 2.2 in voxel units and participates in 47 LORs connecting this crystal to crystals being in the opposite half circle, thus the total number of LORs is $90 \times 47/2 = 2115$. The voxel array to be reconstructed is in the middle of the ring and has 32×32 resolution, i.e. 1024 voxels. The ground truth voxel array has two hot squares, one is of 6×6 voxels where each voxel's activity is 200, the other is of 2×2 voxels where each voxel's activity is 3200. Its measured projection involving Poisson noise is shown in sinogram form in the image on the right, where the horizontal and vertical axes correspond to the signed distance of the line from the center and the angle of the line, respectively.

- a **Ring** phantom of homogeneous activity,
- the **Homogeneity** phantom, built of 8 constant activity cubes with 1.6 MBq activity in total,
- a **Derenzo-like** phantom with rod diameters $1.0, 1.1, \ldots, 1.5$ mm in different segments, virtually filled with 1.6 MBq activity,
- a **Cylinder** phantom that contains a hot and a cold smaller cylinder embedded in the large cylinder and
- the **NEMA NU-2 2007 Human Image Quality (IQ)** phantom [Ass07].

The phantoms are depicted in Figure 1.10. Note that we omitted the off-axis point from the figure for obvious reasons. The duration of the simulated measurement was a varying parameter and given in the Results sections.

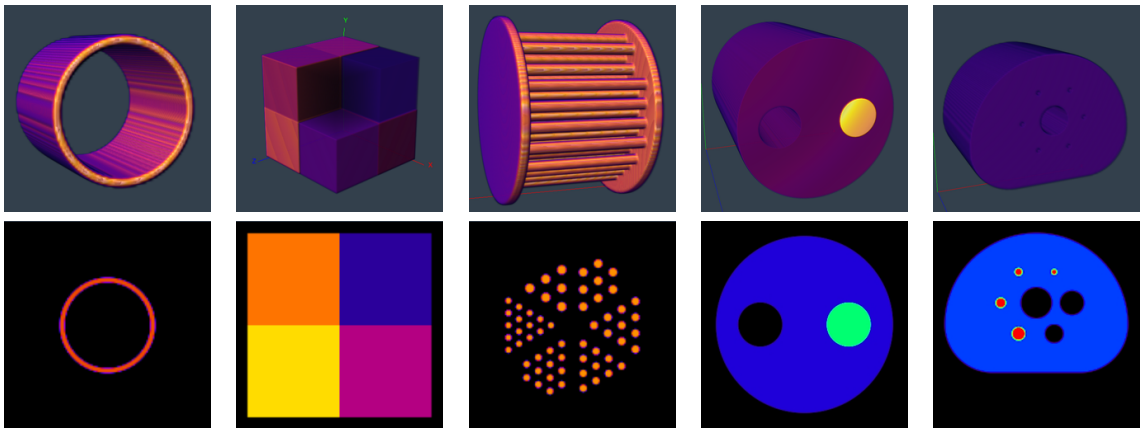


Figure 1.10: Numerical phantoms used in GATE in 3D (upper row) and their relevant slices (lower row).

1.4.2 Distance and error metrics

In order to quantitatively asses the precision of the reconstruction, we used several distance metrics for the actual voxel array and the ground truth solution. The following distance metrics between the n-dimensional simulation vector $\mathbf{s}$ and the n-dimensional phantom, $\mathbf{s}, \mathbf{p} \in \mathbb{R}^n_{\geq 0}$ were used:

1. L_2 error:
$$Error_{L_2}(\mathbf{s}, \mathbf{p}) = 100 \cdot \frac{||\mathbf{s} - \mathbf{p}||_2}{||\mathbf{p}||_2} = 100 \cdot \sqrt{\frac{\sum_{i=1}^{n} (\mathbf{s}_i - \mathbf{p}_i)^2}{\sum_{i=1}^{n} \mathbf{p}_i^2}}.$$
2. *Cross Correlation* (CC) error:
$$Error_{CC}(\mathbf{s}, \mathbf{p}) = 100 \cdot \left(1 - \left|\frac{C_{12}}{\sqrt{C_{11} \cdot C_{22}}}\right|\right),$$
$$C_{11} = \sum_{i=1}^{n} (\mathbf{s}_i - \bar{\mathbf{s}})^2, \quad C_{22} = \sum_{i=1}^{n} (\mathbf{p}_i - \bar{\mathbf{p}})^2, \quad C_{12} = \sum_{i=1}^{n} (\mathbf{s}_i - \bar{\mathbf{s}})(\mathbf{p}_i - \bar{\mathbf{p}}),$$

where $\bar{\mathbf{v}}$ ($\mathbf{v} \in \mathbb{R}^n$) denotes the average of the vector elements:

$$\bar{\mathbf{v}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{v}_i.$$

1.5 Implementation environment

All the methods presented in the dissertation were implemented in CUDA [NVI07] and integrated into the TeraTomo^{TM} [C5] system which served as the prototype for the PET reconstruction framework of Mediso's nanoScan-PET/CT [Med10b]. Tests were run on an NVIDIA GeForce 690 GTX GPU — similar hardware is integrated into the nanoScan-PET/CT. We note that several components of the reconstruction algorithm, e.g. the geometric projection may be implemented in the traditional graphics pipeline of the GPU [C1] instead of CUDA but this would degrade overall performance due to context switches between the two platforms.

1.6 Thesis outline

This thesis work presents efficient techniques to model physical effects of PET, particularly suited for parallel implementation on modern GPUs, as well as sampling techniques that make iterative reconstruction more accurate with practically no additional cost.

The second chapter provides an analysis of the behavior of Monte Carlo sampling in iterative ML-EM reconstruction and justifies the benefits of on-the-fly approximation of the SM both on a mathematical and on an experimental basis. Chapters 3–7 stand for thesis groups.

The first thesis group addresses positron range modeling and presents a solution as spatially varying blur in the frequency domain. The second thesis group proposes two different approaches for geometric projection: a LOR centric method that serves as a fully GPU-conform forward projector and a voxel centric method that is particularly efficient in back projection and also provides excellent performance for small objects as a forward projector. The third thesis group proposes an efficient algorithm to estimate the scattered component in the measured object up to an arbitrary number of scattering events and also introduces a novel way to include the missing higher order scattering with no additional cost. The fourth thesis group addresses scattering and attenuation in the detectors and shows how the combination of pre-computation and Monte Carlo sampling can provide low run-time computational costs even for very small detector sizes. The final thesis group proposes sampling techniques for iterative PET reconstruction that can significantly decrease the error of integral estimators with negligible computational cost. These methods include the application of filtered sampling that suppresses noise and high frequency details that are mainly responsible for sampling errors; the use of multiple importance sampling that combines the benefits of different sampling approaches; and efficient ways of exploiting samples of previous iteration steps as well as eliminating the bias of the tracer density estimation. Theses are also summarized in the last chapter of the dissertation.

Chapter 2

Monte Carlo sampling in the ML-EM scheme

Using numerical quadrature in an iterative process such as the ML-EM, even a small approximation error can accumulate unacceptably. There is an important difference between applying Monte Carlo for estimating a quadrature and using Monte Carlo as a part of an iteration process [J5, J6]. While the goal is an integral quadrature, the convergence rate is known and the error can be minimized by variance reduction techniques and increasing the number of samples. After a review of Monte Carlo integration in Section 2.1, in Section 2.2 we demonstrate that when Monte Carlo is applied in an iteration, the accuracy of a single estimate is not so relevant since later iteration steps may correct the error of an earlier estimate. However, decreasing the samples in a single step means that we can make more iterations under the given budget of samples or computation time. Furthermore, we also investigate the potential of using simplified back projection matrices to speed up the projection.

2.1 Review of Monte Carlo integration

The fundamental idea of the *Monte Carlo* (*MC*) *quadrature* is to express the integral as an expected value, which is then estimated by the average of random samples:

$$\int f(\mathbf{z})\mathrm{d}\mathbf{z} = \int \frac{f(\mathbf{z})}{p(\mathbf{z})} p(\mathbf{z})\mathrm{d}\mathbf{z} = E\left[\frac{f(\mathbf{z})}{p(\mathbf{z})}\right] \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(\mathbf{z}_i)}{p(\mathbf{z}_i)} = I_N, \tag{2.1}$$

where $p(\mathbf{z}_i)$ is a probability density and sample points $\mathbf{z}_1, \ldots, \mathbf{z}_N$ are selected randomly according to this probability density. The convergence to the true value of the integral as the number of samples approaches infinity is ensured by the *law of large numbers*.

To examine the error of the estimate for a finite set of samples $\mathbf{z}_1, \ldots, \mathbf{z}_N$ in Equation 2.1, suppose that the variance of $f(\mathbf{z})/p(\mathbf{z})$ is σ^2. Assuming that samples $\mathbf{z}_i$ are independent random variables, the variance of estimator I_N becomes

$$D^2[I_N] = \frac{1}{N^2}\sum_{i=1}^{N} \sigma^2 = \frac{\sigma^2}{N}. \tag{2.2}$$

The *central limit theorem* tells us that I_N always has normal distribution regardless the distribution of samples $\mathbf{z}_i$. The estimation error of I_N thus can be bounded by $\sigma/\sqrt{N}$ with high confidence. This shows that MC integration avoids the dimensional explosion of classical quadrature rules, since the error is independent of the dimension of the integral. On the other hand, the convergence is relatively slow, as it scales like $1/\sqrt{N}$. Proposed error reduction techniques try to reduce variance σ^2 of the estimator, either by obtaining a better sample distribution as in *importance sampling* or *low discrepancy sampling* [J1], or filtering the integrand $f(\mathbf{z})$ leading to *filtered sampling* (Section 7.1).

2.1.1 Importance sampling

Examining Equation 2.1 we can observe that samples corresponding to larger absolute values of integrand $f(\mathbf{z})$ contribute more to the sum and thus have higher impact on the error. Therefore, it is worth spending more samples for the "important", i.e. higher absolute value regions of the integrand. In other words, the variance of estimator I_N is reduced by making probability density $p(\mathbf{z})$ more proportional to the integrand and as large as possible. This approach is called *importance sampling* [Sob91, SSSK04]. Why p cannot be exactly proportional to f in practice can be understood if we consider that the generation of samples according a probability density requires the inverse of the cumulative probability distribution. This would assume p and thus f to be analytically integrable — making MC integration completely superfluous in this case.

Importance sampling strategies of PET are based on the fact that the integrand is a product of different factors (see Section 1.2.5) and for some of these we can compute the inverse of their integral analytically. Examples include sampling the path free path length of photons in direct particle simulations [J4] or sampling according to the discretized form of the measured inter-crystal scattering probabilities [C6] (Section 6.2). Additionally, as the expected number of hits $\tilde{y}_L$ are expressed as an integral of the product of the positron density $x(\vec{v})$ and the scanner sensitivity $\mathcal{T}(\vec{v} \to L)$ (see Equation 1.7), a widely used approach in the forward projection step of the ML-EM is to sample according to the current estimation of the positron density $x_V^{(n)}$ (Section 4.2.2).

2.1.2 Direct Monte Carlo particle tracing

The traditional solver for the particle transport problem is the *Direct Monte Carlo* (*DMC*) method. DMC generates particles in the volume of interest and simulates their path according to the laws of physics. For the sake of simplicity, in the following we discuss DMC for the case of Photon Tracing (PT), generalization to include positron tracing is straightforward.

In DMC Photon Tracing, first annihilation point $\vec{v}$ is sampled with a density that is proportional to the activity, then the paths of the two annihilation photons are obtained with scanner sensitivity $\mathcal{T}(\vec{v} \to L)$. To do this, an initial direction is drawn from uniform distribution. Two photons are started from the annihilation point and their free paths are sampled to find the photon–material interaction points. At interaction we randomly decide whether absorption or scattering happens with the probability $a = \frac{\sigma_s}{\sigma_t}$. In case of absorption, the photon path is terminated and no LOR is scored. In case of scattering a new direction is generated mimicking the Klein-Nishina formula, and the photon energy is adjusted according to the Compton law. When one of the photons leaves the detector or its energy drops below the discrimination threshold, the photon pair is lost and no LOR is contributed. If photons hit the detector surface, the simulation of this path is terminated and the affected LOR is given contribution $\mathcal{X}/N_{\mathrm{PT}}$ where $\mathcal{X}$ is the total activity and N_{PT} is the number of simulated paths.

There are two main advantages of DMC: it provides a *physically plausible* model, and since it samples particles with a density that is proportional to the activity, it is very efficient in reconstructing small, point like sources. Although its computational burden has been traditionally considered too high for online execution, the computing capacity of GPUs has recently enabled on-the-fly DMC implementations in an iterative reconstruction [WCK$^+$09, LCLC10, KY11b]. DMC methods, on the other hand, have several drawbacks as well:

- As particles travel random paths, DMC particle tracing is a typical instance of *scattering type* algorithms and thus cannot fully utilize the capabilities of the GPU. Recent GPU-based DMC implementations have reported a performance of approximately 10 million paths traced in a second [KY11b]. Considering that state of the art scanners consist of hundreds of millions of LORs and that we wish to spend only a few seconds to compute an iteration step, this practically means that each LOR is approximated by roughly one path sample in average. For comparison, LOR driven projectors are capable of computing

hundreds of path samples per LOR in the same time budget, with each path collecting many annihilation events.

- DMC photon tracing is *not factored*, in the sense that it computes direct and scattered contribution together — however, it may be used in any of the factored phases. This has the important consequence that we cannot distribute samples individually between the different phenomena. For example, motivated by the fact that scattering is a low frequency phenomenon, we may wish to decrease the sampling density of scattered paths. However, with DMC methods this also results in a coarser sampling of the direct contribution. Another important consequence of not being factored is that full DMC methods cannot benefit from the reuse of paths (Section 1.2.5), a very powerful tool of factored methods to greatly increase performance.
- As a high portion of photons leave the system unnoticed without hitting any of the detectors, DMC methods may waste significant amount of computational time for *samples that give no contribution* to the detectors. However, this is also true for many of the fully factorized approaches.

2.2 Error and convergence analysis of the ML-EM iteration

System Matrix (SM) estimations may be different in forward projection and back projection, and due to the numerical errors both differ from the exact matrix $\mathbf{A}$. Let us denote the forward projection SM by $\mathbf{F} = \mathbf{A} + \mathbf{\Delta F}$ and the back projection estimation by $\mathbf{B} = \mathbf{A} + \mathbf{\Delta B}$.

We use the following notations for the normalized back projectors

$$\bar{\mathbf{A}}_{LV} = \frac{\mathbf{A}_{LV}}{\sum_{L'} \mathbf{A}_{L'V}},\ \bar{\mathbf{B}}_{LV} = \frac{\mathbf{B}_{LV}}{\sum_{L'} \mathbf{B}_{L'V}} \implies \bar{\mathbf{B}} = \bar{\mathbf{A}} + \mathbf{\Delta}\bar{\mathbf{B}}.$$

Note that in this case

$$\mathbf{\Delta}\bar{\mathbf{B}} \cdot \mathbf{1} = \mathbf{0}$$

since

$$\sum_L \bar{\mathbf{A}}_{LV} = \frac{\sum_L \mathbf{A}_{LV}}{\sum_{L'} \mathbf{A}_{L'V}} = \sum_L \bar{\mathbf{B}}_{LV} = \frac{\sum_L \mathbf{B}_{LV}}{\sum_{L'} \mathbf{B}_{L'V}} = 1.$$

The question is how these approximations modify the convergence and the fixed point of the iteration scheme. Let $\mathbf{x}^*$ denote the true solution of the ML-EM scheme, which satisfies:

$$\mathbf{A}^T \cdot \frac{\mathbf{y}}{\mathbf{A} \cdot \mathbf{x}^*} = \mathbf{A}^T \cdot \mathbf{1}. \tag{2.3}$$

Let us express the activity estimate in step n as $\mathbf{x}^{(n)} = \mathbf{x}^* + \mathbf{\Delta x}^{(n)}$. Substituting this into the iteration formula

$$\frac{\mathbf{x}^{(n+1)}}{\mathbf{x}^{(n)}} = \frac{\mathbf{B}^T \cdot \frac{\mathbf{y}}{\mathbf{F} \cdot \mathbf{x}^{(n)}}}{\mathbf{B}^T \cdot \mathbf{1}}$$

and replacing the terms by first order Taylor's approximations we obtain:

$$\mathbf{\Delta x}^{(n+1)} \approx \left(\mathbf{1} - \langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L^2} \rangle \cdot \mathbf{F} \right) \cdot \mathbf{\Delta x}^{(n)} + \langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta}\tilde{\mathbf{y}}}{\tilde{\mathbf{y}}} - \mathbf{\Delta}\bar{\mathbf{B}}^T \cdot \frac{\mathbf{y}}{\tilde{\mathbf{y}}}.$$

where $\langle x_V^* \rangle$ is an N_{voxel}^2 element diagonal matrix of true voxel values, $\langle \frac{y_L}{\tilde{y}_L^\alpha} \rangle$ is an N_{LOR}^2 element diagonal matrix of ratios $\frac{y_L}{\tilde{y}_L^\alpha}$, and $\mathbf{\Delta}\tilde{\mathbf{y}} = \mathbf{\Delta F} \cdot \mathbf{x}$ is the error of the expected LOR hits made in the forward projection. Note that Taylor's approximation is acceptable only if function $1/y$ can be well approximated by a line in $\tilde{y}_L \pm \Delta\tilde{y}_L$. The iteration is convergent if

$$\mathbf{T} = \mathbf{1} - \langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L^2} \rangle \cdot \mathbf{F} \tag{2.4}$$

is a *contraction* after certain number of iterations (note that $\mathbf{T}$ is not constant but depends on $\mathbf{x}^{(n)}$ via $\tilde{y}_L$). Even for convergent iteration, the limiting value will be different from $\mathbf{x}^*$ due to the errors of the forward and back projections:

$$\mathbf{\Delta x}^{(\infty)} = \mathbf{S} \cdot \left(\mathbf{\Delta \bar{B}}^T \cdot \frac{\mathbf{y}}{\tilde{\mathbf{y}}} - \mathbf{A}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta \tilde{y}}}{\tilde{\mathbf{y}}} \right) \quad \text{where} \quad \mathbf{S} = \left(\mathbf{A}^T \cdot \langle \frac{y_L}{\tilde{y}_L^2} \rangle \cdot \mathbf{A} \right)^{-1}. \tag{2.5}$$

We can make several observations examining these formulae:

1. As measured hits y_L are Poisson distributed with expectations $\tilde{y}_L$, ratios $y_L/\tilde{y}_L$ have expected value 1 and variance $1/\tilde{y}_L$, thus $E[\mathbf{\Delta \bar{B}}^T \cdot \mathbf{y}/\tilde{\mathbf{y}}] = \mathbf{0}$ and even the variance caused by the back projector error diminishes when the measurement is high dose (i.e. $\tilde{y}_L \gg 1$) and thus the result is statistically well defined. Thus, for high dose measurement, the error made in forward projection is mainly responsible for the accuracy of the reconstruction, which adds the following error in each iteration step:

$$\langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta \tilde{y}}}{\tilde{\mathbf{y}}} = \langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta F} \cdot \mathbf{x}}{\tilde{\mathbf{y}}}. \tag{2.6}$$

2. If the back projection accuracy is not so important, it is worth using a modified normalized SM $\bar{\mathbf{B}}$ to increase the contraction of $\mathbf{T}$ and thus speeding up the iteration.

2.2.1 ML-EM iteration using Monte Carlo quadrature

Due to the stochastic nature of PET, both forward projector $\mathbf{F}$ and back projector $\bar{\mathbf{B}}$ are random variables. We use unbiased MC estimates, i.e.

$$E[\mathbf{F}] = \mathbf{A}, \quad E[\bar{\mathbf{B}}] = \bar{\mathbf{A}}.$$

When these estimates are re-made in every iteration, we can choose whether the same random estimate is used in all iterations, the estimate is modified in each iteration, or even between the forward projection and back projection. Note that as we have to re-compute the matrix elements anyway, the computation costs of different options are the same, the algorithms differ only in whether or not the seed of the random number generator is reset. When iterating with a pre-computed or measured matrix, the random estimates are always the same for every iteration and practically even between the two projectors.

The contribution to the error of a single iteration is defined by Equation 2.6. Errors of different iteration steps accumulate. However, the accuracy can be improved if we use an SM estimation where the expectation value of this contribution is zero since it means that the error contributions of different iteration steps compensate each other and we may get a precise reconstruction even with inaccurate SM estimates. So, our goal is to guarantee that

$$E\left[\langle x_V^* \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta F} \cdot \mathbf{x}}{\tilde{\mathbf{y}}} \right] = E\left[\langle x_V^* \rangle \cdot (\bar{\mathbf{A}}^T + \mathbf{\Delta \bar{B}}^T) \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \frac{\mathbf{\Delta F} \cdot \mathbf{x}}{\tilde{\mathbf{y}}} \right] = 0$$

which, taking into account that both the forward projector and the back projector are unbiased estimators, is held if

$$E\left[\mathbf{\Delta \bar{B}}^T \cdot \langle \frac{y_L}{\tilde{y}_L} \rangle \cdot \mathbf{\Delta F} \right] = 0.$$

Note that this is true if the forward projector is statistically independent from the back projector, but is false when they are correlated. This means that it is worth using independent random samples in each iteration and re-sampling even between forward projection and back projections.

To demonstrate this, we analyze a simple analytical problem, presented in Section 1.4.1. In the 1D case, the error of the reconstruction is tested with random SM approximations, which are obtained by replacing the $5 \cdot 10^5$ analytical SM elements by unbiased MC estimates calculated

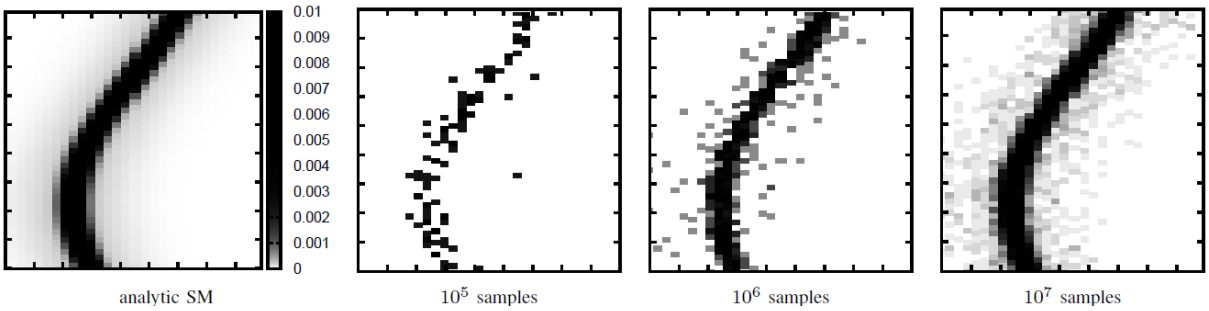


Figure 2.1: One column of the SM: sinograms corresponding to voxel $(10, 10)$, which is the right–upper neighbor of the smaller hot region in Figure 1.8, when the SM is computed analytically or with 10^5 – 10^7 random samples in total. The horizontal and vertical axes of the sinograms correspond to the signed distance of the line from the center and the angle of the line, respectively. Note that when only 10^5 random samples are taken to approximate all of the $2 \cdot 10^6$ SM elements, 95% of the elements get no sample and are replaced by zero, while the remaining 5% are approximated by a constant sample weight that is equal to the sum of all SM elements divided by the number of samples. Increasing the number of samples, more than one sample can contribute to a single SM element, thus zeros and small integer multiples of the sample weight show up.

with 10^4, 10^5, and 10^6 discrete samples in total, respectively. For the $2 \cdot 10^6$ analytical SM elements of the 2D case, the MC estimates consisted of 10^5 – 10^7 samples (Figure 2.1). Note that estimating $5 \cdot 10^5$ ($2 \cdot 10^6$) SM elements of the 1D (2D) case with 10^4 (10^5) discrete samples in total means that most of the SM elements get no sample and thus are replaced by zero, making this a very high-variance estimation that can be considered as a stress test for ML-EM reconstruction.

In the first set of experiments we examine the L_2 error of the reconstruction process of the *fixed case*, i.e. when the same SM approximation is used in all iteration steps (see Figures 2.2 and 2.3 for the 1D and 2D case, respectively). These results indicate that working with the same MC estimate during an EM iteration is generally a bad idea. Reconstructing with a modified SM means that we altered the physical model, so the EM iteration converges to a different solution. *Deterministically matched sampling* takes the same samples in the forward and back projections of a single iteration but regenerates samples for each iteration. Deterministically matched sampling does not help, the error curves are quite similar to those of generated with fixed SM.

Statistically matched sampling, where samples of forward projection are independent of the samples in back projection, has advantages and disadvantages as well. If the sample number is small, then the error curves are strongly fluctuating. The explanation is that matrix $\mathbf{T}$ is just probably a contraction, so the iteration have convergent and divergent stages. If the number of samples is higher, then the iteration becomes stable and its accuracy gets similar to iterating with the analytic SM. Thus, we can conclude that statistically matched sampling is the best option, provided that we are able to guarantee a sufficiently high sampling density.

Figure 2.2 (bottom right) and Figure 2.4 show the reconstruction results after 100 iteration steps for the discussed sampling methods for the 1D and 2D case, respectively, and demonstrate that the fixed and the deterministically matched approaches blur the peaks and edges but are stable, while the statistically matched method behaves similarly to the analytic SM if the sample number is sufficient but may be unstable otherwise, introducing noisy voxels.

2.2.2 Speeding up the convergence with simplified back projectors

We concluded that the reconstruction accuracy of high dose measurements is just slightly affected by the accuracy of the back projector. In a special case when $\mathbf{B} = \mathbf{A} \cdot \mathbf{Z}$ where $\mathbf{Z}$ is an invertible

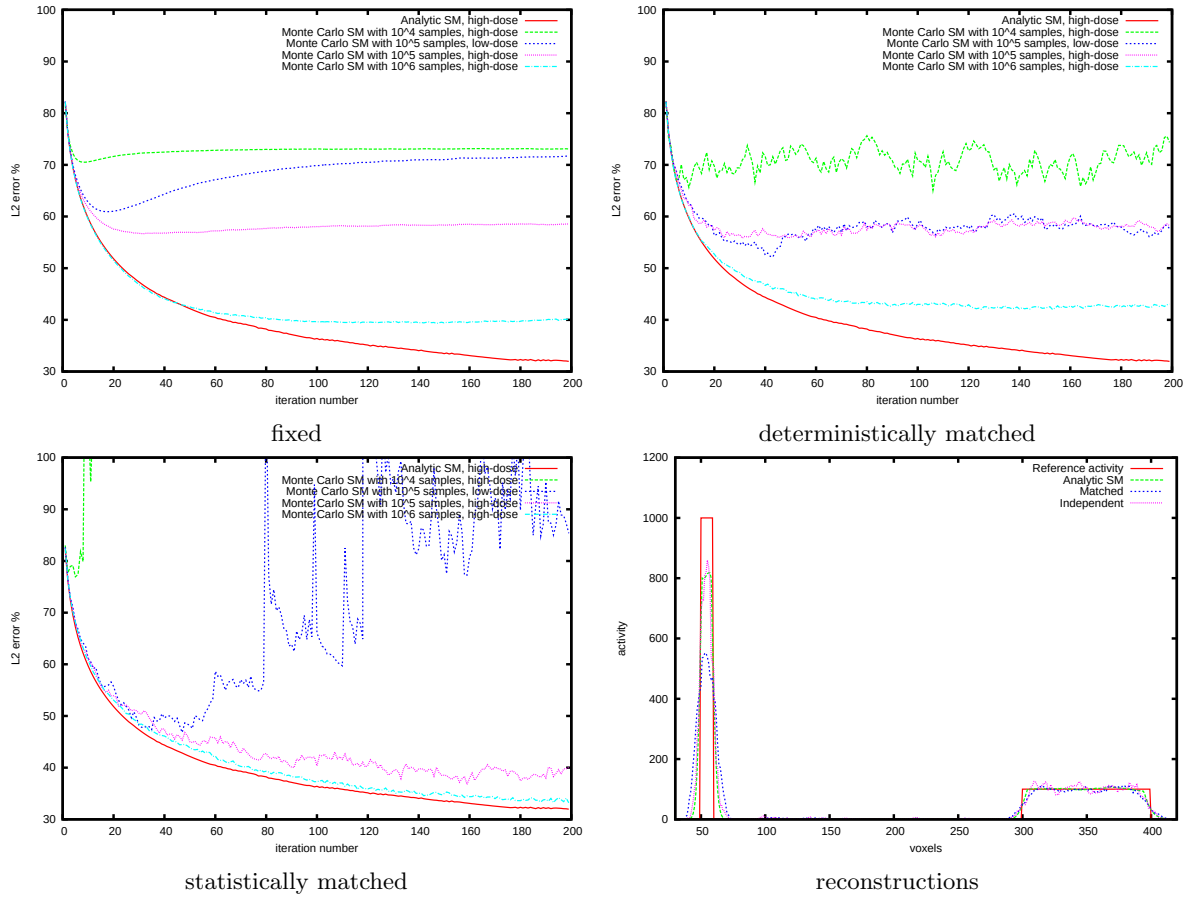


Figure 2.2: Relative L_2 error curves of different sampling strategies and the reconstructed results for the 1D scanner. Fixed case: SM is the same in every iteration step. Deterministically matched stochastic iteration: SM is re-sampled in each iteration step and the forward projector of an iteration step uses the same SM as its back projector. Statistically matched stochastic iteration: SM is re-sampled for every projection, i.e. the forward projection is statistically independent of the back projection.

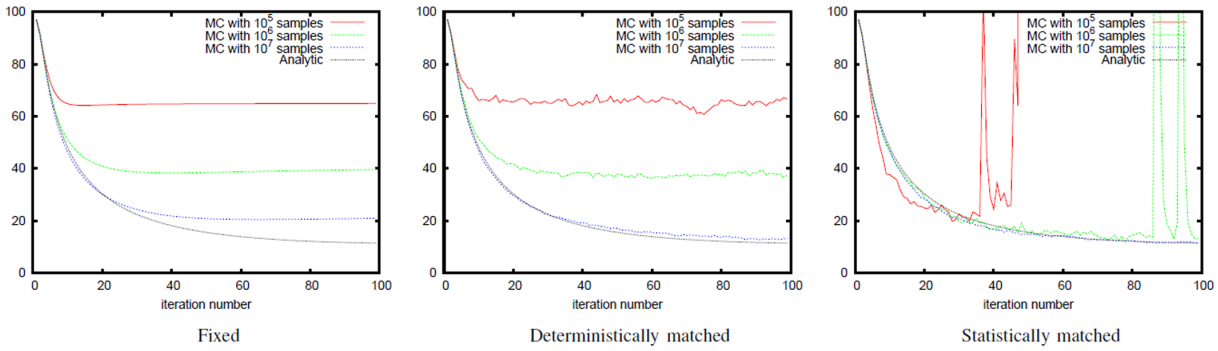


Figure 2.3: Relative L_2 error curves obtained with different sampling techniques in the 2D scanner. Fixed case: SM is the same in every iteration step. Deterministically matched stochastic iteration: SM is re-sampled in each iteration step and the forward projector of an iteration step uses the same SM as its back projector. Statistically matched stochastic iteration: SM is re-sampled for every projection, i.e. the forward projection is statistically independent of the back projection.

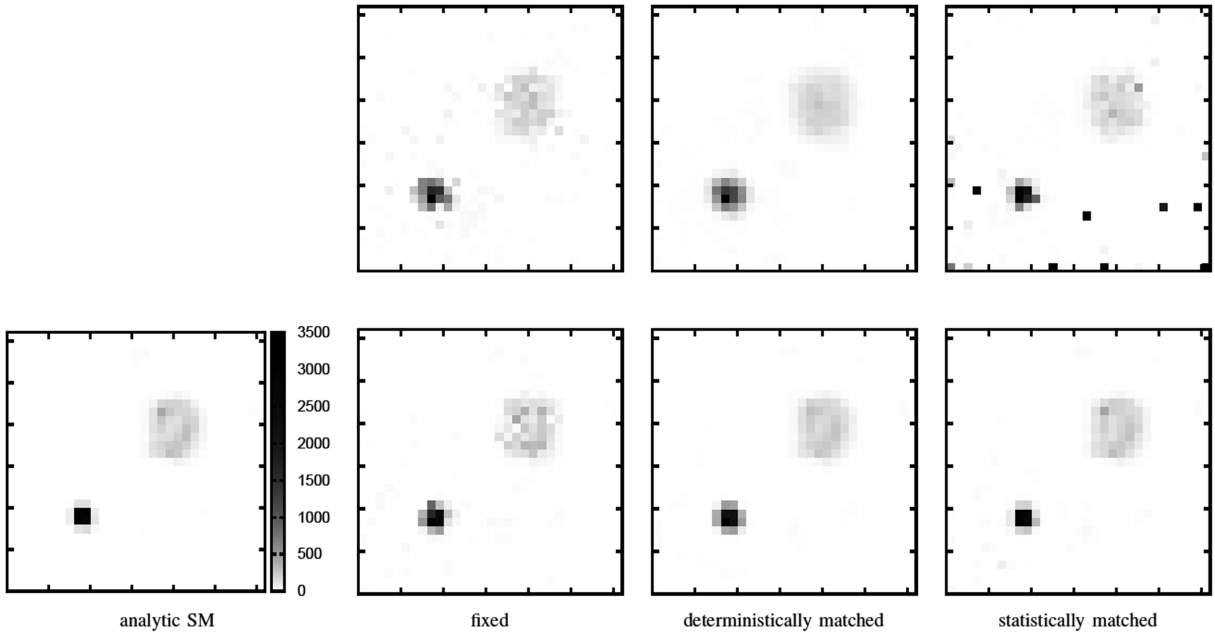


Figure 2.4: Reconstructed activity obtained with 2D analytic SM, 10^5 sample projections (upper row) and 10^6 sample projections (lower row) with the discussed sampling techniques. Note that fixed and deterministically matched iteration are stable but fail to quickly converge to higher peaks. Statistically matched iteration, on the other hand, performs similarly to the analytical SM when the sample number is higher but becomes unstable and generates strong voxel noise when the sample number is smaller.

square matrix of N^2_{voxel} elements, the fixed point is preserved, which can be seen if both sides of Equation 2.3 are multiplied with matrix $\mathbf{Z}$. The convergence speed depends on the contraction of matrix $\mathbf{T}$ (Equation 2.4), which is strong if

$$\langle x^*_V \rangle \cdot \bar{\mathbf{B}}^T \cdot \langle \frac{y_L}{\tilde{y}^2_L} \rangle \cdot \mathbf{A}$$

is close to the identity matrix. We need to find matrix $\mathbf{Z}$ so that for every voxel V just the most significant $\mathbf{A}_{LV}$ elements are kept while others are replaced by zero during the multiplication with $\mathbf{Z}$. As the SM represents a sequence of physical phenomena, this means ignoring voxel space blurring effects, such as positron range.

Using the example of the previous subsection, we examined the convergence of the reconstruction for different activity levels (recall that back projection accuracy becomes important only for low dose measurements).

The results are shown by Figure 2.5. Note that simplified and original back projectors converge to the same result, the approximation is more accurate when the measurement is of high dose. The initial convergence of the simplified back projector is much faster and it becomes poorer only when the iteration overfits the result and therefore the iteration is worth stopping anyway (such overfitting may be avoided with regularization).

2.3 Conclusions

This chapter has investigated the behavior of MC sampling in the iterative ML-EM algorithm. Based on both mathematical derivations and experimental studies, we propose the application of independent sampling and simplified back projector. We can conclude that it is worth to re-compute the SM in every iteration since it has the potential to provide a significantly better

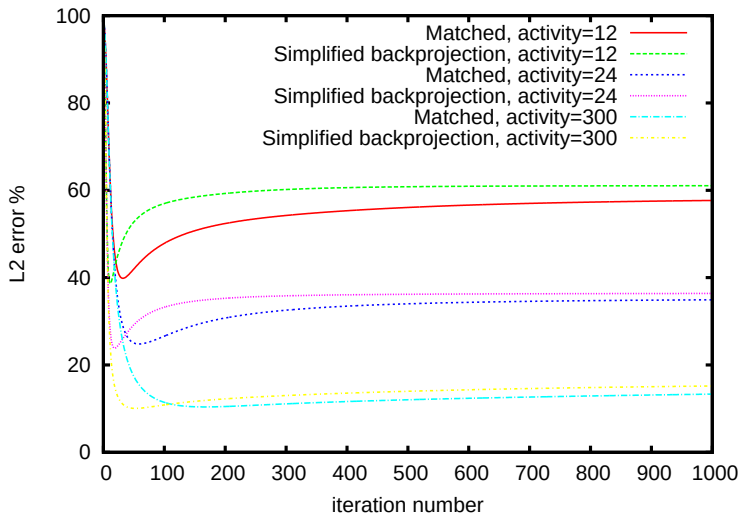


Figure 2.5: Convergence in L_2 for matched and simplified back projectors for different activities.

performance than iterating with a fixed matrix. In addition to parallel computing issues and the inclusion of patient-specific data that were discussed in Section 1.2.4, this gives us the final motivation to develop efficient physical models for PET that can be computed on-the-fly.

The superiority of independent re-sampling is due to the fact that it can gather more information about the system, probably not in a single step but as the iteration proceeds. This additional information helps increase the accuracy. However, independent sampling in forward and back projectors has a drawback that the solution oscillates if the sample density is low, so sample numbers should be carefully selected.

We have also shown that if back projector is properly simplified, then not only its computation can be speeded up, but also the iteration can be made faster.

Chapter 3

Positron Range

Positron range is a phenomenon that the positron is not annihilated at its emission location but moves away from it. Positron range causes a blurring in the reconstruction depending on the kinetic energy of the emitted positron and the material. This blurring is significant in small animal PETs where the voxel size can be an order of magnitude smaller than the average translation of positrons.

Positron range is modeled by conditional probability density $P(\vec{v}_p \to \vec{v}_a)$ of positron annihilation in $\vec{v}_a$ provided that a positron was born in point $\vec{v}_p$. The annihilation density $x^a(\vec{v}_a)$ is obtained from the tracer density $x(\vec{v}_p)$ applying the blurring caused by the positron range:

$$x^a(\vec{v}_a) = \int_{\mathcal{V}} x(\vec{v}_p) P(\vec{v}_p \to \vec{v}_a) \mathrm{d}v_p. \tag{3.1}$$

In the special case when the tissue is homogeneous (which may hold only for small regions but is never met for the entire measured object in practice), we could exploit the translational symmetry of the positron range, i.e. its probability matrix depends just on the distance of positron generation and annihilation:

$$P(\vec{v}_p \to \vec{v}_a) = P(\vec{v}_a - \vec{v}_p)$$

which makes positron range calculation equivalent to a *convolution*

$$x^a(\vec{v}_a) = \int_{\mathcal{V}} x(\vec{v}_p) P(\vec{v}_a - \vec{v}_p) \mathrm{d}v_p.$$

In high-resolution small animal PET systems, the average free path length of positrons may be many times longer than the linear size of voxels. This means that positron range significantly compromises the reconstruction quality if it is not compensated, and also that the material dependent blurring should have a very large support so its voxel space calculation would take prohibitively long. This chapter presents a fast GPU-based solution to compensate positron range effects in heterogeneous media for iterative PET reconstruction.

The performance of frequency domain filtering does not depend on the size of the blurring kernel, but its direct form is ruled out by the fact that we need a spatially variant filtering in heterogeneous media. To handle heterogeneous media, we execute Fast Fourier Transforms for each material type and for appropriately modulated tracer densities and merge these partial results into a density that describes the composed, heterogeneous medium. Fast Fourier Transform requires the filter kernels on the same resolution as the tracer density is defined, so we also present efficient methods for re-sampling the probability densities of positron range for different resolutions and basis functions.

3.1 Previous work on positron range

Analytic methods modeling positron range as a voxel-space blurring operator differ in two aspects: the way the blur is included into the reconstruction and the generalization to heterogeneous material. As Haber et al. showed earlier [HDU90], assuming homogeneous tissue, the blurring due to positron range can be removed by spatial deconvolution. However, deconvolution, when decoupled from the reconstruction method, amplifies image noise [BRL$^+$03]. It is beneficial to integrate the deconvolution into the ML-EM resulting in a two-phase algorithm [ACLO10], where the first phase iterates the standard ML-EM using the annihilation density, from which the positron density is determined in a second ML-EM phase.

Methods that include positron range in an iterative reconstruction apply the blurring in the spatial domain either computing the convolution directly [BRL$^+$03, CGHE$^+$09] or equivalently, multiplying with the pre-computed positron range matrix $\mathbf{P}$ [AM08]. For isotope and material types where the positron range is small, like ^{18}F in bones, the matrix is sparse because the probability that a positron gets far is approximately zero. However, less dense materials like water and air, or isotopes emitting high kinetic energy positrons, like ^{82}Rb, correspond to matrices of much fewer zero elements (Table 3.1). Consequently, applying the blur in the spatial domain may have up to $\mathcal{O}(N_{\text{voxel}}^2)$ complexity which is, assuming high resolution scanners with millions of voxels, unacceptably high for clinical practice. In this chapter, we propose the convolution to be performed in the frequency domain, reducing the complexity to $\mathcal{O}(N_{\text{voxel}} \log N_{\text{voxel}})$, independently of the positron range. This is similar to the method of Haber et al. [HDU90] in the sense of using the Fourier transform, however, we extend it to heterogeneous materials.

Isotope	Mean range (mm)	Max. range (mm)	Kernel size (1^3mm^3 voxel)	Kernel size (0.3^3mm^3 voxel)
^{18}F	0.61	2.3	7^3	17^3
^{15}O	2.00	7.9	17^3	55^3
^{82}Rb	4.24	16.7	35^3	113^3

Table 3.1: Simulated positron range in water according to Cal-González et al. [CGHE$^+$09] and the corresponding size of the blurring kernel in voxels, assuming 1^3mm^3 and 0.3^3mm^3 voxel size.

For the inhomogeneous case with arbitrary geometry, unfortunately there is no unbiased analytic model available [BRL$^+$03]. A blurring operator with shift-variant kernel is locally accurate in a homogeneous neighbourhood, the major challenge is how to mitigate artifacts near material boundaries. Bai et al. [BRL$^+$03] proposes two approaches: an anisotropic truncation of the kernels depending on material type; or to perform successive, material dependent blur with kernels isotropically truncated at a certain radius. Although being more accurate, the latter method requires a prohibitively large number of convolution operations for large values of positron range and high spatial resolution. Alessio et al. [AM08] proposed a rather crude approximation, the coefficients of the sum-of-exponentials probability density model [Der86] (Equation 1.2 in Section 1.1.1) are averaged for the originating and the target voxels, smoothing the artifact near material boundaries. Another approximate approach is to sample the blurring kernel using the water-equivalent distance based on local density, which is equivalent to the distortion of the kernel corresponding to homogeneous water according to the true material [ACLO10]. We note that the method presented in this chapter corresponds to the case of shift-variant kernel without any modifications and thus may produce stronger artifacts at material boundaries than the methods discussed in this paragraph.

3.2 Proposed positron range simulation approach

3.2.1 Probability density re-sampling

In the chapter introduction (Equation 3.1) we discussed that the annihilation density can be modeled as a blurring operator on the tracer density:

$$x^a(\vec{v}_a) = \int_{\mathcal{V}} x(\vec{v}_p) P(\vec{v}_p \to \vec{v}_a) \mathrm{d}v_p.$$

Substituting the finite element approximation of the tracer density, this convolution is expressed by a discrete filtering operation:

$$x^a(\vec{v}_a) = \sum_{V=1}^{N_{\text{voxel}}} x_V \int_{\mathcal{V}} b_V(\vec{v}_p) P(\vec{v}_p \to \vec{v}_a) \mathrm{d}v_p.$$

The finite element coefficient of the annihilation density is computed by multiplying both sides with the *adjoint basis function* $\tilde{b}'_V$, that are orthonormal to the original basis functions, i.e.

$$\int_{\mathcal{V}} b_V \tilde{b}_{V'} \mathrm{d}v = 1 \quad \text{if } V = V' \text{ and zero otherwise.}$$

We use two options, piece-wise constant basis functions when the adjoints are also piece-wise constant basis functions, and tri-linear basis functions when the adjoints are Dirac-delta functions selecting the voxel corners. The result of the scalar product is

$$x^a_{V'} = \int_{\mathcal{V}} \tilde{b}_{V'}(\vec{v}_a) x^a(\vec{v}_a) \mathrm{d}v_a = \sum_{V=1}^{N_{\text{voxel}}} x_V \int_{\mathcal{V}} \int_{\mathcal{V}} \tilde{b}_{V'}(\vec{v}_a) b_V(\vec{v}_p) P(\vec{v}_p \to \vec{v}_a) \mathrm{d}v_p \mathrm{d}v_a = \sum_{V=1}^{N_{\text{voxel}}} \mathbf{P}_{V',V} x_V,$$

where the discrete filter kernel is

$$\mathbf{P}_{V',V} = \int_{\mathcal{V}} \int_{\mathcal{V}} \tilde{b}_{V'}(\vec{v}_a) b_V(\vec{v}_p) P(\vec{v}_p \to \vec{v}_a) \mathrm{d}v_p \mathrm{d}v_a.$$

If piece-wise constant basis functions are used, then matrix element $\mathbf{P}_{V',V}$ is the probability that a positron is annihilated in voxel V' provided that it was born in voxel V.

When the surrounding material is considered homogeneous, the blurring operator degrades to a convolution. The computation of the discrete version of the convolution kernel can also benefit from the spatial invariance. Matrix element

$$\mathbf{P}_{V',V} = \int_{\mathcal{V}} \int_{\mathcal{V}} \tilde{b}_{V'}(\vec{v}_a) b_V(\vec{v}_p) P(\vec{v}_a - \vec{v}_p) \mathrm{d}v_p \mathrm{d}v_a = \mathbf{P}_{O,V_r} \tag{3.2}$$

depends just on the relative location V_r of voxel V' with respect to voxel V, and thus it is enough to compute it for a reference voxel O and all voxels V_r.

In order to simulate positron range, we need effective techniques to implement the filtering operator. We consider two cases, a special one when the material is homogeneous and the filtering becomes spatially invariant, and a general case when the material is inhomogeneous, and the filtering is spatially varying.

3.2.2 Blurring in frequency domain assuming homogeneous material

A convolution can be evaluated both in spatial domain, i.e. voxel space, and in frequency space having applied Fourier transformation. As the computational complexity of filtering in spatial domain is proportional to the product of the voxel numbers in the positron density volume and the filter kernel, spatial filtering gets prohibitively expensive for large kernels. Note that the linear voxel size of small animal PETs may be about 0.1–0.2 mm, while the FWHM of the positron range effect in water is about 1–4 mm depending on the isotope [CGHE+09], thus the required size of the 3D filter kernel is greater than 10^3–10^4 voxels. Approximating the filter kernel by a separable approximation like the Gaussian filter can speed up the process, but the Gaussian would be a rather poor approximation of the positron range phenomenon [LH99]. Fortunately, the convolution can also be evaluated in frequency domain having applied 3D Fast Fourier Transforms $\mathscr{F}$, and the computational complexity of frequency domain filtering is independent of the kernel size:

$$x^a(\vec{v}) = \mathscr{F}^{-1}\left[\mathscr{F}[x(\vec{v})] \cdot \mathscr{F}[P(\vec{v})]\right].$$

The actual form of kernel $P(\vec{v}_a - \vec{v}_p) = P(\vec{v})$ depend on the material–isotope pair. We calculate $P(\vec{v})$ on high resolution off-line with GATE [Jea04] simulations. The noise of MC simulation is filtered out by fitting the simulation data on functions of form

$$P(\vec{v}) = \frac{a\alpha e^{-\alpha|\vec{v}|} + b\beta e^{-\beta|\vec{v}|}}{2\pi|\vec{v}|},$$

which is based on [Der86, PB92, LH99] stating that the probability density of positron range projected onto Cartesian axis X can be well approximated by $p_X(X) = ae^{-\alpha X} + be^{-\beta X}$ where parameters a, α, b, β depends on the material–isotope pair. During fitting, we also impose the requirement that p_X is a probability density, thus it integrates to 1.

Matrix elements $\mathbf{P}_{V',V}$ also depend on the discretization, i.e. the basis functions and on the size of voxels. So, having the probability density of the positron range in a continuous analytical form or defined as histograms of measured data, the positron range effect compensation would require the re-sampling of these functions according to the resolution and the basis functions of the voxel grid (Equation 3.2), and then a convolution operation with the currently estimated tracer density. As in pre-clinical research PETs the voxel size is also a user controllable parameter, the re-sampling is executed on-the-fly applying numerical quadrature when the input data are loaded.

3.2.3 Inhomogeneous material

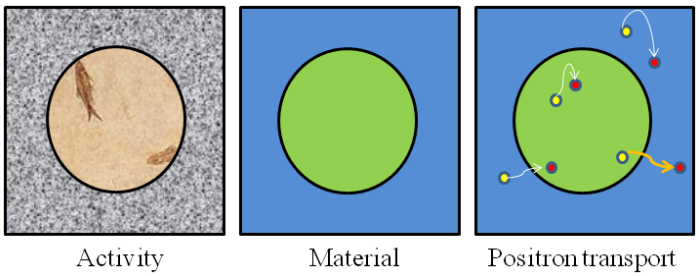


Figure 3.1: Positron range is a spatially variant blurring operator. The path of a positron depends on the material where it is born, on the material where it is annihilated, and also on the material that is visited between the generation and the annihilation.

In inhomogeneous objects, blurring kernel $P(\vec{v}_p \to \vec{v}_a)$ also depends on the material (e.g. bone, air, soft tissue) distribution of the measured object, i.e. the material of every voxel

(Figure 3.1). The precise treatment of this phenomenon would require the consideration of all possible positron paths, which would lead to a high-dimensional integral for every point pair, and would pose prohibitive computational requirements in PET systems. However, assuming that the material is homogeneous, i.e. the blurring kernel is independent of the material type in points $\vec{v}_a$, $\vec{v}_p$ and elsewhere in the object, would be the other extreme approach that would ignore the significantly different probability densities associated with different materials.

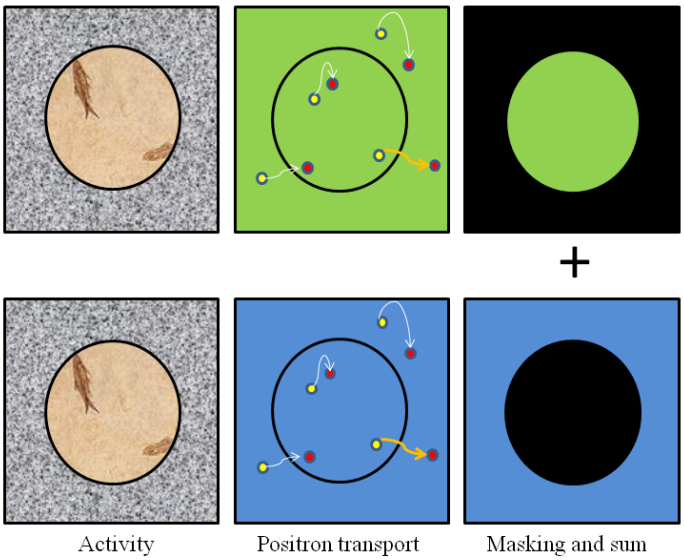


Figure 3.2: Intuitive explanation of the simplified method when the material of the positron generation is used. Blurring each voxel with the filter kernel associated with the material in this voxel means the replacement of a single spatial-variant filtering by one spatial-invariant filtering for each material and a summation.

We propose a practical compromise that is sufficiently accurate for PET reconstruction and can be computed in reasonable time with the support of highly parallel GPU hardware. The basic idea is that instead of considering the material in all points, we take into account the material type only at one end of the positron path. This means that we blur each voxel with the filter kernel associated with the material in this voxel and ignore the fact that there might be a material boundary nearby. This simplification replaces a spatially variant filtering by several spatially invariant convolutions and a summation.

Selecting the material of the positron generation location (Figure 3.2) and denoting the index of the material at point $\vec{v}_p$ by $m(\vec{v}_p)$, we obtain:

$$x^a(\vec{v}_a) \approx \sum_m \int_{\mathcal{V}} x(\vec{v}_p)\xi_m(\vec{v}_p)P_m(\vec{v}_a - \vec{v}_p)\mathrm{d}v_p \tag{3.3}$$

where $\xi_m(\vec{v}_p)$ is an indicator function that is 1 if there is material of index m in point $\vec{v}_p$ and zero otherwise, and $P_m(\vec{v})$ is the probability density of the positron translation between its generation and annihilation in homogeneous material of index m.

Computing Fourier transform $\mathscr{F}$ for each term and then a single inverse Fourier transform, we get:

$$x^a(\vec{v}) \approx \mathscr{F}^{-1}\left[\sum_m \mathscr{F}[x(\vec{v})\xi_m(\vec{v})] \cdot \mathscr{F}[P_m(\vec{v})]\right].$$

Note that this computation requires the Fourier transforms of the blurring functions computed during pre-processing for each material, the Fourier transformation of the positron density once for each material type (usually two or three), and a single inverse Fourier transformation.

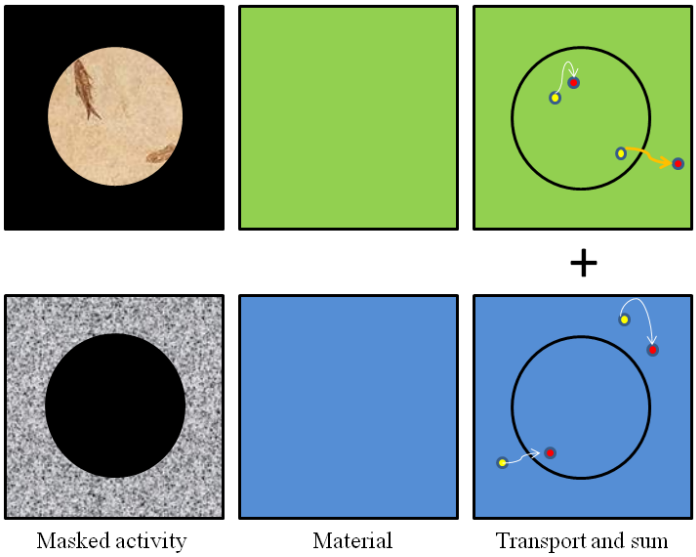


Figure 3.3: Intuitive explanation of the simplified method when the material of the positron annihilation is used. Blurring each voxel with the filter kernel associated with the material in this voxel means the execution of a spatial-invariant filtering for each material and a summation after masking according to the material.

Instead of using the kernel associated with the material of the positron generation location, we can also apply the kernel of the material at the position of the annihilation (Figure 3.3), which leads to the following formula:

$$x^a(\vec{v}_a) \approx \int_{\mathcal{V}} x(\vec{v}_p) P_{m(\vec{v}_p)}(\vec{v}_a - \vec{v}_p) \mathrm{d}v_p = \sum_m \xi_m(\vec{v}_a) \int_{\mathcal{V}} x(\vec{v}_p) P_m(\vec{v}_a - \vec{v}_p) \mathrm{d}v_p.$$

The convolutions in the sum can also be computed via Fourier transformations:

$$x^a(\vec{v}) \approx \sum_m \xi_m(\vec{v}) \mathscr{F}^{-1} \left[\mathscr{F}[x(\vec{v})] \cdot \mathscr{F}[P_m(\vec{v})] \right].$$

This second option is more expensive computationally since here both the number of Fourier transforms and the number of inverse Fourier transforms are equal to the number of materials. The accuracy of the two techniques depends on whether or not the material including most of the radioisotopes occupies a larger part of the object. For typical materials and isotopes, the difference of the reconstructed volumes is negligible.

3.2.4 Positron range in back projection

The ML-EM back projector considers the ratios of measured and estimated LOR values and executes the steps of forward projection backwards in reverse order to update the voxel estimates. The positron range operator can be reversed, the only difference is that if we define kernels according to the material at the location of positron generation in the forward projection, then kernels should correspond to the material at the position of annihilation in the back projection, or vice versa.

As it was shown in Section 2.2.2, however, blurring effects such as the positron range may be skipped in this phase, making forward and back projections *unmatched* [ZG00].

3.3 Results

3D Fourier transformations are computed with the NVIDIA cuFFT library. In this system the positron range calculation for 3 materials at 128^3 and 256^3 resolutions take 0.6 seconds and 2 seconds, respectively.

To demonstrate the potential of the proposed algorithms on pre-clinical scanners, we used Mediso's *nanoScan PET/CT* [Med10b] (see Section 1.1.2). In the first set of experiments, we considered simulation data obtained by GATE.

We reconstructed the *ring phantom* of homogeneous activity put into water and bone materials with and without positron range compensation (Figure 3.4). Note that the homogeneous activity inside the ring could be well reconstructed for ^{18}F and ^{15}O isotopes. The ring geometry also shows up nicely for ^{82}Rb, but the homogeneous activity is compromised at material boundaries. The reason of this artifact is our approximate model which uses the material at the position of the annihilation and assumes that the positron was also born in the same material.

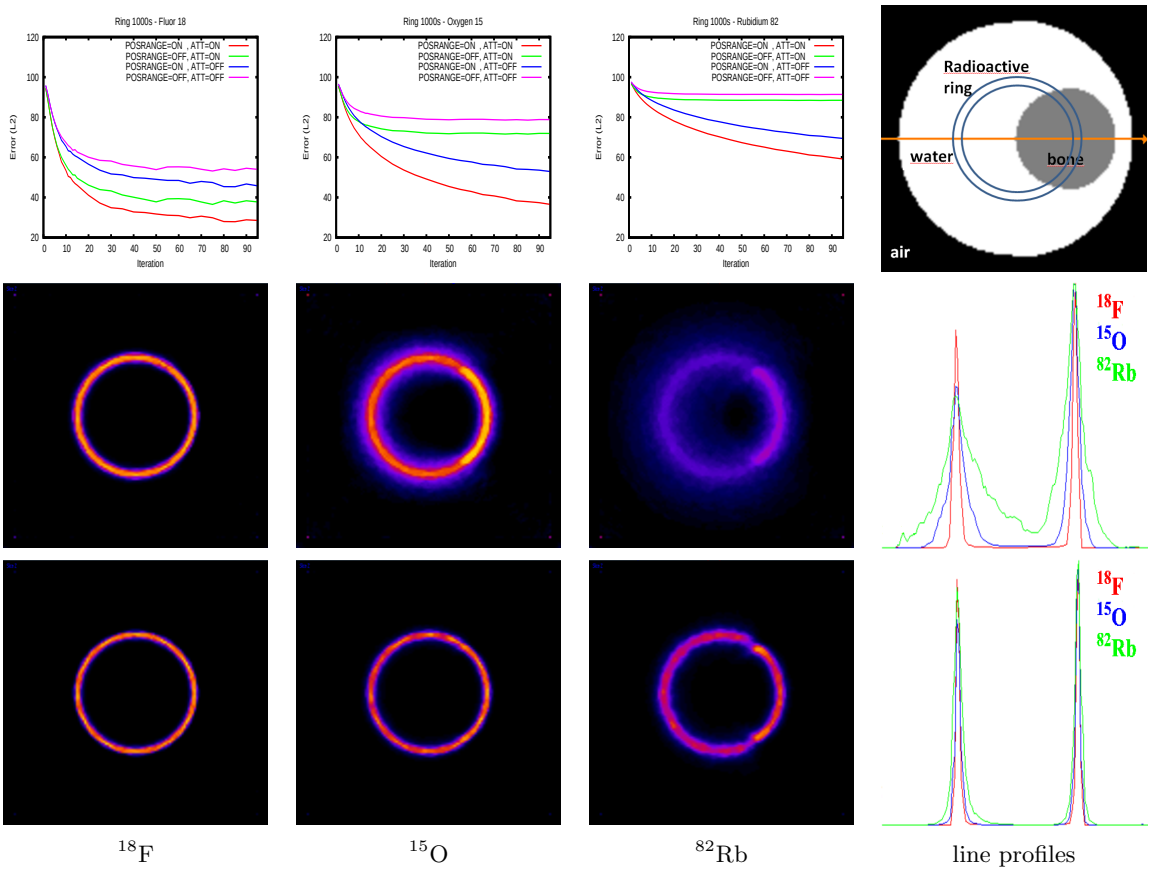


Figure 3.4: Reconstruction of the ring. The upper row contains the L_2 error curves and the material map. The images of the middle and lower rows show the reconstructions without and with positron range compensation, respectively. The first three columns correspond to different isotopes, while the last column shows line profiles.

We also examined real measurements of a *Micro Derenzo phantom* with rod diameters $1.0, 1.1, \ldots, 1.5$ mm in different segments, which was reconstructed at $174^2 \times 146$ resolution (0.15^3 mm^3 voxels) with and without positron range compensation. The transversal slice and the line profiles are shown by Figure 3.5.

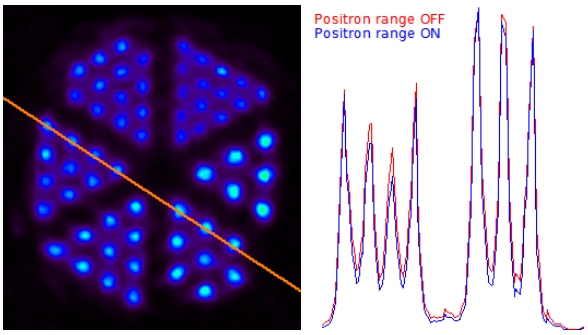


Figure 3.5: Micro Derenzo phantom reconstruction and the line profiles showing the difference produced by positron range compensation.

3.4 Conclusions

This chapter presented an efficient positron range compensation algorithm. The positron range calculation in heterogeneous material is decomposed to a series of positron range calculations in homogeneous materials, once for each material type of the examined object, and a final compositing step. Positron range in homogeneous material, in turn, is evaluated in the frequency domain applying Fast Fourier Transforms. The model runs on the GPU, providing positron range simulations at a negligible cost even for higher volume resolutions.

Chapter 4

Geometric projection

The majority of the measured coincident hits belong to direct photons that reach the detectors without scattering. For this direct component, the System Matrix (SM) is sparse and can be modeled by geometric projection, thus it is worth being treated independently of scattering. Efficient parallel implementation requires the geometric projection to be LOR driven in the forward projector and voxel driven in the back projector. Existing LOR driven forward projector methods use varying sample number to evaluate line integrals and thus assign different computational load to parallel threads causing their divergence. Furthermore, they give analytic solutions to compute the direct contribution, which leads to a biased estimator and thus modifies the fixed point of the iteration. The LOR driven sampling scheme proposed in this chapter offers efficient parallel implementation using the same set of offsets in each thread. Furthermore, we re-sample the surfaces of the detectors in every iteration step of the ML-EM, and use a random offset for the line samples along the line to guarantee that every point that may correspond to a LOR is sampled with a positive probability.

The LOR driven approach may be wasting in the sense that it does not consider the emission density during sample generation. We propose a voxel driven geometric projection scheme that computes the contribution of a voxel to LORs, evenly sampling the detector surfaces. This allows the activity distribution to be taken into account in the forward projection, using importance sampling of the voxels. Furthermore, being a voxel centric approach, it provides an efficient parallel implementation of the back projector.

4.1 Previous work on geometric projection

4.1.1 Direct contribution between detectors

As we shall see later on in this chapter (see Equation 4.4), the direct contribution between two detectors can be expressed as a surface integral on the two detectors, where the integrand contains line integrals between the surface points:

$$\tilde{y}^{geom}_{L(\mathbf{d}_1,\mathbf{d}_2)} = \int_{D_1}\int_{D_2}\int_{\vec{z}_1}^{\vec{z}_2} f(\vec{l})\mathrm{d}l\mathrm{d}z_1\mathrm{d}z_2.$$

In general, these line integrals are computed as a sum of weighted samples taken by marching along the line, often referred to as *ray marching*. Siddon's algorithm [Sid85] is a usual choice for this task. Assuming a piece-wise constant finite-element approximation of the integrand f, the integral can be solved analytically as a sum $\sum l_V f_V$, where l_V is the length of the intersection of the ray with voxel $V = (x, y, z)$ and f_V is the voxel value in V. There are two major criticisms for this approach. The piece-wise constant approximation introduces an unrealistic discontinuity to the model [Jos82]. Furthermore, to evaluate the sum the intersections with the voxels are

computed along the ray which may need different number of loop cycles and divergent conditional instructions for different rays, thus, multiprocessor performance is degraded in a parallel implementation. Even so, there are existing GPU-based iterative PET reconstruction methods that apply Siddon's approach [BVVC12]. Joseph's method [Jos82] applies the trapezoidal rule to numerically estimate the integral and takes one sample per voxel along the ray, which are linearly interpolated from neighbouring values providing a smoother approximation of f. Additionally, the use of equidistant samples is more efficient on GPUs and the linear interpolation is provided with no additional cost, making Joseph's approach more popular in GPU-based iterative PET reconstruction [CM03, BS06]. However, different rays may intersect with different number of voxels causing varying number of loop cycles in threads. Thus, the sampling strategy we propose in Section 4.2.1 takes the same number of samples for every ray.

Surface integrals of the value obtained with the discussed line integral can be estimated by discrete line samples in a line driven method. For performance reasons, usually only a single line sample is used and thus the problem degrades to the computation of a line integral. The *distance-driven approach* [MB04] samples only one endpoint and simultaneously approximates the surface integral of the other endpoint and the line integral. *Solid angle based methods* [QLC+98] approximate surface integrals. In general, the three integrals can be estimated analytically [MDB+08] or with Monte Carlo (MC) quadrature, or we can even mix the two approaches and some integrals are estimated with simple analytical formula while others are computed from random samples. Integrating some variables analytically, we can increase the accuracy when low number of MC samples are used. However, analytical approximations have a deterministic error which makes the method biased, i.e. the error will not converge to zero when the number of MC samples goes to infinity. In order to get an unbiased estimator, Section 4.2.1 proposes random sampling of the detector surfaces and random shifting of the voxel samples along the lines, re-sampled in each iteration step of the ML-EM.

4.1.2 GPU-based projectors

An efficient, gather-style GPU implementation of the forward and back projectors must be LOR driven and voxel driven, respectively. In existing GPU implementations of a geometric forward projector [CM03, BS06, HEV+06], this principle is met. Despite the fact that it fits well to the massively parallel architecture, the LOR driven forward projection may be very inefficient since it completely neglects voxel values, i.e. in terms of importance sampling the sampling density does not mimic the emission density factor of the integrand. In the worst case, when the measured object is a point source, the algorithm wastes most of the samples traversing regions with zero activity while the sampling density around the point source is most likely to be insufficiently low. In Section 4.2.2, we propose a voxel driven projection that may consider the distribution of the activity. Being a voxel driven method, it would require an enormous amount of samples to accurately reconstruct large objects. Section 7.2 describes how to combine the benefits of the two sampling strategies, i.e. providing an accurate reconstruction for both point source-like and large objects with a reasonable amount of samples, according to multiple importance sampling.

Voxel driven back projection approaches [BS06, HEV+06, KY11a] search contributing LORs (for which the volume enclosed by the two endpoints of the LOR intersects with the voxel) by expressing LOR endpoints in polar coordinates, and sample them by looping through angles. Since the axial cross sections of most scanners are not circles but polygons, this sampling of the detectors becomes uneven. The voxel driven projector presented in Section 4.2.2 samples the surface of the detector modules evenly and finds the other endpoint of the LOR via projection through the voxel, leading to more uniform sampling in LOR-space.

4.2 Proposed geometric projectors

If positron range and acollinearity are ignored, the photons generated at the annihilation in $\vec{v}$ have two opposite directions $\vec{\omega}$ and $-\vec{\omega}$ of uniform distribution and this pair contributes to a LOR if the line of place vector $\vec{v}$ and direction $\vec{\omega}$ crosses the surfaces of the LOR's two detectors and none of the photons gets scattered or absorbed (Figure 4.1). Let us denote the intersections of this line with the detector surfaces by $\vec{z}_1$, $\vec{z}_1$, which are unambiguously determined by line point $\vec{v}$ and direction $\vec{\omega}$. As the photon direction is uniform on the half sphere Ω_H of solid angle 2π, the scanner sensitivity assuming zero number of scattering is an integral over the set of directions:

$$\mathcal{T}_0(\vec{v} \to L) = \int\limits_{\vec{\omega}\in\Omega_H} \frac{1}{2\pi} A(\vec{z}_1, \vec{z}_2)\xi_L(\vec{z}_1, \vec{z}_2)\mathrm{d}\omega \tag{4.1}$$

where ξ_L is the indicator function that is 1 if intersection points $\vec{z}_1$ and $\vec{z}_2$ belong to the crystals of LOR L, and

$$A(\vec{z}_1, \vec{z}_2) = \exp\left(-\int\limits_{\vec{l}=\vec{z}_1}^{\vec{z}_2} \sigma_t(\vec{l})\mathrm{d}l\right)$$

is the *attenuation factor*, which expresses the probability that none of the annihilation photons are extincted before they arrive at the detectors with the integral of extinction parameter $\sigma_t(\vec{l})$.

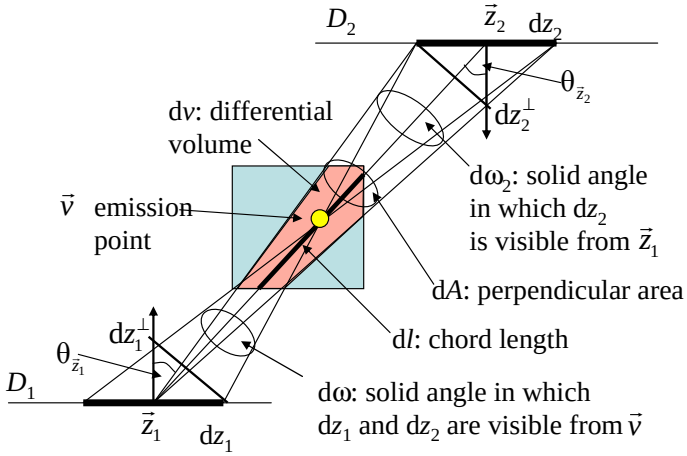


Figure 4.1: Computation of the Jacobian of the change of variables. The differential solid angle at which $\mathrm{d}z_1$ detector surface and $\mathrm{d}z_2$ detector surface are simultaneously seen from emission point $\vec{v}$ is $\mathrm{d}\omega = \mathrm{d}z_1 \cos\theta_{\vec{z}_1}/|\vec{z}_1 - \vec{v}|^2 = \mathrm{d}z_2 \cos\theta_{\vec{z}_2}/|\vec{z}_2 - \vec{v}|^2$. The differential solid angle at which $\mathrm{d}z_2$ is seen from point $\vec{z}_1$ is $\mathrm{d}\omega_2 = \mathrm{d}z_2 \cos\theta_{\vec{z}_2}/|\vec{z}_2 - \vec{z}_1|^2 = \mathrm{d}A/|\vec{z}_1 - \vec{v}|^2$. Finally, the differential volume intersected by lines of $\vec{z}_1$ and $\vec{z}_2$ is $\mathrm{d}v = \mathrm{d}l\mathrm{d}A$, where $\mathrm{d}l$ is the length of the line segment intersecting $\mathrm{d}v$, and $\mathrm{d}A$ is the surface area that is perpendicular to the line.

Including the scanner sensitivity into Equation 4.1, the formula of expected hits becomes

$$\tilde{y}_L^{geom} = \int\limits_{\vec{v}\in\mathcal{V}} x(\vec{v})\mathcal{T}_0(\vec{v} \to L)\mathrm{d}v = \int\limits_{\mathcal{V}}\int\limits_{\Omega_H} \frac{x(\vec{v})}{2\pi} A(\vec{z}_1, \vec{z}_2)\xi_L(\vec{z}_1, \vec{z}_2)\mathrm{d}\omega\mathrm{d}v. \tag{4.2}$$

Evaluating Equation 4.2 directly in a forward projection step leads to scatter-like algorithms, which are not suitable for efficient parallel implementation. Let us change our viewpoint to solve the adjoint problem, i.e. originate photon paths in the detectors and gather the contribution of voxels inside the VOR, expressed as an integral over the detector surfaces. If photon paths are linear, annihilation point $\vec{v}$ and direction $\vec{\omega}$ unambiguously identify detector hit points $\vec{z}_1$ and $\vec{z}_2$, or alternatively, from detector hit points $\vec{z}_1$ and $\vec{z}_2$, we can determine those annihilation

points $\vec{v}$ and directions $\vec{\omega}$, which can contribute: contributing annihilation points are on the line segment between $\vec{z}_1$ and $\vec{z}_2$ and direction $\vec{\omega} = \vec{\omega}_{\vec{z}_1 \to \vec{z}_2}$.

The Jacobian of the change of integration variables, i.e. the *geometry factor* is

$$J^A(\vec{z}_1, \vec{z}_2) = G(\vec{z}_1, \vec{z}_2) = \frac{\mathrm{d}\omega \mathrm{d}v}{\mathrm{d}l \mathrm{d}z_1 \mathrm{d}z_2} = \frac{\cos\theta_{\vec{z}_1} \cos\theta_{\vec{z}_2}}{|\vec{z}_1 - \vec{z}_2|^2} \tag{4.3}$$

where $\theta_{\vec{z}_1}$ and $\theta_{\vec{z}_2}$ are angles between the surface normals and the line connecting points $\vec{z}_1$ and $\vec{z}_2$ on the two detectors, respectively (Figure 4.1). Including the Jacobian of the change of integration variables, the expected number of hits can be expressed as a triple integral over the two detector surfaces D_1 and D_2 of the given LOR and over the line connecting two points $\vec{z}_1$ and $\vec{z}_2$ belonging to the two detectors $\mathbf{d}_1$ and $\mathbf{d}_2$ (Figure 4.1):

$$\tilde{y}^{geom}_{L(\mathbf{d}_1,\mathbf{d}_2)} = \int_{D_1}\int_{D_2} G(\vec{z}_1, \vec{z}_2) X(\vec{z}_1, \vec{z}_2) A(\vec{z}_1, \vec{z}_2) \mathrm{d}l \mathrm{d}z_1 \mathrm{d}z_2 \tag{4.4}$$

where

$$X(\vec{z}_1, \vec{z}_2) = \frac{1}{2\pi} \int_{\vec{z}_1}^{\vec{z}_2} x(\vec{l}) \mathrm{d}l$$

is a line integral of *emission density* $x(\vec{l})$ between endpoints $\vec{z}_1$ and $\vec{z}_2$.

4.2.1 LOR driven sampling

The integral over the pair of detector surfaces can be estimated by N_{ray} discrete line samples, i.e. point pairs $(\vec{u}_i, \vec{w}_1)$, $i = 1, \ldots, N_{\text{ray}}$, on the two detectors. We take N_{march} equidistant points $\vec{l}_{ij}$ along each line segment $(\vec{u}_i, \vec{w}_1)$ and evaluate the line integral with the trapezoidal quadrature. Step size Δl_i can be determined from the length of the line segment where it is inside the volume of interest. Note that this scheme is applicable for any finite function series representation of annihilation density $x(\vec{v})$. Its implementation does not need conditional instructions and is very fast if $x(\vec{l}_{ij})$ is fetched from a 3D texture of the GPU since the probability that neighboring threads need neighboring voxels is high, thus the texture cache works efficiently. Tri-linear interpolation is directly supported by the texturing hardware and higher order spline interpolation can also be effectively traced back to tri-linear interpolation [SH05].

To sample all points with positive probability, we start the ray marching with a random offset that is uniformly distributed in $[0, \Delta l_i]$. With these, the integral estimator is:

$$\tilde{y}^{\mathrm{A1}}_L \approx \frac{D_1 D_2}{N_{\text{ray}}} \sum_{i=1}^{N_{\text{ray}}} \sum_{j=1}^{N_{\text{march}}} G(\vec{u}_i, \vec{w}_i) \frac{x(\vec{l}_{ij})}{2\pi} \Delta l_i A_L(\vec{u}_i, \vec{w}_i).$$

Comparing this estimator to the integrand of Equation 4.2, we can conclude that the weighting scheme of this LOR driven approach is:

$$\tilde{y}^{\mathrm{A1}}_L = \sum_{i=1}^{N_{\text{ray}}} \sum_{j=1}^{N_{\text{march}}} \frac{x(\vec{l}_{ij}) A(\vec{u}_i, \vec{w}_i)/(2\pi)}{d^{\mathrm{A1}}(\vec{l}_{ij}, \vec{\omega}_i)} \quad \Longrightarrow \quad d^{\mathrm{A1}}(\vec{l}, \vec{\omega}) = \frac{N_{\text{ray}}}{D_1 D_2 G(\vec{u}, \vec{w}) \Delta l},$$

where $\vec{u}$, $\vec{w}$ and Δl can be unambiguously determined from the line of $\vec{l}$ and $\vec{\omega}$, and from the geometry of the detector and the volume to be reconstructed. Note that this sampling method does not generate points and directions that do not correspond to a given LOR L, thus indicator function $\xi_L(\vec{u}, \vec{w})$ (Equation 4.1) has value 1 for all samples.

The LOR centric method has several advantages in forward projection. As it is a gathering algorithm, it requires no atomic operations on the GPU. On the other hand, it samples

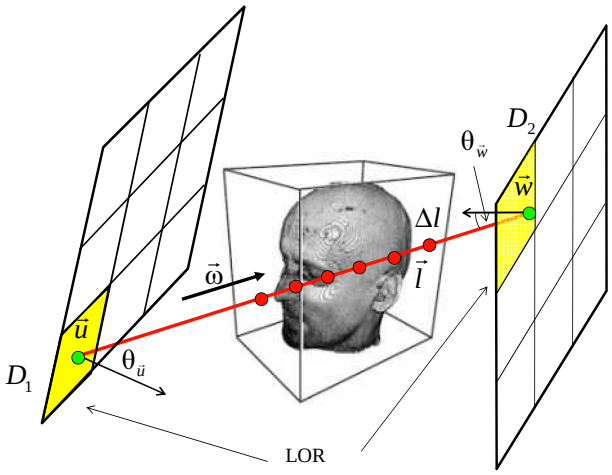


Figure 4.2: A single computational thread of the LOR driven projection takes a detector pair and marches on rays between sample points $\vec{u}, \vec{w}$ of the detector surfaces.

annihilation points occupying the 3D space, which can be well supported by the 3D texture hardware.

The forward projection algorithm executed by a single thread that is responsible for the LOR connecting detector (p, q) and detector (r, s) is listed in the following. For the sake of simplicity, the pseudocode ignores attenuation.

Forward(p, q, $\vec{o}_1$, $\vec{w}_1$, $\vec{h}_1$, $\vec{n}_1$, // *indices and the geometry of the first detector*
r, s, $\vec{o}_2$, $\vec{w}_2$, $\vec{h}_2$, $\vec{n}_2$) // *indices and the geometry of the second detector*
$\tilde{y}[p, q, r, s] = 0$;
for $i = 1$ **to** N_{ray} **do**
$\vec{z}_1 = \vec{o}_1 + \vec{w}_1(p + u_1^{(i)})/N_a + \vec{h}_1(q + u_2^{(i)})/N_t$;
$\vec{z}_2 = \vec{o}_2 + \vec{w}_2(r + u_3^{(i)})/N_a + \vec{h}_2(s + u_4^{(i)})/N_t$;
$\vec{dz} = \vec{z}_2 - \vec{z}_1$;
$G = -D_1 D_2(\vec{n}_1 \cdot \vec{dz})(\vec{n}_2 \cdot \vec{dz})/|\vec{dz}|^4$;
$(\vec{l}_{start}, \vec{l}_{end})$ = Intersection(line $\vec{z}_1 \to \vec{z}_2$, volume cube) + random offset;
$\vec{l}_{step} = (\vec{l}_{end} - \vec{l}_{start})/N_{march}$;
$\Delta l = |\vec{l}_{step}|$;
$X = 0$; // *Line integral:* $X = \int x\mathrm{d}l$
for $(\vec{l} = \vec{l}_{start}; \vec{l}\ != \vec{l}_{end}; \vec{l}\ += \vec{l}_{step})$ **do**
$X\ += x[\vec{l}] \cdot \Delta l$;
endfor
$\tilde{y}[p, q, r, s]\ += G \cdot X/(N_{ray} \cdot 2\pi)$
endfor
end

The "Intersection" function computes the intersection of the line and the axis-aligned cube representing the volume to be reconstructed. N_a and N_t denote the resolution of the detector module in the axial and transaxial directions, respectively. Vectors $\vec{o}_i$, $\vec{w}_i$, $\vec{h}_i$ and $\vec{n}_i$ $(i = 1, 2)$ stand for the corner position, axial, transaxial and normal vector of a detector module.

4.2.2 Voxel driven sampling

In voxel driven sampling, first annihilation point $\vec{v}$ is sampled, then direction $\vec{\omega}$ is obtained based on $\vec{v}$, thus we express the probability density of a complete sample in a product form:

$$p(\vec{v},\vec{\omega}) = p_{\vec{\omega}}(\vec{\omega}|\vec{v})p_{\vec{v}}(\vec{v}).$$

First N_v volume points $\vec{v}_i$ are obtained from a *Volume of Interest* (VoI) with a density that is proportional to the activity according to the principles of importance sampling:

$$p_{\vec{v}}(\vec{v}) = \frac{x(\vec{v})}{\mathcal{X}} \quad \text{where} \quad \mathcal{X} = \int\limits_{\mathcal{V}_{VoI}} x(\vec{v})\mathrm{d}v$$

where $\mathcal{X}$ is the total activity in the VoI.

Note that if the VoI is smaller than the total volume, then this approach does not lead to an unbiased estimator. Thus, if it is used alone, then the VoI should not be focused on a smaller region. However, when this method is combined with other techniques (Section 7.2), then the combined method can be valid even if the individual methods are biased.

The scanner sensitivity is approximated from the solid angle subtended by the two detector surfaces of the LOR from point $\vec{v}$, taking line samples via point $\vec{v}$ (Figure 4.3). If we sample directions by placing uniformly distributed hit points $\vec{u}$ on the detector surface D_1, then the probability density of the line direction is

$$p_{\vec{\omega}}(\vec{\omega}|\vec{v}) = \frac{|\vec{v}-\vec{u}|^2}{D_1 \cos\theta_{\vec{u}}}$$

where $\theta_{\vec{u}}$ is the angle between $\vec{v}-\vec{u}$ and the surface normal of the detector. Note that this sampling method may generate line samples that do not intersect crystal surface D_2 of LOR L. However, as the integrand in Equation 4.2 is zero due to the indicator function for these lines, the expectation gives back the integral.

Putting just a single sample on each detector surface D_1, the expected number of hits in LOR L and the corresponding density are

$$\tilde{y}_L^{\mathrm{A2}} \approx \frac{1}{N_\mathrm{v}} \sum_{i=1}^{N_\mathrm{v}} \frac{x(\vec{v}_i)A(\vec{u}_i,\vec{w}_i)\xi_L(\vec{u}_i,\vec{w}_i)/(2\pi)}{p_{\vec{v}}(\vec{v}_i)p_{\vec{\omega}}(\vec{\omega}_i|\vec{v}_i)} \quad \Longrightarrow \quad d^{\mathrm{A2}}(\vec{v},\vec{\omega}) = \frac{N_\mathrm{v} x(\vec{v})|\vec{v}-\vec{u}|^2}{\mathcal{X} D_1 \cos\theta_{\vec{u}}}.$$

The voxel driven method in forward projection has the advantage that it can focus on high activity regions. Point source like objects can be reconstructed with very few samples. However, it requires atomic operations in forward projection and a single thread accesses many LORs stored in a 4D data structure, which are slow on the GPU. On the other hand, when this scheme is applied to all the voxels in back projection, it fits well to the massively parallel architecture. The pseudocode of the resulting back projection is listed in the following, N_a and N_t denote the resolution of the detector module in the axial and transaxial directions, respectively.

```
Backward(V,                                          // voxel ID
         o1, w1, h1, n1,                             // primary detector
         o2, w2, h2, n2)                             // secondary detector
    Sample v from b_V(v);
    Enum = Denom = 0;
    for p = 0 to N_a − 1 do
        for q = 0 to N_t − 1 do
            z1 = o1 + w1(p + 0.5)/N_a + h1(q + 0.5)/N_t;
            z2 = project z1 through v
            r, s = index of z2
            Δω = D1(n1 · (v − z1))/|v − z1|^3;
```

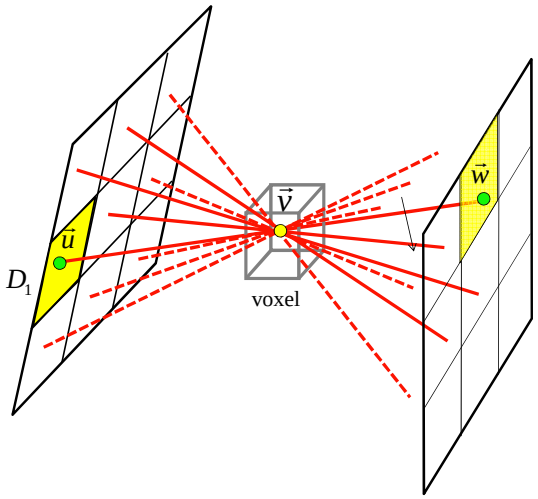


Figure 4.3: A single computational thread of the voxel driven projection samples $\vec{v}$ in proportion to the positron density $x(\vec{v})$ and processes a line crossing this point for each LOR.

```
            if (0 ≤ r < N_t AND 0 ≤ s < N_a) then
              Denom += Δω;
              Enum += Δω · y[p,q,r,s]/ỹ[p,q,r,s];
            endif
          endfor
        endfor
        if (Denom > 0) then x[V] *= Enum/Denom;
    end
```

4.3 Results

The accuracy of geometric projection is crucial in high resolution small animal PET where the voxel edge length can be significantly smaller than the edge length of the detector crystals. Thus, we modeled Mediso's *nanoScan PET/CT* [Med10b] (Section 1.1.2).

To validate the proposed geometric projection methods, we took three different mathematical phantoms, an off-axis *Point source*, the *Derenzo*, and the *Homogeneity* (Section 1.4.1). We used GATE [Jea04] to generate a "ground truth" reference projection $\tilde{y}_L^{\text{ref}}$ with 10^{12} samples and compared the LOR space L_2 error of the proposed projectors calculated with the multiples of $N_{\text{ray}} = 1$, $N_{\text{march}} = 36$ and $N_{\text{v}} = 10^4$. Voxel samples N_{v} of the voxel centric projector were generated with importance sampling, mimicking the current estimation of the emission density. To allow the comparison of techniques working with different sample types, the LOR space L_2 error is depicted in Figure 4.4 with respect to the computation time of a single projection.

In Figure 4.4 we can observe that increasing the computation time and thus the number of MC samples, the error converges to zero in both cases, thus both the proposed voxel driven and LOR driven methods are unbiased estimators. By comparing the computation times needed to reach a given error level, we can note that voxel driven sampling is particularly efficient for the Point, while LOR driven sampling is good for the Homogeneity.

In the second phase of the evaluation, the projectors were included in ML-EM reconstruction. To obtain measured value y_L, we assigned 0.1 MBq activity to the *Point source*, 5 MBq activity to the *Derenzo*, 1.2 MBq activity in total to the *Homogeneity*, and simulated a 1000 sec long measurement for each with GATE, which mimics physical phenomena and thus obtains the

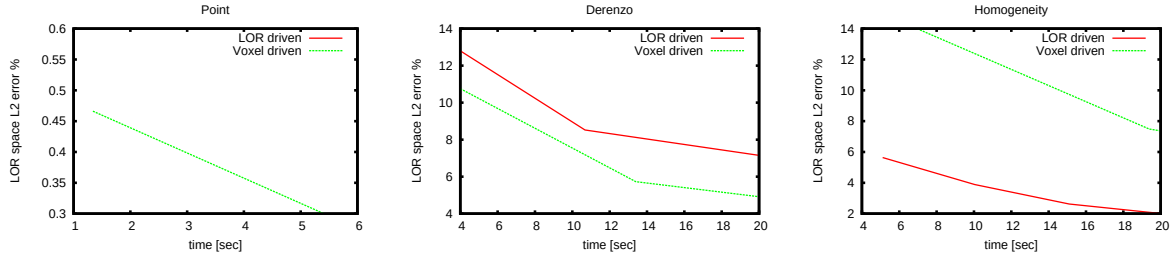


Figure 4.4: LOR space L_2 error of different projectors with respect to the computation time of the projection for the Point (left), Derenzo (middle), and the Homogeneity (right) phantoms. Note that the left-hand figure does not include the curve of the LOR driven sampling because its error is an order of magnitude higher than those of the voxel driven method.

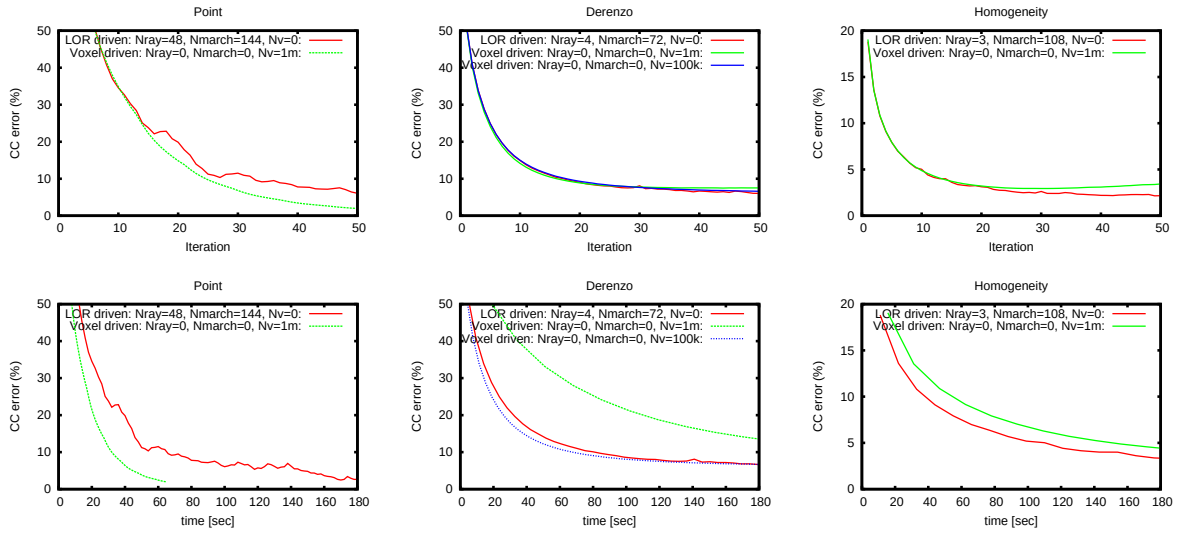


Figure 4.5: Voxel space CC error curves with respect to the iteration number (first row) and to the reconstruction time (second row) of the reconstructed Point (left), Derenzo (middle) and Homogeneity phantoms (right). The error were made with different N_{ray}, N_{march} and N_{v} samples. The method is LOR driven when the number of voxel samples N_{v} is zero. The method is voxel driven when the number of LOR samples N_{ray} is zero. We executed full EM iterations in all cases.

measured data with realistic Poisson noise. The *Signal-to-Noise Ratios* (*SNR*)

$$\mathrm{SNR} = \frac{\sum_{L=1}^{N_{\mathrm{LOR}}} \tilde{y}_L^{\mathrm{ref}}}{\sum_{L=1}^{N_{\mathrm{LOR}}} |y_L - \tilde{y}_L^{\mathrm{ref}}|}$$

of the Point, Derenzo and Homogeneity measurements are 22.99, 10.22 and 4.04, respectively.

From the measured data, the three phantoms are reconstructed on a grid of $144^2 \times 128$ voxels of edge length 0.23 mm. Figure 4.5 shows the voxel space *Cross Correlation* error of the reconstruction for the three phantoms using different N_{ray}, N_{march} and N_{v} parameters as the function of the iteration number. When N_{ray} is zero, the method is voxel driven. When N_{v} is zero, we run a LOR driven algorithm. Note that when too few samples are used, the error curve fluctuates and the algorithm may stop converging after certain steps. The sufficient number of samples depends not only on the resolution of the voxel grid but also on the phantom.

Different methods are associated with significantly different computation times. To show this, in the second row of Figure 4.5 we also include the errors as functions of the time in seconds devoted to execute forward projections. As expected, the Point phantom can be efficiently reconstructed with the voxel driven method, while the LOR centric approach is good for the Homogeneity phantom.

4.4 Conclusions

In this chapter we proposed two different approaches for geometric projection of PET. The LOR driven projection performs uniform sampling in LOR-space, making it suitable for objects with large, homogeneous regions. Additionally, when used in the forward projection, uniform sampling avoids branching and diverging loop cycles in the GPU code thus it can fully utilize the enormous computational power of the massively parallel hardware. Being an unbiased estimator, the method can be included in an iterative reconstruction without modifying the fixed point. In contrast, the proposed voxel driven method may utilize importance sampling to capture fine details of point source like objects even with a few samples. However, it leads to a scattering type algorithm in the forward projector, thus it can take significantly less samples under a given time budget than the LOR driven method. On the other hand, it provides a very efficient, gathering type back projector with nearly uniform sampling of the LORs intersecting a given voxel.

The LOR driven projector performs well for large, low frequency regions, while the voxel driven approach is superior when the tracer is concentrated into small regions. As one method fails where the other is very efficient and vice versa, it is highly desirable to combine their benefits. Fortunately, as it will be demonstrated in Section 7.2, this is possible via the use of *multiple importance sampling*.

Chapter 5

Scattering in the measured object

Scattering means that the photon directions are modified by the material of the examined object. As the average free path length of 511 keV photons in water is about 10 cm, this effect is negligible in small animal PETs where the object size is small, but is significant in human PETs where about 40 % of the detected photons go through at least one scattering.

The solution of the particle transport problem, which is the core part of tomography reconstruction, is mathematically equivalent to the evaluation of a Neumann series of increasing dimensional integrals, where the first term represents the direct contribution, the second the single scatter contribution, the third the double scattering etc. High dimensional integrals are computationally very expensive and unfortunately, they are object-dependent, i.e. no parts of the computations can be ported to an off-line phase without sacrificing accuracy. Thus, this infinite series is truncated after a few (typically after the first or second) terms.

Ignoring in-scattering, the integro-differential equation describing the radiant intensity on a linear path can be solved analytically (Equation 1.5):

$$I(\vec{l}(t),\vec{\omega},\epsilon) = A_\epsilon(t_0,t)I(\vec{l}(t_0),\vec{\omega},\epsilon) + \int_{t_0}^{t} A_\epsilon(\tau,t)I^e(\vec{l}(\tau),\epsilon)\mathrm{d}\tau.$$

The solution obtained without the in-scattering integral can be used to calculate the full solution if we explicitly sample scattering points, apply this formula for the line segments between the scattering points (Figure 5.1), and integrate in the domain of scattering points. Sampling one scattering point between the two detector crystals, we obtain the single scatter contribution, sampling two scattering points, we get the double scatter, etc. The path of the photon pair will be a *polyline* containing the emission point somewhere inside one of its line segments (Figure 5.2). This polyline includes scattering points $\vec{s}_1,\ldots,\vec{s}_S$ where one of the photons changed its direction in addition to detector hit points $\vec{z}_1 = \vec{s}_0$ and $\vec{z}_2 = \vec{s}_{S+1}$. The values measured by detector pairs will then be the total contribution, i.e. the integral of such polyline paths of arbitrary length. When segments are considered, we can use the analytic expression of the solution in Equation 1.5 since the in-scattering integral can be ignored because this contribution is taken into account by other higher order terms. This way, the solution of the transport problem is expressed as a sum of contributions of different path lengths. The terms are increasing dimensional integrals since scattering points may be anywhere. The sum of these integrals is called the *Neumann series*. First, while keeping the discussion as general as possible, we address only the *single scattering problem* ($S = 1$), i.e. when exactly one of the photons scatters, and exactly once.

To express the contribution of a polyline path, we take its line segments one-by-one and consider a line segment as a *virtual LOR* with two virtual detectors of locations, $\vec{s}_{i-1}$ and $\vec{s}_i$, and of differential areas projected perpendicularly to the line segment, $\mathrm{d}A^\perp_{i-1}$ and $\mathrm{d}A^\perp_i$ (Figure 5.2). The contribution of a virtual LOR at its endpoints, i.e. the expected number of photon pairs going through $\mathrm{d}A^\perp_{i-1}$ and $\mathrm{d}A^\perp_i$ is $C(\vec{s}_{i-1},\vec{s}_i)\mathrm{d}A^\perp_{i-1}\mathrm{d}A^\perp_i$, where contribution C is the product of

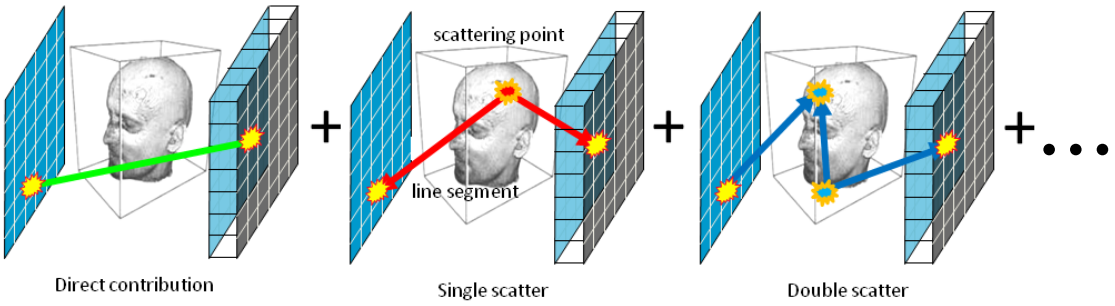


Figure 5.1: Expressing the solution of the multiple scattering problem as a Neumann series corresponds to the decomposition of the path space according to the length of the paths.

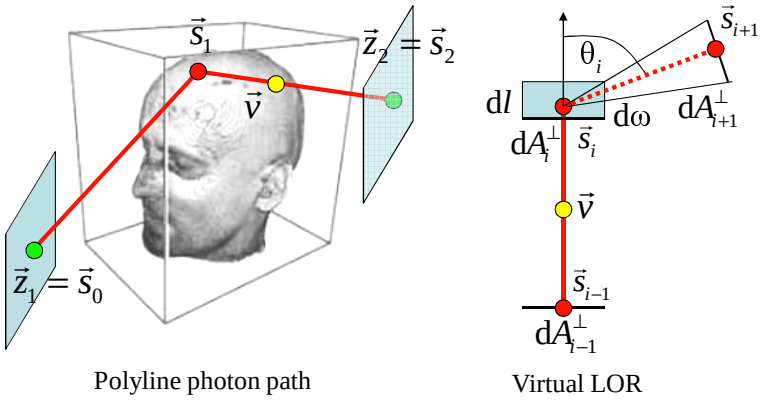


Figure 5.2: The scattered photon path is a polyline (left) made of virtual LORs (right). The left figure depicts the case of single scattering $S = 1$.

several factors:

$$C(\vec{s}_{i-1}, \vec{s}_i) = G(\vec{s}_{i-1}, \vec{s}_i) X(\vec{s}_{i-1}, \vec{s}_i) T_1(\vec{s}_{i-1}, \vec{s}_i) B_1(\vec{s}_{i-1}, \vec{s}_i),$$

where $G(\vec{s}_{i-1}, \vec{s}_i)$ is the *geometry factor* defined in Equation 4.3 having $\cos\theta_{\vec{z}_1} = \cos\theta_{\vec{z}_2} = 1$, $X(\vec{s}_{i-1}, \vec{s}_i)$ is the *total emission* along the line segment, $T_{\epsilon_0}(\vec{s}_{i-1}, \vec{s}_i)$ is the *total attenuation due to out-scattering*, and $B_{\epsilon_0}(\vec{s}_{i-1}, \vec{s}_i)$ is the *total attenuation due to photoelectric absorption*, assuming photon energy ϵ_0 (Equation 1.6):

$$G(\vec{s}_{i-1}, \vec{s}_i) = \frac{1}{|\vec{s}_{i-1} - \vec{s}_i|^2}, \qquad X(\vec{s}_{i-1}, \vec{s}_i) = \frac{1}{2\pi} \int_{\vec{s}_{i-1}}^{\vec{s}_i} x(\vec{l}) \mathrm{d}l,$$

$$T_{\epsilon_0}(\vec{s}_{i-1}, \vec{s}_i) = e^{-\int_{\vec{s}_{i-1}}^{\vec{s}_i} \sigma_s(\vec{l}, \epsilon_0) \mathrm{d}l}, \qquad B_{\epsilon_0}(\vec{s}_{i-1}, \vec{s}_i) = e^{-\int_{\vec{s}_{i-1}}^{\vec{s}_i} \sigma_a(\vec{l}, \epsilon_0) \mathrm{d}l}$$

In the line segment of the emission, the original photon energy has not changed yet, thus $\epsilon_0 = 1$.

The integral of the contributions of paths of S scattering points is the product of these factors. For example, the integral of the contribution of paths of one scattering point is [WNC96]

$$\tilde{y}_L^{\text{scatter}} \approx \tilde{y}_L^{(1)} = \int_{D_1} \int_{D_2} \int_{\mathcal{V}} \sigma_s(\vec{s}) P(\cos\theta, 1) \mathcal{P}(\vec{z}_1, \vec{s}, \vec{z}_2) \mathrm{d}s \mathrm{d}z_2 \mathrm{d}z_1 \tag{5.1}$$

where

$$\mathcal{P}(\vec{z}_1, \vec{s}, \vec{z}_2) = \mathcal{P}(\vec{z}_1, \vec{s}) + \mathcal{P}(\vec{s}, \vec{z}_2) =$$

$$\cos\theta_{\vec{z}_1}\cos\theta_{\vec{z}_2}\left(C(\vec{z}_1,\vec{s})G(\vec{s},\vec{z}_2)T_{\epsilon_0}(\vec{s},\vec{z}_2)B_{\epsilon_0}(\vec{s},\vec{z}_2)+C(\vec{s},\vec{z}_2)G(\vec{z}_1,\vec{s})T_{\epsilon_0}(\vec{z}_1,\vec{s})B_{\epsilon_0}(\vec{z}_1,\vec{s})\right)$$

is the total contribution of polyline $\vec{z}_1,\vec{s},\vec{z}_2$, consisting of the contributions $\mathcal{P}(\vec{z}_1,\vec{s})$, $\mathcal{P}(\vec{s},\vec{z}_2)$ of line segments $\vec{z}_1,\vec{s}$ and $\vec{s},\vec{z}_2$, respectively. Here $\theta_{\vec{z}_1}$ is the angle between the first detector's normal and the direction of $\vec{z}_1$ to $\vec{s}$, $\theta_{\vec{z}_2}$ is the angle between the second detector's normal and the direction of $\vec{z}_2$ to $\vec{s}$. The photon's energy level ϵ_0 is obtained from the *Compton formula* for scattering angle θ formed by directions $\vec{s}-\vec{z}_1$ and $\vec{z}_2-\vec{s}$. Probability $P(\cos\theta,1)$ that scattering in $\vec{s}$ happens at angle θ is obtained from the Klein-Nishina formula (Section 1.1.1).

5.1 Previous work on scatter estimations

Early approaches tried to measure the scattered contribution directly, either using multiple energy windows [GSJ+91, SFK94] or an auxiliary, septa-extended scan [CMH93, CWJ+05]. However, since the scattered photons cannot be perfectly separated from the direct hits by neither of these methods, this practically turns the scatter component of the statistical noise model of the ML-EM into deterministic noise. Nowadays, *model-based scatter correction* methods are more popular which estimate the number of scattered photons $\tilde{y}_L^{\text{scatter}}$ from a given annihilation density. As opposed to the first analytical models [BEB+83, SK91, BM94] that estimate the scatter as an integral transform empirically derived for water or a general anatomical model of the human body, model-based scatter correction methods are object-dependent as they take into account the transmission scans. Although there is a growing research interest to include Time of Flight (ToF) data to scatter models [WSK06, Wat07, IMS07], in the following we consider only methods that are applicable for a wider family of PET scanners, possibly without ToF support (such as the scanners of Section 1.1.2).

5.1.1 Out-of-FOV scattering

Scattered paths may reach regions that are outside of the field of view (FOV). Photons may born outside of the FOV and scattered inside or born inside the FOV, leaving it and then scatter back from the measured object or the gantry, neither of which can be modeled in a physically plausible manner due to the lack of accurate out-of-FOV emission and transmission data. Although transmission scans may have wider field of view than PET scans providing a bigger material volume than the PET FOV, transmission data of the entire gantry is rarely available. The traditional way to compensate for these effects is to scale the computed scattered contribution, the corresponding factor is calculated by comparing either the estimated direct component with the estimated scattered contributions in so-called "*tails*" of the sinogram [WNC96, WCMB04] corresponding to regions outside of the object, or the estimated direct component with the measurements [KY11b]. Note that this is independent of the actual scattering simulation. In the following, we assume that out-of-FOV effects are either negligible due to the scanner geometry or already modelled by one of the aforementioned methods.

5.1.2 Single scatter simulations

Ollinger et al. [OJ93] and Watson [WNC96] independently described an analytical model for the *single scattering simulations* (*SSS*), assuming that photoelectric absorption is negligible (i.e. $B_{\epsilon_0}(\vec{v}_1,\vec{v}_2)=1$) and considering only Compton scattering. The two approaches basically differ in the evaluation strategy of the model. Watson's method has become more popular, since it offers a greater ease of implementation and by evaluating the contribution of the two line segments of the scatter path together, it may also be more efficient. The algorithm approximates the volumetric integral over the scattering points of Equation 5.1 by taking N_{scatter} scattering point samples:

$$\tilde{y}_L^{(1)} \approx \sum_{\vec{s}} \sigma_s(\vec{s})P(\cos\theta,1)(\mathcal{P}(\vec{z}_1,\vec{s})+\mathcal{P}(\vec{s},\vec{z}_2)).$$

In practice, most of the computational capacity is spent on the line integrals of $\mathcal{P}(\vec{z}_1, \vec{s})$ and $\mathcal{P}(\vec{s}, \vec{z}_2)$. The naive approach [WNC96] evaluates these simultaneously for each LOR, computing $\mathcal{O}(N_{\text{LOR}})$ line integrals. Utilizing that the energy dependence of the integrals $T_{\epsilon_0}(\vec{a}, \vec{b})$ of the scattering cross section θ_s can be expressed as a simple scaling of the 511 keV integrals [Oll96], an improved version of this method [Wat00] requires only $\mathcal{O}(N_{\text{scatter}} N_{\text{Det}})$ ray marchings, where N_{Det} is the number of detector crystals. Figure 5.3 illustrates the steps of the algorithm. First, scattering points are selected in the volume of interest. Generally, this is done either randomly, discarding samples with a low scattering coefficient [Wat00], or on a uniform grid [KY11a]. Based on the observation that more scattering events happen in dense regions, Section 5.2 proposes the application of importance sampling in this phase. In the second phase, each detector crystal is connected to each of the scattering points, and along these line segments the line integrals of the activity and attenuation due to Compton scattering are computed, assuming 511keV photons (i.e. $\epsilon_0 = 1$). Existing implementations ignore photoelectric absorption, since it has a very low probability in soft tissues. However, in dense materials like bones or especially metal implants this assumption no longer holds. Thus, in our model, presented in Section 5.2, we consider photoelectric absorption as well. For performance reasons, existing (GPU-based) fully 3D implementations [BTD09, KY11a] down-sample [WBD+02] the set of detectors and include an additional LOR up-sampling pass. Scattering is assumed to be a low frequency phenomenon, so coarse sampling of the detectors is adequate. Similarly to the case of photoelectric absorption discussed above, this assumption is violated for dense materials. In Section 5.2 we show that an efficient GPU implementation allows to compute the line integrals between all detectors and scattering points in reasonable time. In the final phase, previously computed paths are *reused*. The line segments sharing a scattering point are paired, resulting in N_{scatter} polylines in each LOR. When a polyline is formed, the scattering angle and the Compton formula are evaluated, and the line integrals are corrected according to the ratios of the real photon energy and 511 keV. As a side effect of reusing scattering samples for every LOR, the approximation errors in different LORs are correlated, thus the reconstruction will be free of dot noise typical in other Monte Carlo (MC) algorithms.

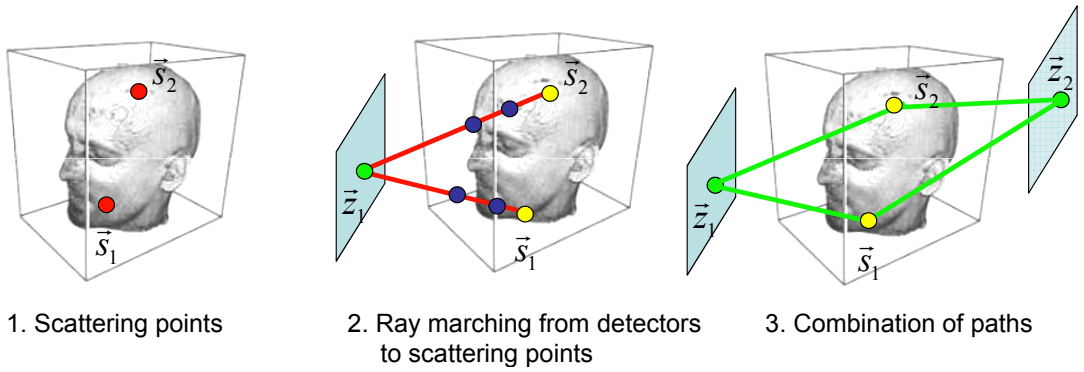


Figure 5.3: Steps of the single scatter simulation sampling process.

5.1.3 Multiple scatter models

Watson's method can be extended to include double scatter [TAR+05] or scattering of arbitrary order [J2] (Section 5.2.3), if we compute line integrals between scattering points. However, both the computational and code complexity grows rapidly with the number of allowed scattering events S. On the other hand, as the number of at most S-times scattered photons increases approximately as geometric series, simulating additional bounces has smaller and smaller impact on image quality and thus, it is worth using only a coarse but very fast approximation for

higher order scattering. For brain PET, single scatter comprise at least 75% of the scattering events [Oll96, TAR+05], while for a chest scan of obese people this ratio may reduce to 30%, therefore, the optimal point where we can safely truncate the number of allowed scattering events in the accurate model and use a fast approximation for additional bounces without compromising image quality mainly depends on the size of the subject and the scanner geometry.

Russian roulette [SK08], which stops particle paths randomly at interaction points [WCK+09], gives an unbiased estimator for the missing higher order scattering. However, Russian roulette always increases the variance of the estimator, thus it trades bias for noise. The other drawback of Russian roulette is that different paths have different length, which poses efficiency problems to SIMD like parallel hardware architectures like the GPU [B1, LSK10]. In his single scattering model, Watson [WNC96] compensated for multiple scatters together with out-of-FOV effects by scaling the single scatter component, which practically assumes that singly and multiply scattered events have the same spatial distribution up to a constant scaling factor. Goggin and Ollinger [GO94] approximated multiple scattering as the convolution of the single scatter distribution with a one-dimensional Gaussian kernel in LOR-space. The width of the Gaussian kernels was constant and determined by MC simulations, while its spatially varying amplitude was set according to the mean path length along the LOR. Later, this approach was extended to 3D PET and the benefits of smoothing the path length were shown [QMT10]. However, spatially varying filtering in LOR-space is still rather costly to model a phenomenon that has only a minor impact on the measurements.

Section 5.3 presents a simple approximate method to improve the accuracy of scatter computation in PET without increasing the computation time. We exploit the facts that higher order scattering is a low frequency phenomenon and the Compton effect is strongly forward scattering in 100–511 keV range. Analyzing the integrals of the particle transfer, we come to the conclusion that the directly not evaluated terms of the Neumann series can approximately be incorporated by the modification of the scattering cross section while the highest considered term is calculated, which has practically no overhead during the reconstruction. We note that recently, a similar approach was developed independently by Abhinav et al. [JKB+12] for optical imaging.

5.2 New improvements of the single scatter model

Watson's method [WNC96] is a popular choice of single scatter simulation and its implementation becomes very efficient with the reuse of line segments [Wat00]. Here we propose several improvements for this algorithm. First, in order to make the method suitable for dense materials, we show how to include photoelectric absorption into the model, without loosing the ability to pre-compute paths. Additionally, we propose the use of importance sampling for the selection of scattering samples. By giving an efficient GPU implementation that includes path reuse, we also show that the method can work in 3D without needing to downsample the detector space. Finally, we describe its generalization to arbitrary number of line segments.

5.2.1 Path reuse with photoelectric absorption

We consider the contribution of photon paths as an integral over the Cartesian product set of the volume. This integration domain is sampled globally, i.e. a single sample is used for the computation of all detector pairs. Sampling parts of photon paths globally and *reusing* a partial path for all detector pairs allow us to significantly reduce the number of samples.

When the attenuation is computed, we should take into account that the photon energy changes along the polyline and the scattering cross section also depends on this energy, thus different cross section values should be integrated when the annihilations on a different line segment are considered. As we wish to reuse the line segments and not to repeat ray-marching redundantly, each line segment is marched only once assuming photon energy $\epsilon_0 = 1$, and

attenuations T_1 and B_1 for this line segment are computed. Then, when the place of annihilation is taken into account and the real value of the photon energy ϵ_0 is obtained, initial attenuations T_1 [Oll96] and B_1 are transformed:

$$\sigma_s(\vec{l},\epsilon_0) = \sigma_s(\vec{l},1)\cdot\frac{\sigma_s^0(\epsilon_0)}{\sigma_s^0(1)}, \qquad \sigma_a(\vec{l},\epsilon_0) = \frac{\sigma_a(\vec{l},1)}{\epsilon_0^3}.$$

Using this relation, we can write

$$T_{\epsilon_0} = e^{-\int_{\vec{s}_{i-1}}^{\vec{s}_i}\sigma_s(\vec{l},\epsilon_0)\mathrm{d}l} = e^{-\frac{\sigma_s^0(\epsilon_0)}{\sigma_s^0(1)}\int_{\vec{s}_{i-1}}^{\vec{s}_i}\sigma_s(\vec{l},1)\mathrm{d}l} = T_1^{\frac{\sigma_s^0(\epsilon_0)}{\sigma_s^0(1)}}.$$

$$B_{\epsilon_0} = e^{-\int_{\vec{s}_{i-1}}^{\vec{s}_i}\sigma_a(\vec{l},\epsilon_0)\mathrm{d}l} = e^{-\frac{1}{\epsilon_0^3}\int_{\vec{s}_{i-1}}^{\vec{s}_i}\sigma_a(\vec{l},1)\mathrm{d}l} = B_1^{\frac{1}{\epsilon_0^3}}.$$

The energy dependence of the cross section $\sigma_s^0(\epsilon_0)$ is a scalar function, which can be pre-computed and stored in a table (Figure 5.4).

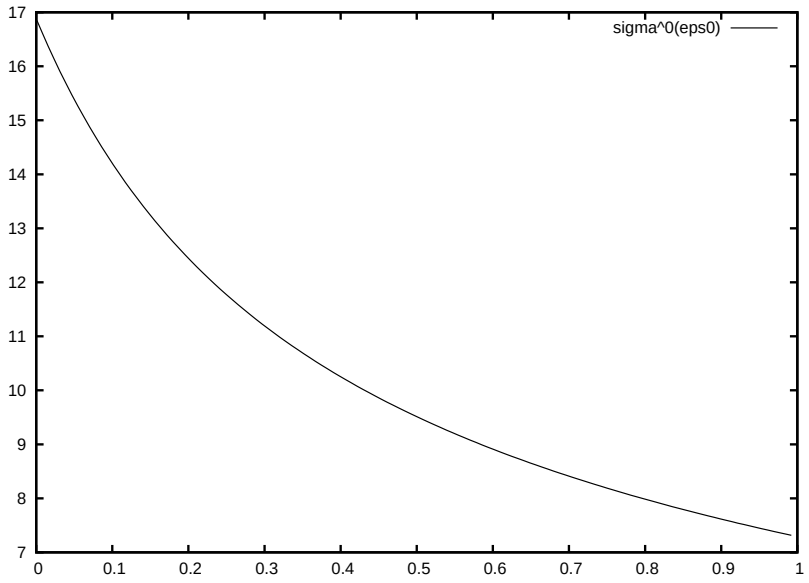


Figure 5.4: Normalized scattering cross section $\sigma_s^0(\epsilon_0)$.

5.2.2 Monte Carlo integration with importance sampling

Considering importance sampling, we should find a density p for scattering points that mimics the integrand of Equation 5.1. When inspecting the integrand, we should take into account that we evaluate a set of integrals (i.e. an integral for every LOR) using the same set of global samples, so the density should mimic the common factors of all these integrals.

The common factor is the electron density of the scattering points, so we mimic this function when sampling points. We store the scattering cross section at the energy level of the electron, $\sigma_s(\vec{v},1)$, which is proportional to the electron density. As the electron density function is provided by the CT reconstruction as a voxel grid, we, in fact, sample voxels. The probability density of sampling point $\vec{v}$ is:

$$p(\vec{v}) = \frac{\sigma_s(\vec{v},1)}{\int_{\mathcal{V}}\sigma_s(\vec{v},1)\mathrm{d}v} = \frac{\sigma_V}{\mathcal{K}}\frac{N_{\mathrm{voxel}}}{\mathcal{V}},$$

where σ_V is the scattering cross section at the energy level of the electron in voxel V, $\mathcal{K} = \sum_{i=1}^{N_{\text{voxel}}} \sigma_V$ is the sum of all voxels, and $\mathcal{V}$ is the volume of interest.

Note that this scheme ignores the annihilation intensity during the sampling of scattering points. However, especially when the sources are concentrated, it is worth increasing the density around the concentration since line segments ending in the sample points would probably cross the high activity region. To consider this, we can also take into account the current activity estimation in the sample density. So we look for the sample density in the following form

$$p(\vec{v}) \propto \sigma_s(\vec{v}, 1)(\alpha x(\vec{v}) + (1-\alpha)x_{ave}),$$

where $x_{ave} = \sum_{i=1}^{N_{\text{voxel}}} x_V / N_{\text{voxel}}$ is the average activity in the volume and α is a factor describing how strongly the activity affects the sampling density. If $\alpha = 0$, then we get back the previous case that mimics only the electron density, i.e. the scattering cross section. If $\alpha = 1$, then the density will be proportional to both the activity and the electron density, which may ignore zero activity parts where scattering may happen, so this extreme case is a biased estimator. Realistic α values must be in $[0, 1)$.

The proportionality ratio can be obtained by satisfying the constraint that the integral of a probability density must be 1:

$$p(\vec{v}) = \frac{\sigma_s(\vec{v}, 1)(\alpha x(\vec{v}) + (1-\alpha)x_{ave})}{\int_{\mathcal{V}} \sigma_s(\vec{v}, 1)(\alpha x(\vec{v}) + (1-\alpha)x_{ave})\mathrm{d}v} = \frac{\sigma_s(\vec{v}, 1)(\alpha x(\vec{v}) + (1-\alpha)x_{ave})}{\alpha \int_{\mathcal{V}} \sigma_s(\vec{v}, 1)x(\vec{v})\mathrm{d}v + (1-\alpha)x_{ave}) \int_{\mathcal{V}} \sigma_s(\vec{v}, 1)\mathrm{d}v} =$$

$$\frac{\sigma_V(\alpha x_V + (1-\alpha)x_{ave})}{\alpha\mathcal{S} + (1-\alpha)x_{ave}\mathcal{K}} \cdot \frac{N_{\text{voxel}}}{\mathcal{V}} \tag{5.2}$$

where $\mathcal{S} = \sum_{i=1}^{N_{\text{voxel}}} \sigma_V x_V$.

The single scattered contribution estimation with density $p(\vec{v})$ is the following:

$$\tilde{y}_L^{(1)} \approx \frac{D_1 D_2}{N_{\text{scatter}}} \frac{\alpha\mathcal{S} + (1-\alpha)x_{ave}\mathcal{K}}{\sigma_V(\alpha x_V + (1-\alpha)x_{ave})} \cdot \frac{\mathcal{V}}{N_{\text{voxel}}} \cdot \sum_{j=1}^{N_{\text{scatter}}} P_{KN}(\cos\theta_j, 1)\mathcal{P}_j$$

where θ_j is the scattering angle at $\vec{s}_j$, ϵ_0 is the energy level of the photon after this Compton scattering if originally it had the energy of the electron, and

$$\mathcal{P}_j = \mathcal{P}(\vec{z}_1, \vec{s}_j, \vec{z}_2)$$

is the total emission weighted by the attenuation of path $\vec{z}_1, \vec{s}_j, \vec{z}_2$.

5.2.3 Generalization to arbitrary number of bounces

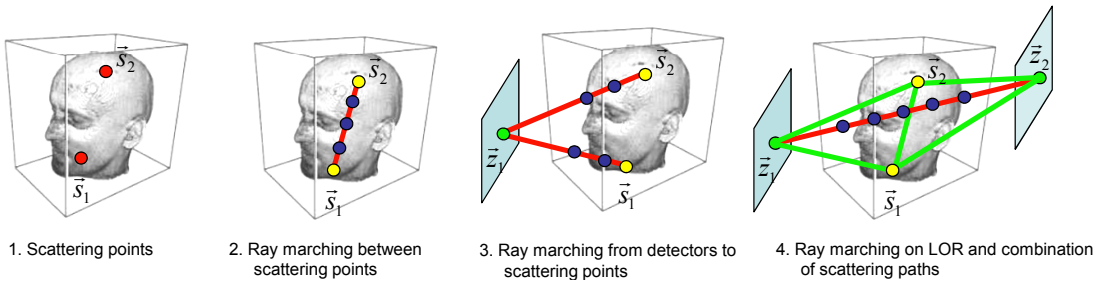


Figure 5.5: Steps of the multiple scatter simulation sampling process.

Scattered contribution is a sequence of increasing dimensional integrals, where the integrand is the contribution of a multi-bounce photon path. As the computation of a single segment of

such a path requires ray-marching and therefore is rather costly, we reuse the segments of a path in many other path samples. The basic steps of the path sampling process, considering at most S scattering points, are shown by Figure 5.5:

1. First, N_{scatter} scattering points $\vec{s}_1, \ldots, \vec{s}_{N_{\text{scatter}}}$ are sampled according to $p(\vec{v})$.

2. In the second step global paths are generated. If we decide to simulate paths of at most S scattering points, N_{path} ordered subsets of the scattering points are selected and paths of S points are established. If statistically independent random variables were used to sample the scattering points, then the first path may be formed by points $\vec{s}_1, \ldots, \vec{s}_S$, the second by $\vec{s}_{S+1}, \ldots, \vec{s}_{2S}$, etc. Each path contains $S-1$ line segments, which are marched assuming that the photon energy has not changed from the original electron energy. Note that building a path of length S, we also obtain many shorter paths as well. A path of length S can be considered as two different paths of length $S-1$ where one of the end points is removed. Taking another example, we get $S-1$ number of paths of length 1. Concerning the cost, rays should be marched only once, so the second step altogether marches on $N_{\text{path}}(S-1)$ rays.

3. In the third step, each detector is connected to each of the scattering points in a deterministic manner. Each detector is assigned to a computation thread, which marches along the connection rays. The total rays processed by the second step is $N_{\text{Det}}N_{\text{scatter}}$.

4. Finally, detector pairs are given to GPU threads that compute the direct contribution and combine the scattering paths ending up in them.

5.3 A new simplified multiple scattering model

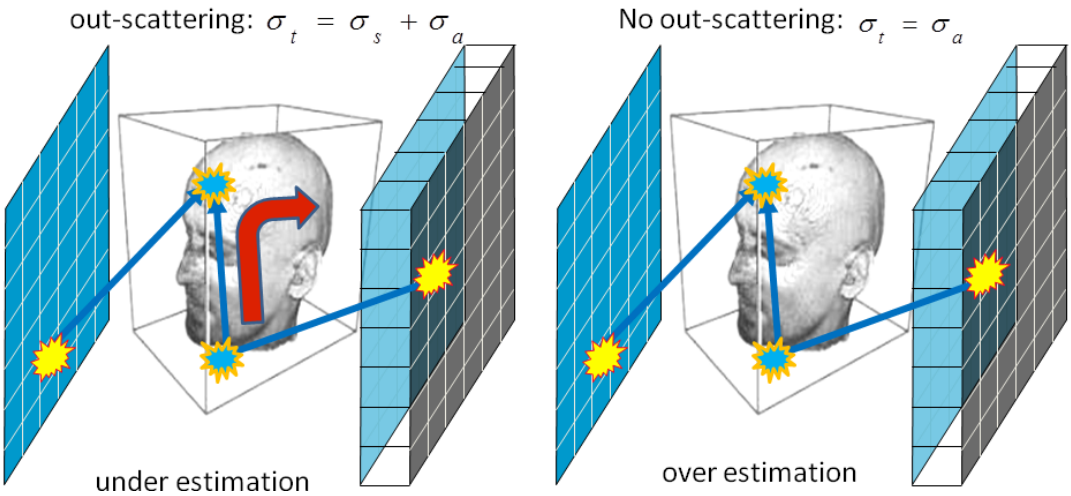


Figure 5.6: While computing the highest order term, replacing the extinction coefficient by the absorption cross section and by the sum of the absorption and scattering cross sections results in an overestimation and an underestimation, respectively.

The last really evaluated term of the Neumann series, which represents the highest bounce or longest paths, can be computed in two different ways (Figure 5.6):

- We use the same absorption and scattering cross section in the last term as in others, and ignore the higher terms of the Neumann series, which is an *underestimation* because we

loose out-scattered photons that may contribute to in-scattering of higher and therefore not computed terms.

- We ignore out-scattering while the last considered term is calculated since this would be the in-scattering of even higher order bounces, which are ignored, thus the energy balance is better maintained if out-scattering on this level is also ignored. This approach leads to a global *overestimation* because allowing neither out-scattering nor in-scattering corresponds to the assumption that scattered photons are never lost by the system, which is not true in reality. This approach replaces scattered paths by a shorter line segment, so absorption or dropping the energy below the energy window is less likely. If Compton scattering is considered, changing the photon direction results in an energy drop, which makes absorption even more likely, which even further increases the gap between the contribution of scattered and linear paths. By "global overestimation" we mean that ignoring scattering may decrease the contribution to detectors that can be reached mainly by scattered paths, but increases the linearly reachable path by a larger extent.

In this section we propose a simple trick to approximate the true value between the underestimation and overestimation. The approximation is based on the recognition that both the underestimating and the overestimating cases correspond to the modification of volume properties in Equation 1.3, and are members of a much wider family, which also includes cases in between the extreme ones. The two extreme approximations correspond to replacing the Klein-Nishina differential cross section by Dirac-delta $\delta(\vec{\omega} - \vec{\omega}_{\text{in}})$ scaled by zero or by the scattering cross section, respectively. In-betweening approximations can be obtained by additionally scaling of the Dirac-delta by parameter λ that is between 0 and 1, which leads to our simplified scattering model:

$$\frac{\mathrm{d}\sigma_s(\vec{l}, \vec{\omega}_{\text{in}} \cdot \vec{\omega}, \epsilon_{\text{in}})}{\mathrm{d}\omega_{\text{in}}} \approx \lambda \sigma_s(\vec{l}, \epsilon) \delta(\vec{\omega} - \vec{\omega}_{\text{in}})$$

where $\epsilon_{\text{in}} = \epsilon$ since the Compton effect does not change the photon energy when the direction is not altered. We emphasize that this simplified model is used only when the line integrals of the highest considered term are evaluated, in all other cases, the original Klein-Nishina formula is applied.

Substituting the simplified model into Equation 1.3, we obtain

$$\vec{\omega} \cdot \vec{\nabla} I(\vec{l}, \vec{\omega}, \epsilon) = -(\sigma_a(\vec{l}, \epsilon) + \sigma_s(\vec{l}, \epsilon)) I(\vec{l}, \vec{\omega}, \epsilon) + I^e(\vec{l}, \epsilon) + \lambda \sigma_s(\vec{l}, \epsilon) I(\vec{l}, \vec{\omega}, \epsilon).$$

The term coming from the in-scattering integral can be interpreted as the modification of the scattering cross section, so we get a pure differential equation that is similar to Equation 1.4 obtained by ignoring the in-scattering term:

$$\vec{\omega} \cdot \vec{\nabla} I(\vec{l}, \vec{\omega}, \epsilon) = -(\sigma_a(\vec{l}, \epsilon) + \sigma'_s(\vec{l}, \epsilon)) I(\vec{l}, \vec{\omega}, \epsilon) + I^e(\vec{l}, \epsilon),$$

where $\sigma'_s = (1 - \lambda)\sigma_s$. The solution of the differential equation can be expressed in the same form as Equation 1.5 having replaced scattering cross section σ_s by $(1 - \lambda)\sigma_s$. The accuracy of this approximation depends on the proper choice of λ and on how strongly the phase function is forward scattering and is similar to a Dirac-delta function. Intuitively, parameter λ expresses the probability that a photon scattered more than the limit caused by the truncation of the Neumann series gets lost for the system.

We have two options to find an appropriate λ parameter. Based on its probabilistic interpretation, the probability that a photon gets lost due to scattering more times than the considered limit can be determined by an off-line simulation with e.g. GATE [Jea04]. This probability depends on the tomograph geometry, the size of the the object and also on the maximum number of bounces, so several simulation studies are needed for different measurement protocols. The other option is simpler and is based on the geometric evaluation of the tomograph. The details of this approximation is discussed in the next subsection. As we shall demonstrate in

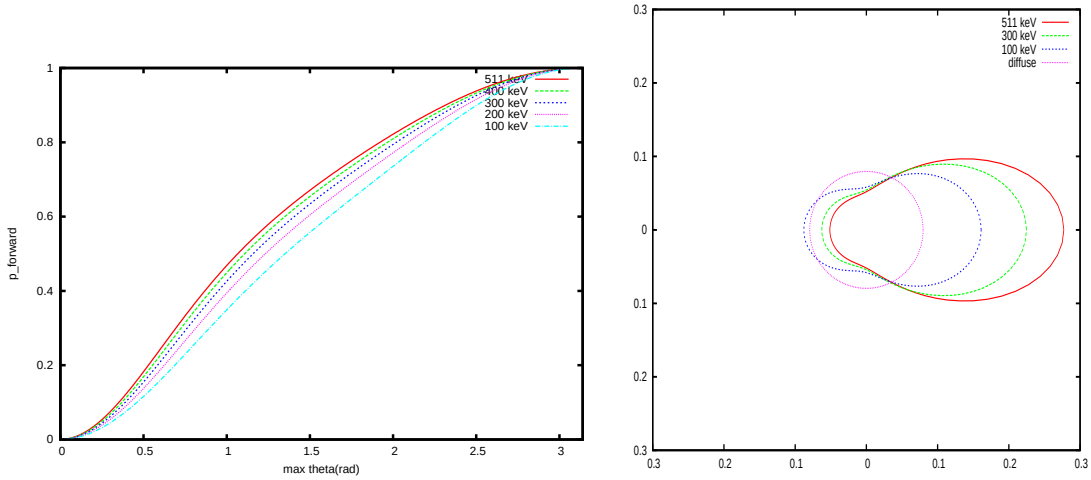


Figure 5.7: The forward scattering probability with respect to the allowed maximum scattering angle in radian (left) and the Klein-Nishina phase function assuming different incident photon energies (right). For comparison, we also show the phase function of isotropic diffuse materials.

the Results section, the reconstruction quality is not strongly sensitive to the exact choice of parameter λ, so the exact value is not important, and good results can be obtained with rough approximations of λ as well.

5.3.1 Determination of parameter λ

The modification of the scattering cross section during the evaluation of the line integrals of the highest simulated bounce requires parameter λ, which determines the probability of scattering when the direction is not significantly changed. The main reasons of the energy loss in a scattering only media is that the photon leaves the system without interacting with the detector ring and that the energy of the photon drops below the minimum value of the energy window (typically 100 keV) due to the Compton effect.

Back-scattering means lost photons since when the direction is reversed once, the photons of the annihilation pair arrive at detector modules that are not in coincidence, so this photon pair is ignored by the electronics. Multiple direction reverses are unlikely and reduce the photon energy significantly (one full back-scattering reduces the energy of an 511 keV photon to the third of its original energy and two full back-scattering events to the fifth), thus these photons will be ignored since their energy is outside of the energy window.

Considering forward scattering only, a conservative approach to λ would be the computation of the probability that the photon hits the same detector crystal after scattering as it would hit traveling a linear path. Clearly, such scattering events are not even recognized by the measurement system. This probability is equal to the integral of the phase function over the solid angle subtended by the detector crystal surface.

However, this conservative estimation ignores the fact that the scatter component already has low frequency characteristics, so the beam that is analyzed can be assumed to be wider. So, even if a photon changes its direction so significantly that it arrives at a different detector crystal, which results in a contribution drop for this LOR, we can assume that other paths parallel to the current one can be handled similarly, so their loss is a positive contribution in the considered LOR. In a wider homogeneous beam, changing photon directions compensates each other. So, the solid angle in which the phase function needs to be integrated can be wider, and can be increased up to the solid angle in which the detector modules being in coincidence relation can be seen. In Mediso's AnyScan PET/CT geometry (Section 1.1.2), the maximum perturbation angle of a line that ensures that the intersected modules will be the same is about 30–45 degrees (0.5–0.6 radians).

Figure 5.7 shows the integral of the phase function of Compton scattering in solid angles defined by the maximum scattering angle. Note that in the 0.5–0.6 radian range, the integral is between 0.2–0.3 and is roughly independent of the photon energy, thus the lambda parameter should be selected around this value.

5.3.2 Application in the scatter compensation for PET

The proposed method can be used together with zero (i.e. attenuation compensation), first, second, etc. order scattering compensation at no additional cost. When the accurate model is evaluated, e.g. with the method presented in Section 5.2, we limit the length of paths. The simplified forward scattering model should be used when the longest paths are computed, simply modifying the scattering cross section. We consider two reconstruction scenarios based on the definition of the maximum length:

- **Attenuation only** reconstruction completely ignores scattering and computes only the direct contribution, i.e. photon paths that connect crystals by line segments. In this extreme case, our proposed modification should work on the level of direct contribution computation, and the contribution of ignored scattered paths is approximately reintroduced by the modification of the scattering cross section while attenuation of the direct contribution is computed.
- **Single scatter compensation** can be further improved by including the energy of multiple scattering in the contribution of single scattered paths.

5.4 Results

The methods are demonstrated with simulating Mediso's AnyScan PET/CT system [Med10a] (Section 1.1.2) with GATE [Jea04] for the NEMA NU2-2007 human IQ phantom. For the AnyScan PET/CT, the optimal λ is 0.2–0.3 according to the geometric considerations of Section 5.3.1. The proposed methods are fully implemented on the GPU.

We consider the two discussed scenarios. In the first one, we extend an attenuation only approach that calculates no scattering to approximately consider scattering effects. In the second scenario, we start with a single scattering compensation algorithm (Section 5.2 with $S = 1$ and $N_{\text{scatter}} = 300$ scattering samples) and improve it to include multiple scattering effects.

Adding scattering compensation to attenuation compensation in PET

If we wish to approximately include single and multiple scattering without computing these factors, then the scattering cross section should be modified during the attenuation calculation as

$$\sigma_s' = \sigma_s(1 - \lambda)$$

where λ is a system parameter depending on the solid angle in which the detectors are seen.

Figure 5.8 (left) shows the L_2 error curves of the reconstruction with attenuation simulation while the classical approach is used and when the scattering cross section is modified. For comparison, we also included the error curves when single scattering is simulated, and when multiple scattering is also approximated by modifying the scattering cross section during single scatter simulation. The transverse slices of the reconstructed data are shown by right of Figure 5.8.

Observe that the simple modification of the scattering cross section during attenuation calculation significantly improved the L_2 error curves and also the reconstruction images. Both the error curves and images produced by the improved attenuation only method are similar to those of obtained with a single scatter compensation, however, we cannot reach that quality.

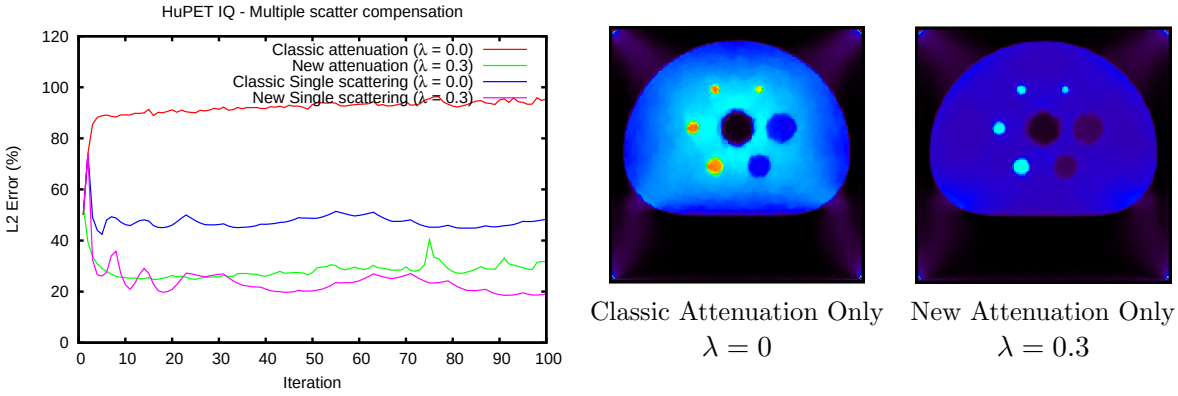


Figure 5.8: L_2 error curves (left) and transverse slices (right) of the reconstruction of the human IQ phantom while attenuation only simulation is executed in forward projection and also in back projection. In this phase, we control the scattering cross section with the λ parameter. Setting it to zero, we get the case of classical attenuation simulation back. Setting $\lambda = 0.3$ represents our new method. For comparison, we also included the L_2 curves when single scattering is explicitly calculated.

Adding multiple scattering compensation to single scattering methods

Here we present our results for the simplified forward scattering model, when included in the single scatter compensation of Section 5.2.

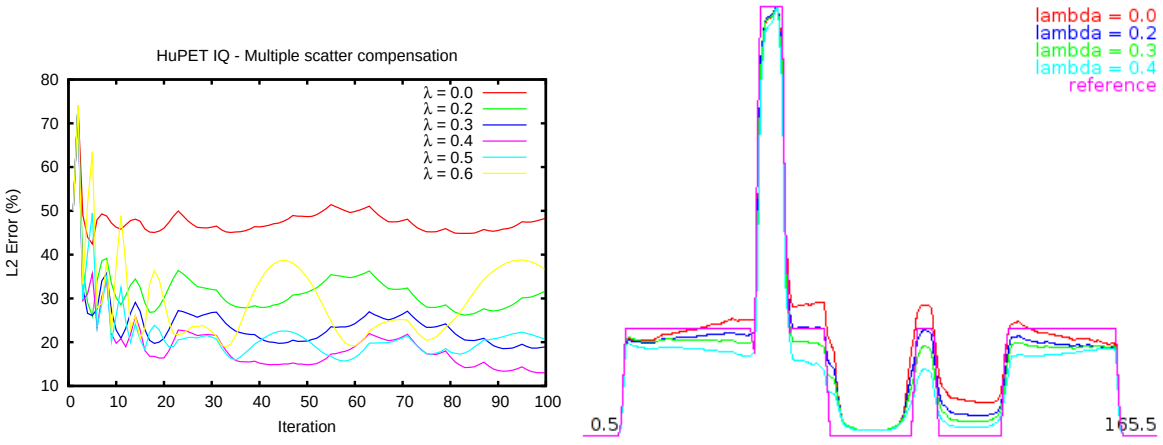
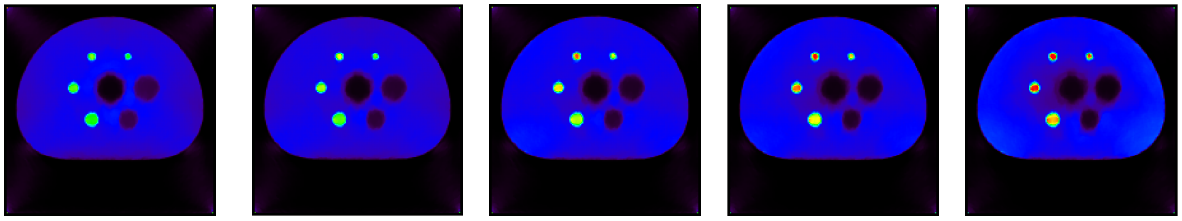


Figure 5.9: L_2 error curves (left) and line profiles (right) of the Single scatter compensation reconstruction of the IQ phantom while the λ parameter is selected in the 0.2–0.6 range. The wavy error curves are due to the fact that scattering is re-calculated just in every third iteration step to reduce computation time.

Figure 5.10 shows the transaxial slices of the reconstruction of the human IQ phantom, Figure 5.9 depicts L_2 error curves (left) and line profiles (right), and Figure 5.11 and Figure 5.12 the NEMA evaluation. When λ is small, the results are similar to that of classical single scatter compensation where the ignored energy causes an overestimation of the activity in the center of the phantom. Increasing λ this artifact disappears and the homogeneous region is reconstructed with constant activity. However, increasing λ beyond the reasonable range, the missing energy will be overcompensated, which shows up as a decreased reconstructed activity around the spine when $\lambda \geq 0.4$. The method is quite robust to the particular selection of λ, an arbitrary value in



Classic Single Scatter | Forward Scatter $\lambda = 0.2$ | Forward Scatter $\lambda = 0.3$ | Forward Scatter $\lambda = 0.4$ | Forward Scatter $\lambda = 0.5$

Figure 5.10: Transverse slices and line profiles of single scatter compensation reconstruction of the human IQ phantom with different λ parameters.

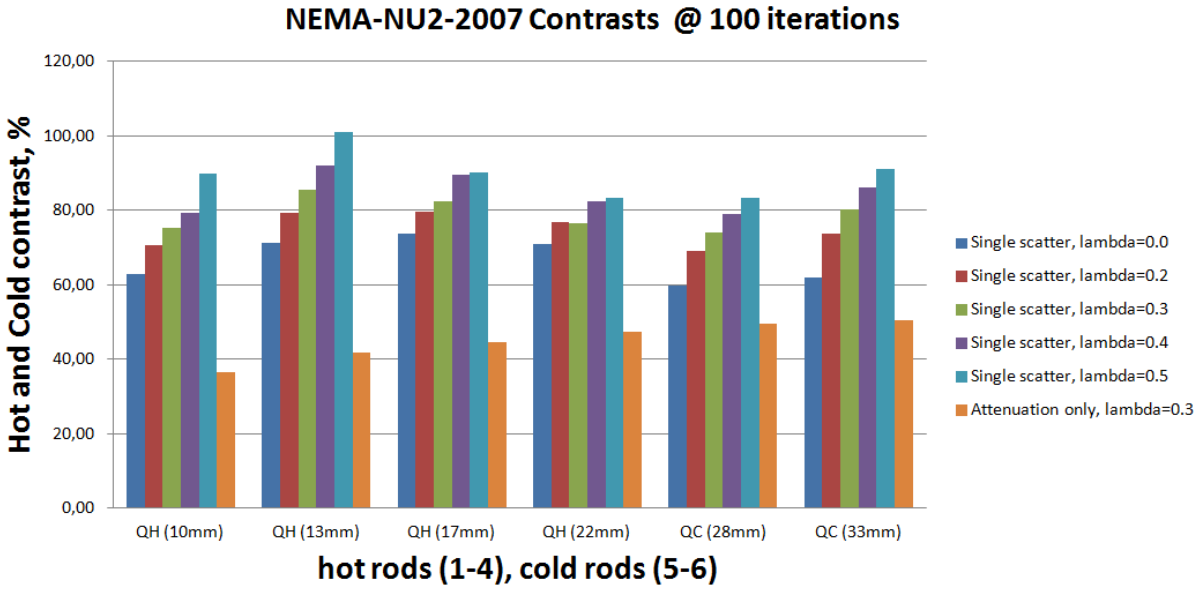


Figure 5.11: Contrast factors of the IQ phantom according to the NEMA standard. Note that increasing parameter λ, contrasts also increase.

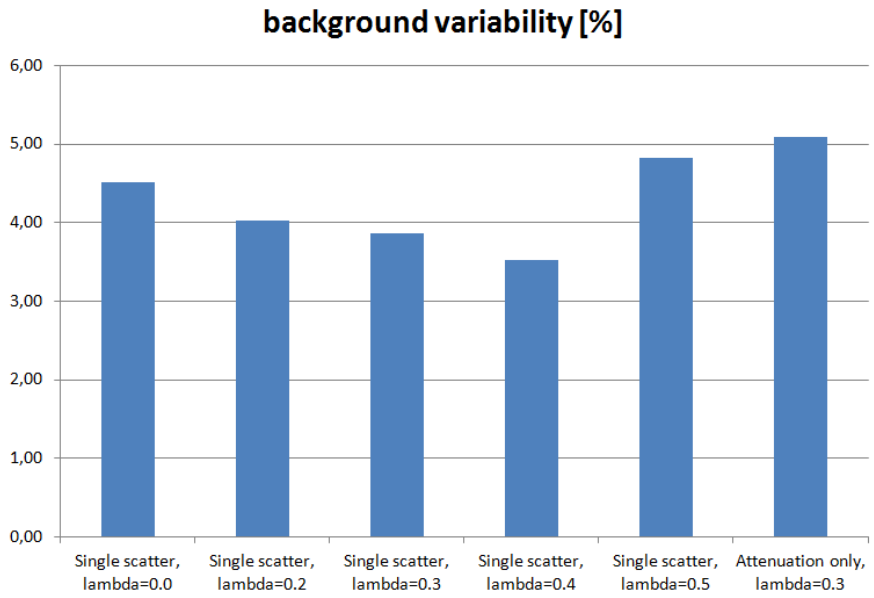


Figure 5.12: Background variability of the IQ phantom according to the NEMA standard. Note that increasing parameter λ, the background variability first improves then starts degrading.

$[0.2, 0.3]$ would improve the results without this artifact.

Examining the NEMA evaluation, we can observe that contrasts grow with higher λ, and background variability first decrease that start to rapidly increase if $\lambda > 0.4$. The reason is overcompensation, which means that the center of the phantom gets darker than expected.

5.5 Conclusions

This chapter proposed a GPU based scatter compensation algorithm for the reconstruction of PET measurements that accurately computes lower order scattering up to an arbitrary number of scattering points and gives a free approximation for the missing bounces. The resulting approach can reduce the computation time of the fully 3D PET reconstruction to a few minutes.

The scattering simulation approach is restructured to exploit the massively parallel nature of GPUs. Based on the recognition that the requirements of the GPU prefer a detector oriented viewpoint, we solve the adjoint problem, i.e. originate photon paths in the detectors. The detector oriented viewpoint also allows us to reuse samples, that is, we compute many annihilation events with tracing a few line segments.

The proposed method to take into account higher scattering in PET reconstruction corrects the scattering cross section when the highest computed bounce is evaluated, approximately including the contribution of the bounces above this level. As having modified the scattering cross section, the algorithm remains the same, the proposed correction has no computational overhead. We demonstrated the proposed correction with two scenarios. In the first one, no scattering is directly computed, only the attenuation calculation is modified according to the new proposal. In the second scenario, we improved the accuracy of a single scatter compensation method by approximately include multiple scattering contribution. The correction is based on a single parameter that describes the probability that a photon gets lost for the total system because of scattering. We discussed a simple approximation to obtain this parameter and showed that reconstruction quality can be increased even with roughly approximated parameter values.

Chapter 6

Detector model

Photons may scatter not only in the measured object but also in detector crystals. The average free path length of a 511 keV gamma photon in LYSO, which is a typical detector material is about 13 mm, which is an order of magnitude larger than the size of the crystal in small animal PETs and about three times bigger than the edge length of the detector crystals in a human PET. Thus, detector modeling is a must in small animal PET and also improves the reconstruction in human PET.

This chapter presents a fast algorithm to simulate inter-crystal scattering to increase the accuracy of PET. Taking advantage of the fact that the structure of the detector is fixed, we show how most of the corresponding scattering calculation can be ported to a pre-processing phase. Pre-computing the scattering probabilities inside the crystals, the final system response is the convolution of the geometric response obtained with the assumption that crystals are ideal absorbers and the crystal transport probability matrix. This convolution is four-dimensional which poses complexity problems as the complexity of the naive convolution evaluation grows exponentially with the dimension of the domain. We use Monte Carlo (MC) method to attack the curse of dimension. We demonstrate that these techniques have just negligible overhead.

Crystal transport probabilities are given by the (precomputed or measured) transport function $E_t(\vec{z}, \vec{\omega}, \epsilon_0 \to \mathbf{d})$ (Section 1.1.1). Ignoring scattering in the measured object and incorporating E_t into the expected value formula of a LOR $\tilde{y}_L$, we get the following model:

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} = \int_{\mathcal{D}} \int_{\mathcal{D}} G(\vec{z}_1, \vec{z}_2) X(\vec{z}_1, \vec{z}_2) A(\vec{z}_1, \vec{z}_2) E_t(\vec{z}_1, \vec{\omega} \to \mathbf{d}_1) E_t(\vec{z}_2, \vec{\omega} \to \mathbf{d}_2) \mathrm{d}z_2 \mathrm{d}z_1 \tag{6.1}$$

where $\vec{\omega}$ is the direction vector pointing to $\vec{z}_2$ from $\vec{z}_1$.

Note that in theory we should integrate over the total surface $\mathcal{D}$ of all detector modules. Of course, in practice, only the closer neighborhood of the detector is important. When this integral is evaluated, the incident direction of the photon is defined by the line of $\vec{z}_1$ and $\vec{z}_2$. If scattering in the measured object is ignored, then all incident photons have the same energy ($\epsilon_0 = 511$ keV). Thus, we shall omit the photon energy in all subsequent equations. The transport function is defined for points and directions, and needs to be represented by finite data. Sophisticated finite-element techniques would require too much memory, so we use a simpler discrete, piecewise constant approximate scheme.

In order to reduce the data needed to model detectors, we factorize the model and store only averages for different surface points per detector crystal. The factorization separates the crystal sensitivity and the probability of inter-crystal photon transport. We assign *detector sensitivity* $\mu_{\mathbf{d}}$ to each crystal $\mathbf{d}$, which is the expected number of events reported in this detector by the output of the measuring system, provided that a photon has been absorbed here:

$$\mu_{\mathbf{d}} = E[\text{number of events} \mid \text{photon has been absorbed in this detector}].$$

This value represents the specific properties of this crystal, like gamma sensitivity and response of the associated electronics, and is typically different from crystal to crystal.

The photon transport between the crystals is represented by a *crystal transport probability*:

$$p_{\mathbf{i}\to\mathbf{d}}(\vec{\omega}) = P[\text{absorbed in crystal } \mathbf{d} \mid \text{photon arrived at crystal } \mathbf{i} \text{ from direction } \vec{\omega}].$$

We assume first that the detector modules are infinitely large (later this assumption will be lifted to make the model realistic) and crystals are similar, thus this probability depends just on the translation between crystal $\mathbf{i}$ and crystal $\mathbf{d}$:

$$p_{\mathbf{i}\to\mathbf{d}}(\vec{\omega}) = p(\mathbf{d}-\mathbf{i}, \vec{\omega}).$$

It is enough to specify the probabilities supposing that the photon has arrived in a given crystal $\mathbf{i}^*$, which is supposed to be in the origin of a coordinate system. The measurements for a specific $\vec{\omega}$ are given in the form of *crystal transport probability function* $p_{\mathbf{i}^*\to\mathbf{d}}(\vec{\omega})$ for different crystals $\mathbf{d}$ assuming the input crystal to be in origin $\mathbf{i}^*$. If we are interested in the response of crystal $\mathbf{i}$ other than $\mathbf{i}^*$, then the whole system is translated by vector $\mathbf{i}-\mathbf{i}^*$.

Additionally, we suppose that the crystals are small with respect to the distance of the detector modules, so direction $\vec{\omega}$ of the LOR is constant for those detectors which are in the neighborhood of $\mathbf{d}$ and where $p_{\mathbf{i}\to\mathbf{d}}$ is not negligible.

The sum of the crystal transport probabilities is the *detection probability*, i.e. the probability that the photon does not get lost, or from a different point of view, does not leave the module without absorption:

$$\nu(\vec{\omega}) = \sum_{\mathbf{d}} p_{\mathbf{i}^*\to\mathbf{d}}(\vec{\omega}).$$

As transport probability $p_{\mathbf{i}^*\to\mathbf{d}}$ depends only on the translation between $\mathbf{i}^*$ and $\mathbf{d}$, the same sum can also be obtained running the input crystal index $\mathbf{i}$:

$$\nu(\vec{\omega}) = \sum_{\mathbf{i}} p_{\mathbf{i}\to\mathbf{d}}(\vec{\omega}).$$

Sum ν depends also on the orientation of the module of interest since it is determined by the rotations of the module and incident direction ω.

We consider a LOR connecting crystals $\mathbf{d}_1$ and $\mathbf{d}_2$. The relation between the discretized model and the original one is a simple averaging:

$$\frac{1}{D_{\mathbf{i}}}\int_{D_{\mathbf{i}}} E_t(\vec{z}, \vec{\omega}\to\mathbf{d})\mathrm{d}z = p_{\mathbf{i}\to\mathbf{d}}(\vec{\omega})\cdot\mu_{\mathbf{d}}$$

where $D_{\mathbf{i}}$ is the surface of the detector. The expected LOR value $y^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)}$ of Equation 6.1 is approximated as a convolution:

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} = \sum_{\mathbf{i}}\sum_{\mathbf{j}}\int_{D_{\mathbf{i}}}\int_{D_{\mathbf{j}}} G(\vec{z}_1,\vec{z}_2)X(\vec{z}_1,\vec{z}_2)A(\vec{z}_1,\vec{z}_2)E_t(\vec{z}_1,\vec{\omega}\to\mathbf{d}_1)E_t(\vec{z}_2,\vec{\omega}\to\mathbf{d}_2)\mathrm{d}z_2\mathrm{d}z_1 \approx$$

$$\sum_{\mathbf{i}}\sum_{\mathbf{j}}\int_{D_{\mathbf{i}}}\int_{D_{\mathbf{j}}} G(\vec{z}_1,\vec{z}_2)X(\vec{z}_1,\vec{z}_2)A(\vec{z}_1,\vec{z}_2)\mathrm{d}z_2\mathrm{d}z_1\cdot p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}_{\mathbf{i},\mathbf{j}})\cdot\mu_{\mathbf{d}_1}\cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega}_{\mathbf{i},\mathbf{j}})\cdot\mu_{\mathbf{d}_2} =$$

$$\sum_{\mathbf{i}}\sum_{\mathbf{j}}\tilde{y}^{geom}_{L(\mathbf{i},\mathbf{j})}\cdot p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}_{\mathbf{i},\mathbf{j}})\cdot\mu_{\mathbf{d}_1}\cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega}_{\mathbf{i},\mathbf{j}})\cdot\mu_{\mathbf{d}_2}. \tag{6.2}$$

Note that direction vector $\vec{\omega}_{\mathbf{i},\mathbf{j}}$ depends on the two crystals $\mathbf{i}$ and $\mathbf{j}$ whose surfaces the photons enter. However, if the photons cannot scatter far in the crystal, we can assume that the directions are similar and are equal to direction $\vec{\omega}$ between absorber crystals $\mathbf{d}_1$ and $\mathbf{d}_2$. With this simplification, we obtain the following convolution-like expression:

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} \approx \mu_{\mathbf{d}_1}\mu_{\mathbf{d}_2}\mathcal{C}(\tilde{y}^{geom}, \mathbf{d}_1, \mathbf{d}_2) \tag{6.3}$$

where the *convolution operation* for arbitrary $f(\mathbf{i},\mathbf{j})$ is:

$$\mathcal{C}(f,\mathbf{d}_1,\mathbf{d}_2)=\sum_{\mathbf{i}}\sum_{\mathbf{j}} f(\mathbf{i},\mathbf{j})\cdot p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega})\cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega})$$

where $\vec{\omega}$ is the direction between detector crystals $\mathbf{d}_1$ and $\mathbf{d}_2$.

So far, we have assumed that the detector modules are infinitely large, i.e. there are no edges. To handle the finite module geometry, let us add "virtual" detectors beyond the edges, but assume that these virtual detectors never get photons, that is, $\tilde{y}^{geom}_{L(\mathbf{i},\mathbf{j})}$ is constant zero if either $\mathbf{i}$ or $\mathbf{j}$ is a virtual detector. Due to this assumption, the "virtual detectors" do not alter the estimator, but allow us to use the same formula as for the infinite case. Practically, it means that we generate offsets with exactly the same algorithm close to the edge as inside the module, but the line integral between the points is set to zero if any of the offseted points is outside the module.

6.1 Previous work on detector modeling

In theory, the transport function $E_t(\vec{z},\vec{\omega},\epsilon_0\to\mathbf{d})$ depends on both the energy and incident angle of the incoming photon. However, as discussed later in Section 6.3.4, a physically plausible model would make factorization impossible, greatly increasing the complexity of the methods. If the scattering in the measured medium is ignored, which is a good estimation for pre-clinical systems [JSC+97], all incident photons have the same energy (511 keV). Thus, most methods, including the one that we shall present in this chapter, are derived for 511 keV photon energy. Another approach is followed by Stute et al. [SBM+11], they pre-compute E_t using MC simulation as the average over the energy distribution obtained with a simulated patient, however, they ignore the fact that even the average distribution depends on the position of the crystal w.r.t. the patient.

Simpler detector models neglect inter-crystal scattering and consider only the penetration of photons into the detectors which leads to minor modifications of the geometric projection of Equation 4.4. Absorption of the photons happen inside the detector crystals and if scattering is ignored then the attenuation of a gamma ray within a particular crystal is expressed by the the absorption coefficient σ_{att} and the length l of the linear path. This can be expressed as volumetric integrals within the detectors, weighted by the attenuation:

$$\tilde{y}^{att}_{L(\mathbf{d}_1,\mathbf{d}_2)}=\int_{V_1}\int_{V_2} G(\vec{z}_1,\vec{z}_2)X(\vec{z}_1,\vec{z}_2)A(\vec{z}_1,\vec{z}_2)p_{att}(\vec{z_1},\vec{\omega}_{\vec{z_2}\to\vec{z_1}})\cdot p_{att}(\vec{z_2},\vec{\omega}_{\vec{z_1}\to\vec{z_2}})\mathrm{d}z_2\mathrm{d}z_1 \qquad (6.4)$$

where V_1 and V_2 are the volumes of the two detectors,

$$G(\vec{z}_1,\vec{z}_2)=\frac{1}{|\vec{z_1}-\vec{z_2}|^2}$$

is the *volumetric geometry factor* and

$$p_{att}(\vec{z},\vec{\omega})=\sigma_{att}e^{-\sigma_{att}l}$$

is the *absorption probability density* function specifying the probability density that the photon coming from direction $\vec{\omega}$ is absorbed in point $\vec{z}$. A rather crude approximation of Equation 6.4 is to neglect attenuation and shift the line endpoints $\vec{z}_1$, $\vec{z}_2$ of Equation 4.4 from the detector surface into the crystals uniformly by the average depth of γ-photon interactions [MDB+08], i.e. to increase the radius of the scanner, called the *effective radius model* [C8]. Equation 6.4 may be solved numerically with Gaussian [MDB+08] or MC quadrature [C8], while analytic approaches also exist [SPC+00, SSD+03, PL11].

It has been demonstrated that the inclusion of inter-crystal scattering into the model results in an increase of image quality [DFF+07, C8]. We note, however, that some existing scanners [PL09, CLBT+12] allow to record individual photon–detector interactions independently which makes the effect of inter-crystal scattering less significant [PL11]. In most devices, including the ones we tested (Section 1.1.2), this feature is not available so the effect has to be modeled. A common technique is to model the transport function $E_t(\vec{z}, \vec{\omega}, \epsilon_0 \rightarrow \mathbf{d})$ as a 4D *blurring operator* in LOR-space [MDB+08, RLT+08, SBM+11, C8], which was described in the chapter introduction. The shape of the kernels strongly depend on the incident angle and thus methods that pre-compute E_t by averaging over the set of possible angles [RLT+08, SBM+11] introduce significant error [RLT+08]. If scattering in the subject is ignored, the incident direction is determined by the two endpoints of the LOR — which was followed by the introduction. Hence, angle dependent kernels, generated off-line for a set of incident angles, can be integrated into the model [MDB+08, C8] (also used by Section 6.3). We note that even if scattering in the subject is considered, this approximation is still fairly accurate. Since scattering is a low-frequency phenomenon and in the energy range of PET photons are more likely to scatter forward [ZM07, C9], the representative direction corresponding to direct hits is close to the mean incident direction. Additionally, since the probability of photoelectric interaction in the detectors increases with decreasing energy of the photons, scattered photons are less likely to leave the incident crystal [OJ93].

The support of the blurring kernels may be more than ten crystals wide in each direction of the two detectors that make up the LOR. Thus, the evaluation of Equation 6.2, i.e. computing a 4D discrete convolution in the form of

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} \approx \sum_{\mathbf{i}} \sum_{\mathbf{j}} \tilde{y}^{geom}_{L(\mathbf{i},\mathbf{j})} \cdot w_{\mathbf{i},\mathbf{j}}(\vec{\omega}_{\mathbf{i},\mathbf{j}}),$$

with $w_{\mathbf{i},\mathbf{j}}(\vec{\omega}_{\mathbf{i},\mathbf{j}})$ representing the kernel weights, is challenging, even for off-line models [MDB+08]. For larger detector sizes, the majority of the photons stay inside a small neighborhood and thus kernels may be cut in order to reduce their dimensions to a manageable size [SBM+11]. Unfortunately, for detector sizes close to a millimeter — such as the case for Mediso's *nanoScan-PET/CT* [Med10b] — this would not help much, the shrunk kernels are still huge to evaluate the convolution using traditional quadrature rules. As proposed in the following, MC sampling can significantly reduce the computation cost making it feasible for on-the-fly evaluation even for large kernel supports. Another way to gain a significant speed up is to factorize the geometric projection from the detector model. Earlier methods [SBM+11, C8] overlooked the fact that the direct contribution of a LOR $\tilde{y}^{geom}_L$ is present in many convolutions evaluating the detector model $\tilde{y}^{det}_{L'}$ for neighbouring LORs and compute it on-the-fly several times, which is redundant. The method presented in Section 6.3 first computes and stores the geometric projection, while in the second phase of evaluating the convolution in LOR-space, the geometric contribution is read from the memory of the GPU.

6.2 LOR-space filtering

The set of LORs is a 4D data, which means that the blurring caused by inter-crystal scattering may be expressed as a 4D filtering on the LORs. The general form of a *spatial-invariant filter* is:

$$\tilde{L}(\mathbf{r}) = \int L(\mathbf{r} - \mathbf{s}) w(\mathbf{s}) \mathrm{d}s,$$

where $\tilde{L}(\mathbf{r})$ is the filtered value at location $\mathbf{r}$, $L(\mathbf{r})$ is the original signal, and $w(\mathbf{s})$ is the *filter kernel* for this point. The domain of integration is 2D in image processing, but can be arbitrary in other applications. MC quadrature can significantly decrease computation time of filtering, especially for higher dimensions.

For the sake of notational simplicity, we introduce the method in one-dimension, but the generalization to arbitrary dimensions is also straightforward. Let us consider the

$$\tilde{L}(X) = \int_{-\infty}^{\infty} L(X-x)w(x)\mathrm{d}x$$

one-dimensional convolution, and find integral $\tau(x)$ of the kernel and also its inverse $x(\tau)$ so that the following conditions hold

$$\frac{\mathrm{d}\tau}{\mathrm{d}x} = w(x) \quad \text{i.e.} \quad \tau(x) = \int_{-\infty}^{x} w(t)\mathrm{d}t.$$

If kernel $w(t)$ is a probability density, i.e. it is non-negative and integrates to 1, then $\tau(x)$ is non-decreasing, $\tau(-\infty) = 0$, and $\tau(\infty) = 1$. In fact, $\tau(x)$ is the cumulative probability distribution function of the probability density. If filter kernel w is known, then $x(\tau)$ can be computed and inverted off-line for sufficient number of uniformly distributed sample points. Substituting the $x(\tau)$ function into the filtering integral we obtain

$$\tilde{L}(X) = \int_{-\infty}^{\infty} L(X-x)w(x)\mathrm{d}x = \int_{-\infty}^{\infty} L(X-x)\frac{\mathrm{d}\tau}{\mathrm{d}x}\mathrm{d}x = \int_{0}^{1} L(X-x(\tau))\mathrm{d}\tau.$$

Approximating the transformed integral taking uniformly distributed samples in τ corresponds to a quadrature of the original integral taking M non-uniform samples in x. This way we take samples densely where the filter kernel is large and fetch samples less often farther away, but do not apply weighting.

6.3 Proposed detector modeling using LOR-space MC filtering

According to Equation 6.2, the final expected number of hits is given by a long weighted sum of the expected number of events between the neighboring crystals, i.e. the LOR value obtained with the geometric model. Note that this is similar to image filtering, but now the space is not 2D but 4D (see Section 6.2). The long sum is evaluated by MC estimation taking N_{det} random samples of detector pairs $(\mathbf{i}(1),\mathbf{j}(1)), (\mathbf{i}(2),\mathbf{j}(2)), \ldots, (\mathbf{i}(N_{\mathrm{det}}),\mathbf{j}(N_{\mathrm{det}}))$:

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} \approx \mu_{\mathbf{d}_1}\mu_{\mathbf{d}_2}\mathcal{C}(\tilde{y}^{\mathrm{geom}},\mathbf{d}_1,\mathbf{d}_2) = \sum_{\mathbf{i}}\sum_{\mathbf{j}} \tilde{y}^{geom}_{L(\mathbf{i},\mathbf{j})} \cdot p_{\mathbf{i}\to\mathbf{d}_1} \cdot \mu_{\mathbf{d}_1} \cdot p_{\mathbf{j}\to\mathbf{d}_2} \cdot \mu_{\mathbf{d}_2} \approx$$

$$\frac{1}{N_{\mathrm{det}}} \cdot \sum_{s=1}^{N_{\mathrm{det}}} \frac{\tilde{y}^{geom}_{L(\mathbf{i}(s),\mathbf{j}(s))} \cdot p_{\mathbf{i}(s)\to\mathbf{d}_1} \cdot \mu_{\mathbf{d}_1} \cdot p_{\mathbf{j}(s)\to\mathbf{d}_2} \cdot \mu_{\mathbf{d}_2}}{p_s}$$

where p_s is the probability of sample s. A sample is associated with a pair of offset vectors $\mathbf{d}_1 - \mathbf{i}$ from $\mathbf{d}_2 - \mathbf{j}$. According to *importance sampling* [Chr03], p_s is made proportional to the crystal transport probability:

$$p_s = \frac{p_{\mathbf{i}(s)\to\mathbf{d}_1} \cdot p_{\mathbf{j}(s)\to\mathbf{d}_2}}{\sum_{\mathbf{i}}\sum_{\mathbf{j}} p_{\mathbf{i}\to\mathbf{d}_1} \cdot p_{\mathbf{j}\to\mathbf{d}_2}} = \frac{p_{\mathbf{i}(s)\to\mathbf{d}_1} \cdot p_{\mathbf{j}(s)\to\mathbf{d}_2}}{\nu_1(\vec{\omega}) \cdot \nu_2(\vec{\omega})}.$$

Thus, the final estimator is:

$$\tilde{y}^{det}_{L(\mathbf{d}_1,\mathbf{d}_2)} \approx \frac{\nu_1(\vec{\omega}) \cdot \nu_2(\vec{\omega}) \cdot \mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}}{N_{\mathrm{det}}} \cdot \sum_{s=1}^{N_{\mathrm{det}}} \tilde{y}^{geom}_{L(\mathbf{i}(s),\mathbf{j}(s))}. \tag{6.5}$$

This method runs a geometric first pass, which is the same algorithm as developed to execute the forward projection of the geometric reconstruction. This pass results in LOR values $\tilde{y}^{geom}_{L}$. Then, the 4D LOR map is filtered. We visit again each LOR, find neighbors of its two crystals according to a prepared random map, and add up the values stored in the LOR selected by the two sampled neighbors.

6.3.1 Detector modeling in back projection

In back projection the system matrix is simplified and we ignore blurring effects like positron range and scattering. Using piece-wise constant approximation of the transport function we get:

$$\mathbf{A}_{L(\mathbf{d}_1,\mathbf{d}_2),V} \approx \sum_{\mathbf{i}} \sum_{\mathbf{j}} \mathbf{D}_{L(\mathbf{i},\mathbf{j}),V} \cdot p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_1} \cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_2}$$

where $\mathbf{D}_{LV}$ is the system matrix simulating geometric effects and attenuation.

In the back projection of ML-EM reconstruction, we have to evaluate the numerator and the denominator of the scaling factor for each voxel. The numerator is

$$\sum_L A_{LV} \cdot \frac{y_L}{\tilde{y}_L} = \sum_{\mathbf{d}_1} \sum_{\mathbf{d}_2} A_{L(\mathbf{d}_1,\mathbf{d}_2),V} \cdot \frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}}.$$

Let us substitute the factorization of the system matrix into this expression:

$$\sum_L A_{LV} \cdot \frac{y_L}{\tilde{y}_L} \approx \sum_{\mathbf{d}_1} \sum_{\mathbf{d}_2} \sum_{\mathbf{i}} \sum_{\mathbf{j}} \mathbf{D}_{L(\mathbf{i},\mathbf{j}),V} \cdot p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_1} \cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_2} \cdot \frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}} =$$

$$\sum_{\mathbf{i}} \sum_{\mathbf{j}} \mathbf{D}_{L(\mathbf{i},\mathbf{j}),V} \cdot \sum_{\mathbf{d}_1} \sum_{\mathbf{d}_2} p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_1} \cdot p_{\mathbf{j}\to\mathbf{d}_2}(\vec{\omega}_{\mathbf{d}_1\to\mathbf{d}_2}) \cdot \mu_{\mathbf{d}_2} \cdot \frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}}.$$

As the crystal transport probability depends just on the distance, we can reverse the direction:

$$p_{\mathbf{i}\to\mathbf{d}_1}(\vec{\omega}) = p_{\mathbf{d}_1\to\mathbf{i}}(-\vec{\omega}).$$

Note that the nominator can also be expressed as a geometric back projection from a term obtained with convolution:

$$\sum_L A_{LV} \cdot \frac{y_L}{\tilde{y}_L} \approx \sum_{\mathbf{i}} \sum_{\mathbf{j}} \mathbf{D}_{L(\mathbf{i},\mathbf{j}),V} \cdot \mathcal{C}\left(\frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}} \cdot \mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}, \mathbf{i}, \mathbf{j} \right)$$

where the filtered term is:

$$\mathcal{C}\left(\frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}} \cdot \mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}, \mathbf{i}, \mathbf{j} \right) = \sum_{\mathbf{d}_1} \sum_{\mathbf{d}_2} p_{\mathbf{d}_1\to\mathbf{i}}(-\vec{\omega}) \cdot p_{\mathbf{d}_2\to\mathbf{j}}(-\vec{\omega}) \cdot \frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}} \cdot \mu_{\mathbf{d}_1}\mu_{\mathbf{d}_2}.$$

Now, let us consider the denominator of the back projection formula:

$$\sum_L A_{LV} = \sum_{\mathbf{d}_1} \sum_{\mathbf{d}_2} A_{L(\mathbf{d}_1,\mathbf{d}_2),V} \approx \sum_{\mathbf{i}} \sum_{\mathbf{j}} \mathbf{D}_{L(\mathbf{i},\mathbf{j}),V} \cdot \mathcal{C}\left(\mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}, \mathbf{i}, \mathbf{j} \right).$$

Summarizing, we can build the detector model into the back projection formula by computing a convolution for $\frac{y_{L(\mathbf{d}_1,\mathbf{d}_2)}}{\tilde{y}_{L(\mathbf{d}_1,\mathbf{d}_2)}} \cdot \mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}$ and $\mu_{\mathbf{d}_1} \cdot \mu_{\mathbf{d}_2}$ before executing a geometric back projection. The LOR filtering scheme is exactly the same as we perform in forward projection, just the direction vector needs to be reversed, this is why it is called *Inverse LOR filtering*.

6.3.2 Pre-computation

The input of our process is the *crystal transport probability* defined on the crystal structure, which has been computed by GATE in the following way [LCLC10]. Incident photons arriving from a direction of given inclination and azimuth angles at uniformly distributed points on the detector surface are simulated and the probabilities that this photon is absorbed in another crystal are computed. These probabilities can be visualized as a two dimensional image, where arrival probabilities are depicted by gray levels (see Figure 6.1 for an example). Also, this data can be interpreted as the (discretized) sub-critical probability density function of the offset

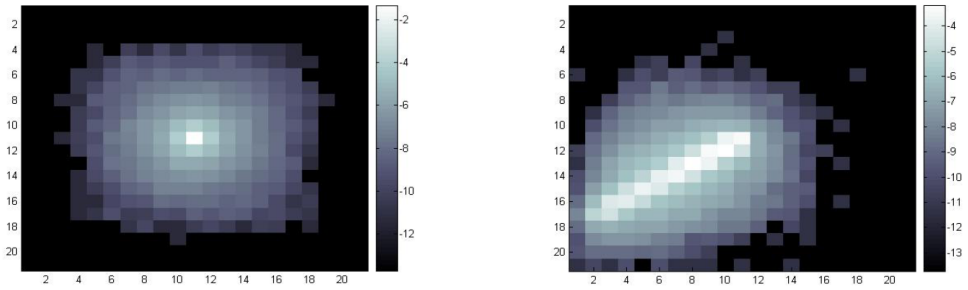


Figure 6.1: Flashing probabilities at right angle and in the case when both the inclination and azimuth angles are 40 degrees (logarithmic scale) [LCLC10].

vector between the photon impact point and the absorption point. There is a separate map for each discrete direction defined by a pair of inclination and azimuth angles.

The distinction according to azimuth angles is justified by the fact that crystals on the detector surface are squares and there are gaps filled by air between them.

The problem is that these images are too large to be sampled efficiently. So, during the pre-computation, we pre-generate relaxed MC sample sets that contain just a few samples, but their cumulative distribution is as close to the simulated distribution as possible. The relaxation step is included since we shall use these samples for high dimensional integration where the error is proportional to how well the density mimics the integrand and to the "distance" between the empirical cumulative distribution of the generated samples and the distribution used for the sample generation [SKS09].

As a result, we get the desired number of offset sample vectors for a given incoming inclination angle. The process is performed for all inclinations for which input measurement data exist, completing a *sample offset set*. For the actual reconstruction, several independent sample sets are generated for each inclination angle and sample number. By independent, we mean that the two sets are generated with independent random numbers.

6.3.3 Detector sampling during reconstruction

Offset sets were obtained by considering just one detector, so two *independent* sample sets should be used to sample the detector pair. According to the observations made in Section 2.2, we re-sample the system matrix in every iteration. We use two new independent offset sets for every iteration step, one for each of the two endpoints of LORs. Different LORs use the same offsets in an iteration step, which allows high performance on GPUs. If the iteration is longer than the half of the number of available sets, the sets are started to be used again.

6.3.4 Scatter compensation with detector response

So far we ignored scattering from the model when deriving the formulae for LOR filtering. If both scatter compensation and detector response are important (the measured object is big as in human PET and crystals are small as in small-animal PET), then both scattering and LOR filtering should be executed. However, we should be aware that it is only approximately feasible since these operators cannot be factorized. The explanation is that LOR space blurring $\mathbf{L}$ is also incident angle and energy dependent. For the direct component, we can use the direction of the LOR and the energy level of the electron $\epsilon_0 = 1$. However, for the scattered component, the incident direction is the direction of the scattering points and not the other detector, and the energy level depends on where the annihilation happened and what the scattering angle was.

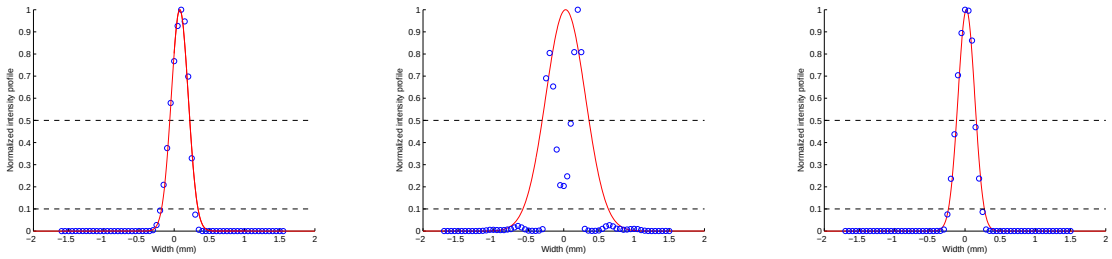
For example, for the case of single scatter simulation of Section 5.2 ($S = 1$), an approximate factoring would be the blurring of the LOR space once for each scattering point (their number

is typically a few hundred). When the line segments are combined (Step 4), then the number of hits on energy level 1 and the number of hits on lower level as well as the average energy are obtained (note that if annihilation happens on the line segment ending in this detector, then the energy level is 1; on the other hand, if annihilation happened in the other connected segment, then the energy level depends on the scattering angle as defined by the Klein-Nishina formula). Having obtained these LOR images, a LOR filtering is executed for the two energy levels. Unlike filtering the direct contribution, the incident direction is computed from the direction between the detector and the scattering point. However, this is too costly computationally.

Thus, to allow the combination of expected hits due to direct contribution and scattering, we assume that the incident direction of scattering paths is similar to the LOR direction, and the incident energy is 1 relative to the energy of the electron if the energy is in the set energy window.

6.4 Results

Since detector modeling is more important in pre-clinical PET, we used Mediso's *nanoScan PET/CT* (Section 1.1.2). On the massively parallel hardware of the GPU, LOR filtering with 64 random samples requires 7.2 seconds in $1:3$ coincidence mode, which is negligible with respect to the time of geometric LOR computation.



Ideal geometric LOR image
Geometric reconstruction

Realistic LOR measurement
Geometric reconstruction

Realistic LOR measurement
Reconstruction with detector model

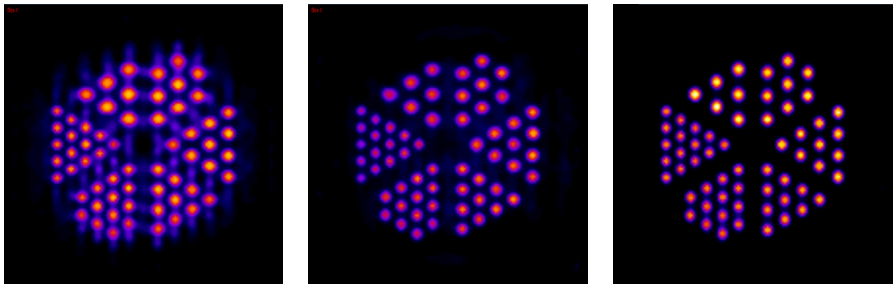
Figure 6.2: Intensity profile of an off-axis point source reconstructed using voxels of size 0.05 mm^3 after 25 EM iterations.

The spatial resolution offered by the proposed method is analyzed using an *off-axis point source* given 0.1 MBq activity for 10 second. To analyze geometric calculations, we got GATE to compute LOR images with only geometric projection and also with realistic detector model. The ideal geometric LOR image is reconstructed with the geometric model, while the realistic LOR image with both the geometric approach and with the method including detector response. Figure 6.2 shows the intensity profiles of the point source reconstructed using voxels of 0.05 mm^3. The FWHM and FWTM of geometric reconstruction are 0.35 mm and 0.55 mm, respectively, if the simulation involves only geometric effects. The FWHM and FWTM grow to 0.8 mm and 1.4 mm, respectively, if a realistic simulation, including detector blurring is reconstructed with the geometric model. However, when the reconstruction algorithm also models these phenomena, the FWHM and FWTM can be reduced to 0.35 mm and 0.55 mm, respectively, i.e. the blur due to attenuation and scattering in the detectors is fully compensated.

We compared our method with different approximations of the detector model: the *effective radius model* [C8] (Section 6.1) and a model when *only absorption* in the detectors is considered and inter-crystal scattering is ignored. Measurements of a Derenzo-like phantom were simulated using GATE. We considered two cases: a high-dose and a low-dose case, simulating 1000s (Figure 6.3) and 10s (Figure 6.4) measurements, respectively. Based on the observations made

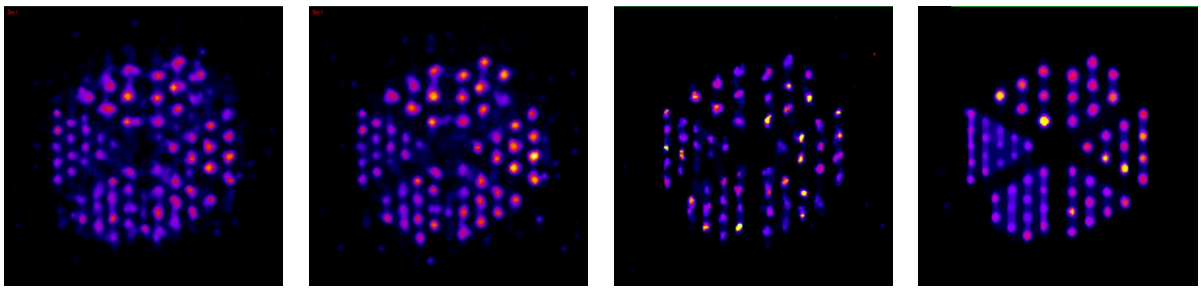
in Section 2.2, for high-dose measurements the accuracy of the back projector may be reduced without compromising image quality, thus, we used the effective radius model in the back projection. Figure 6.3 clearly shows that for the 1000s case, LOR filtering greatly outperforms the other methods, as these produce a disturbing background noise between the rods which is completely eliminated by LOR filtering. However, in the low-dose case of Figure 6.4 the accuracy of the back projector also becomes important, thus using an unmatched reconstruction with LOR filtering in the forward and effective radius model in the back projectors, respectively, the structure of the rods are not preserved. Performing inverse LOR filtering before the back projection step helps in recovering structural information at the expense of decreased contrast.

We also demonstrate the benefits of LOR filtering on the Cylinder phantom. As Figure 6.5 shows, using black detector model during the reconstruction results in a noisy image (left), while realistic detector model greatly reduces noise level and increases homogeneity (right).



Effective radius Absorption only LOR filter, unmatched

Figure 6.3: Reconstructions of the Derenzo 1000s phantom.



Effective radius Absorption only LOR filter, unmatched LOR filter, matched

Figure 6.4: Reconstructions of the Derenzo 10s phantom.

Figure 6.6 demonstrates the reconstruction result of a physical measurement taken by Mediso's nanoPET/CT system. Here the target resolution was $324 \times 315 \times 315$ voxels. Using LOR filtering, the reconstructed image has much higher contrast and lower noise level.

6.5 Conclusions

This chapter proposed the application of 4D convolution as a means to simulate inter-crystal scattering in PET reconstruction. While the incorporation of a realistic detector model significantly improves the quality of reconstructions, its time is negligible due to the efficient MC evaluation scheme. Generally, we can state that image processing methods can be and are worth being generalized to higher dimensions as well, but we have to address the curse of dimension, for which MC and quasi-MC techniques offer solutions.

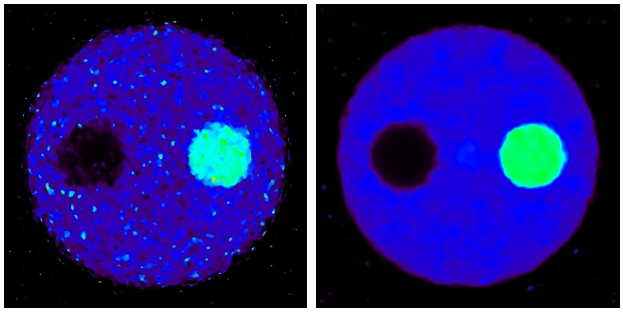
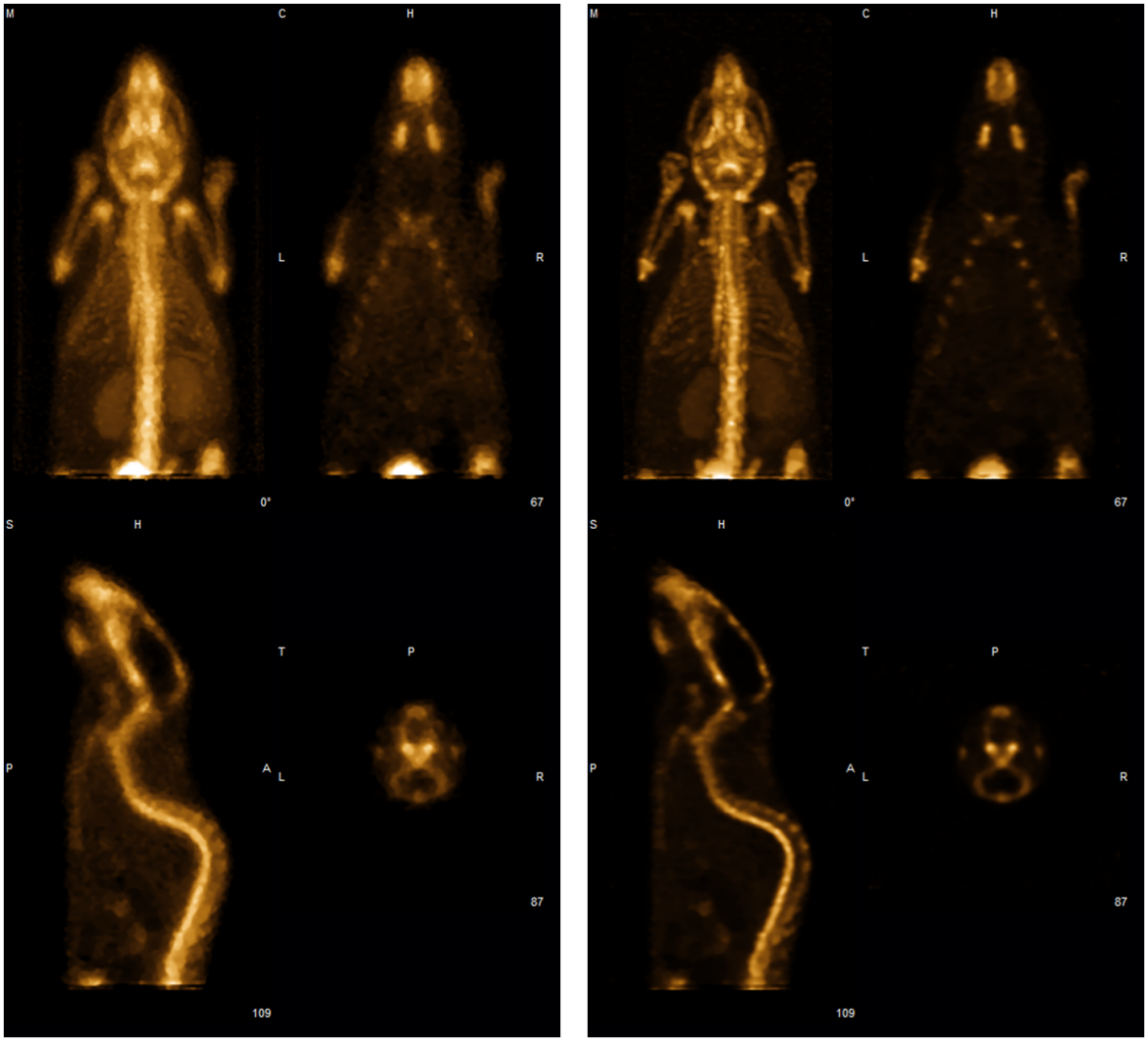


Figure 6.5: Reconstruction of the Cylinder phantom with the effective radius model (left) and with LOR filtering (right).



effective radius model LOR filtering

Figure 6.6: Mouse ^{18}F bone PET study taken by NanoPET/CT reconstructed with the effective radius model (left) and the proposed LOR filtering scheme (right). Data courtesy of P. Blower, G. Mullen, and P. Mardsden, Rayne Institute, King's College, London.

Chapter 7

Sampling techniques

The error of numerical quadrature that estimates the high-dimensional integrals of PET depends on the number of samples, which is limited by the time budget available for the reconstruction process. Consequently, we should spend the samples as effectively as possible by means of gathering information about the integrand and contributing to multiple integral estimators. On the other hand, since we aim at reducing the integration error that is achieved under the given time budget, the computational burden of sampling strategies should be low compared to that of the evaluation of integrals.

There exist several techniques for utilizing a sample more efficiently in a single estimator that have not been applied to PET so far. Filtered sampling smoothes the integrand, or from the sample's point of view, extends the range in which it can capture fine details. Section 7.1 shows its application to iterative PET for improving the accuracy of the forward projection. Multiple importance sampling allows the combination of sampling strategies that capture different parts of the integrand accurately, i.e. the sample weight is mainly influenced by the strategy that better mimics that region of the integrand. In PET, LOR driven projections are good at distributing samples such that each LOR has a sufficiently high minimum sampling density, while voxel driven approaches can focus on higher value regions of the emission density. In other words, LOR driven methods deal with "difficult" LORs, likewise, voxel driven methods deal with "difficult" voxels. Combining these strategies in the forward projection we can handle both at the same time, as it is demonstrated in Section 7.2.

Factorization of the system matrix (SM) enables the reuse of samples during a single application of a projection operator. With minor modifications of the ML-EM, it is also possible to reuse a sample in subsequent iteration steps. Section 7.3 proposes two different approaches: a method based on linear combination of the expected LOR-values of different iteration steps, and the application of the Metropolis–Hastings algorithm for PET [MRR$^+$53, KSKAC02].

7.1 Filtered sampling

For a given number of samples, the error of Monte Carlo (MC) quadrature depends on the distribution of sample points, and the variation of the integrand divided by the sample density. *Filtered sampling* [CK07] replaces the integrand by another function that has a similar integral but smaller variation, then its integral can be estimated more precisely from discrete samples (Figure 7.1). Reducing the variation means the filtering of high frequency fluctuations by a *low-pass filter*. This filter should eliminate frequencies beyond the limit corresponding to the density of the sample points. On the other hand, it should only minimally modify the integral.

In this dissertation, we propose the application of filtered sampling to increase the accuracy of forward projection during PET reconstruction. We emphasize that the objective of filtered sampling is not to reconstruct the signal but to compute its integrand more accurately and thus eliminating high frequency details can still preserve sharp features of the tracer density estimation of the ML-EM scheme.

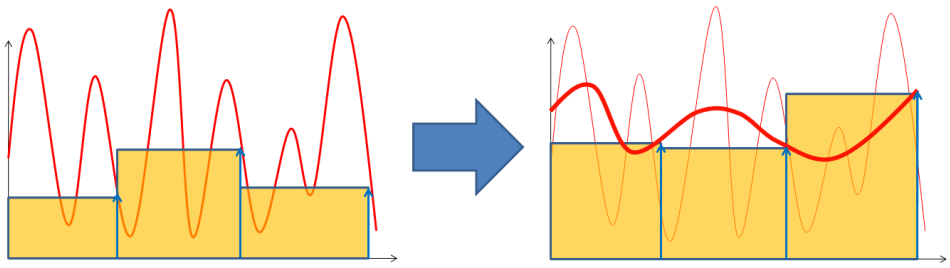


Figure 7.1: Filtered sampling reduces the approximation error of the integral quadrature by reducing the variance of the integrand.

7.1.1 Proposed filtered sampling scheme for PET

Filtered sampling introduces a pre-filtering phase in order to reduce the variance of the integrand. To see how this pre-filtering affects the reconstruction, let us consider the ML-EM reconstruction process. It can also be interpreted as a *control loop* (Figure 7.2), including forward projection

$$\tilde{y}_L = \mathcal{F}(x) = \sum_{V=1}^{N_{\text{voxel}}} \mathbf{A}_{LV} x_V$$

and back projection

$$s_V = \mathcal{B}(\tilde{y}_L) = \frac{\sum_L \mathbf{A}_{LV} \frac{y_L}{\tilde{y}_L}}{\sum_L \mathbf{A}_{LV}}, \qquad x'_V = x_V \cdot s_V.$$

This loop is stabilized when $x^{(n+1)} = x^{(n)}$, that is when scaling factors s_V are 1, which means that this loop solves the following equation for x:

$$\mathcal{B}(\mathcal{F}(x)) = 1. \tag{7.1}$$

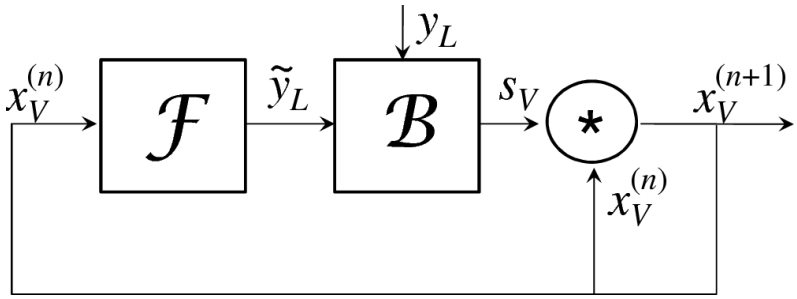


Figure 7.2: The reconstruction as a control loop. Forward projection $\mathcal{F}$ takes the actual voxel values $x_V^{(n)}$ and computes the expectation of LOR events $\tilde{y}_L$. Back projection $\mathcal{B}$ calculates a correction ratio s_V for every voxel from the expected LOR events $\tilde{y}_L$ and the measured LOR hits y_L.

Including filtering operator $\mathcal{G}$ into this loop (Figure 7.3) maps the iteration result x_V to filtered voxel value $\breve{x}_V$. The modified system stabilizes when the scaling factors s_V are 1, thus we get

$$s_V = \mathcal{B}(\mathcal{F}(\breve{x})) = \mathcal{B}(\mathcal{F}(\mathcal{G}(x))) = 1.$$

Note that this is the same equation for $\breve{x}$ as the original one (Equation 7.1) for x, thus considering $\breve{x}$ to be the output of the control system, the modified system behaves similarly to the original one. In the modified system we always have two tracer density estimates x_V and $\breve{x}_V$, that

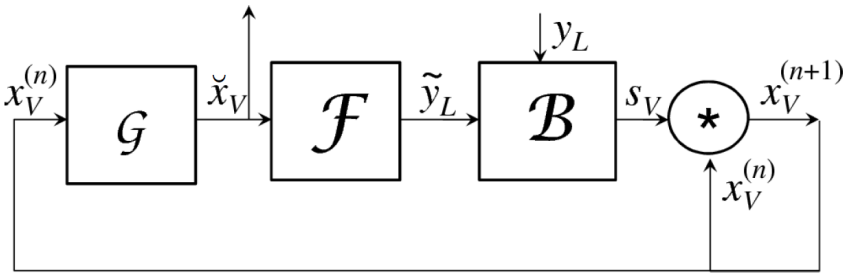
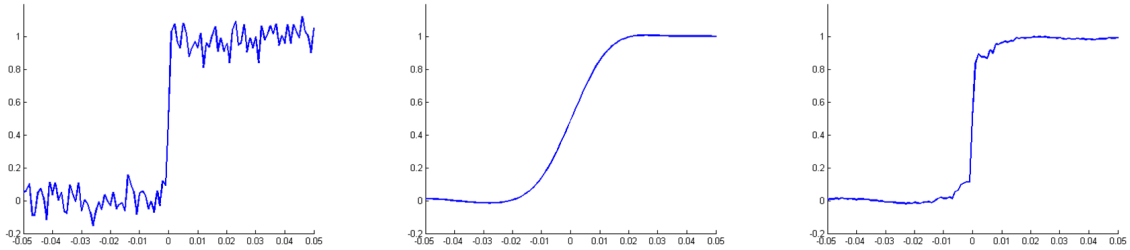


Figure 7.3: The reconstruction loop of filtered sampling. Forward projection $\mathcal{F}$ computes the expected LOR hits $\tilde{y}_L$ from the filtered voxel values $\breve{x}_V$ that are computed by applying filter $\mathcal{G}$ to the result of previous iteration $x^{(n)}$. Back projector $\mathcal{B}$ calculates the scaling factor s_V for each voxel, i.e. it obtains the product of the ratios $y_L/\tilde{y}_L$ of the measured and computed LOR hits with the transpose of the SM and divides the results by the column sums of the SM. Note that filtering only affects the input of the forward projection step, the correction made by the back projection is applied to the unfiltered estimate of the radiotracer density $x^{(n)}$.



Noisy edge | Gaussian filtered edge | Bilateral filtered edge

Figure 7.4: Effect of different filters on a noisy edge. In addition to damping high frequency noise, Gaussian filter also introduces blurring. Bilateral filters, on the other hand, filter noise while keeping sharp transitions of the data.

are related as $\breve{x} = \mathcal{G}(x)$. In addition to solution $\breve{x}$, we also get a sharpened reconstruction $x = \mathcal{G}^{-1}(\breve{x})$.

As a low-pass filter $\mathcal{G}$, we experimented with the *Gaussian* and the *Bilateral filters* [TM98]. The advantage of the Gaussian filter is that it can be defined by its mean and standard deviation, and the mean is conveniently set to zero while the standard deviation is set according to the noise that needs to be suppressed. However, as shown in Figure 7.6 and Figure 7.7, Gaussian filtering cannot preserve sharp object boundaries. A possible explanation is that sharp boundaries or edges are not bandlimited signals and have high frequency components beyond the Nyquist limit and the reasonable range of the numerical precision. These high frequency components are eliminated by the Gaussian but cannot be reproduced by its inverse.

Bilateral filters, on the other hand, preserve edges and object boundaries when their parameters are appropriately set (Figure 7.4). In the most commonly used case, the weights of these filters are products of two Gaussians: one is defined in the spatial domain, the other is in the intensity domain. More specifically, Bilateral filter B is defined as

$$B(\hat{v}, \sigma_d, \sigma_r) = \frac{\int G_{\sigma_d}(||\vec{v} - \hat{v}||) G_{\sigma_r}(x(\vec{v}) - x(\hat{v})) x(\vec{v}) \mathrm{d}\vec{v}}{\int G_{\sigma_d}(||\vec{v} - \hat{v}||) G_{\sigma_r}(x(\vec{v}) - x(\hat{v})) \mathrm{d}\vec{v}}$$

with $\hat{v}$ denoting the filtered voxel, σ_d and σ_r are the *spatial variance* and the *intensity space variance* parameters, respectively, and G_σ denotes the one dimensional Gaussian function of

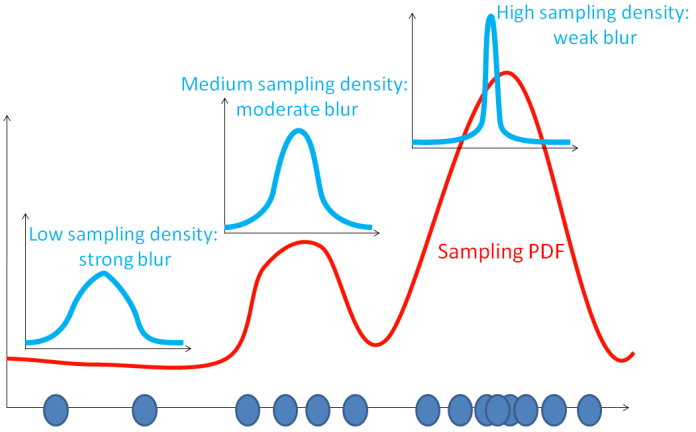


Figure 7.5: Spatially varying filtering based on the sampling PDF. Low sampling density cannot capture high variance details, thus, a strong blur is used to decrease the variance of the integrand. When the sampling density is high enough, there is no need to eliminate high frequency details.

standard deviation σ:

$$G_\sigma(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{t^2}{2\sigma^2}\right).$$

The intensity-dependent Gaussian weight ensures that neighboring voxels located on the same side of a step-like signal as the filtered voxel $\hat{v}$ get higher weights while voxels located on the other side of the edge give less contribution to the filter output, better preserving the edge. The amount of blur is controlled by spatial variance parameter σ_d, while the amount of detail kept is determined by intensity variance parameter σ_r. However, the appropriate value of σ_r is less straightforward to find, since it is given in intensity space which is object dependent.

The optimal spatial parameter for both filters can be obtained from the probability density function (PDF) of the samples used by forward projection (Figure 7.5), thus, the filter size may vary through the reconstructed volume. Voxel driven methods in most cases sample the volume proportionally to the intensity, which directly generates the PDF in each iteration. LOR driven approaches, on the other hand, usually aim for uniform sample density in LOR-space, which means that the sample density in voxel-space is approximately the same in every iteration, and the corresponding PDF can be approximated in a preprocessing step.

MIP-mapping can substantially improve the performance of filters that have voxel-dependent kernel size on a parallel architecture such as the GPU [CK07]. The first step of MIP-mapping is to build a *Gaussian pyramid* [Ros84], which is a gradual down-sampling of the original three dimensional voxel array by a factor of 2, by iteratively applying the Gaussian filter. The upper levels of the pyramid thus correspond to the lower resolution variant of the original volume where the higher frequency details are properly eliminated, whereas the lowest level stores the original array. After this pre-processing, MIP-mapping executes spatially varying filtering by sampling the volume of the resolution level corresponding to the required level of blurring. The MIP-map level can be a non-integer scalar pointing between two neighbouring levels of the pyramid. In this case, the final output is the interpolated value of the samples taken from these two levels.

The optimal filter size may be smaller than the voxel size meaning that the local sampling density is already able to capture the highest frequency details in the data that are allowed by the finite element representation. As a consequence, if the time budget given for the reconstruction allows a sufficient sample density (e.g. the maximal distance between samples is smaller than the voxel size throughout the volume of interest), filtered sampling provides no benefits.

7.1.2 Results

Due to the high arithmetic performance and bandwidth of the GPU, the execution time of the filtering step is negligible compared to that of the projection operators even for higher resolution volumes. Thus, our proposed method has practically no overhead.

To demonstrate the positive effects of using filtered sampling, we simulated a noisy measurement of a Derenzo-like phantom where the simulation corresponds to a 10 second measurement with GATE modeling Mediso's *nanoScan-PET/CT* (Section 1.1.2). We reconstructed the measured values with and without filtered sampling, setting standard deviation σ of the Gaussian to 2. We used only $N_{\text{ray}} = 1$ and $N_{\text{march}} = 36$ samples. Results are shown in Figure 7.6. Note that by using the presented filtering method, approximately the same image quality could be achieved as obtained with 8 times more samples by the original method. Figure 7.7 compares the line profiles for the Gaussian and the Bilateral filters. Using the Gaussian filter we are unable to preserve sharp boundaries, whereas with Bilateral filters the thickness of the rods are preserved.

Figure 7.6: Effects of filtered sampling on the Derenzo phantom. Using filtered sampling results in a similar image quality to a reconstruction using 8 times more samples. Note that filtered sampling using the Gaussian filter makes the rods narrower while the Bilateral filter better preserves their thickness.

To demonstrate that filtering allows to increase the resolution without the need of significantly more samples, we reconstructed a higher activity Derenzo at $288 \times 288 \times 256$ resolution as well without increasing the number of samples (Figure 7.8). At such high resolution, the Gaussian clearly demonstrates its potential since without using it, the iteration process does not converge.

Figure 7.9 shows the results for the GATE-simulated Homogeneity phantom (corresponding to a 2 second measurement), consisting of eight homogeneous cubes having different activity levels. Gaussian filtering greatly reduces the noise coming from the low-dose simulation and the low sampling rate while also stabilizing convergence.

7.1.3 Conclusions

This section proposed a filtering method to decrease the variance of the integrand of the high dimensional integrals in the forward projection step of an iterative ML-EM algorithm. We proposed the application of low-pass filtering before the forward projections, while back projection still corrects the original unfiltered voxel array. We have proven that this approach does not compromise the reconstruction and preserves the stability even if high resolution voxel arrays are reconstructed with a low number of MC sampling. All steps are implemented on the GPU where the added computational cost of filtering is negligible with respect to forward and back projection calculations.

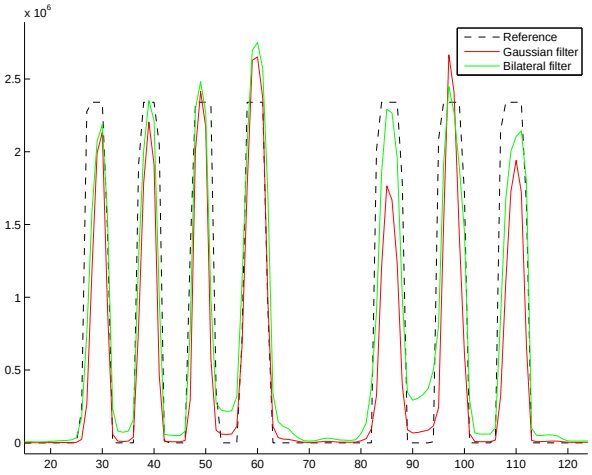
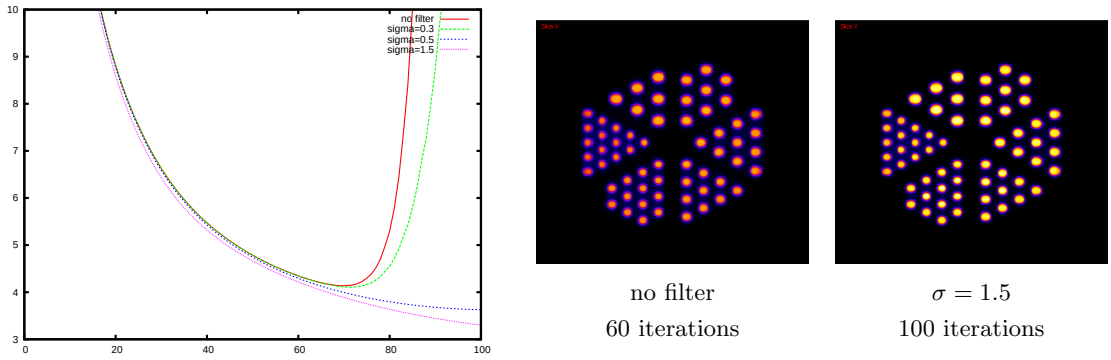


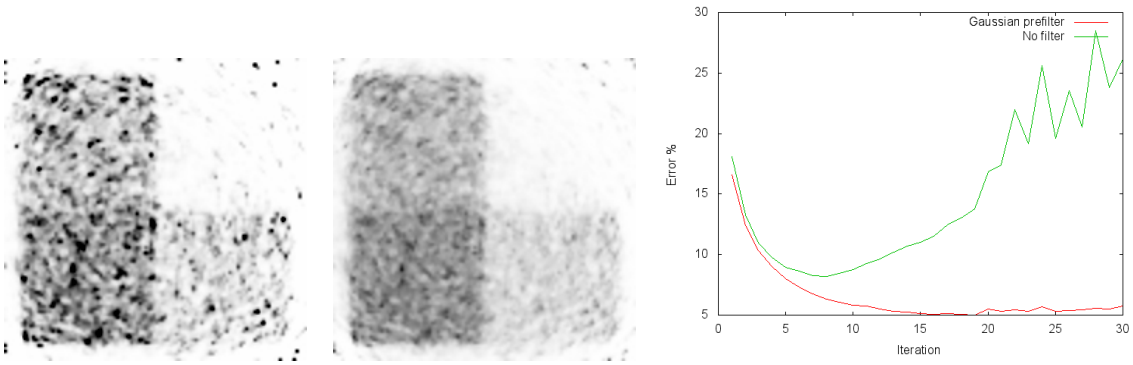
Figure 7.7: Line profile for the Derenzo phantom of Figure 7.6. Note that filtered sampling using the Gaussian filter makes the rods narrower while the Bilateral filter better preserves their thickness.



no filter
60 iterations

$\sigma = 1.5$
100 iterations

Figure 7.8: CC errors for different filtering kernel widths (left), and axial slices (right) when the Derenzo is reconstructed at double, i.e. $288 \times 288 \times 256$ resolution. We computed 100 iterations. As the reconstruction without filtering diverge, the corresponding image is shown after 60 iterations.



no filter Gaussian pre-filter

Figure 7.9: Reconstruction and error curves of the Homogeneity phantom simulated by GATE.

7.2 Multiple importance sampling

So far, we presented different LOR driven and voxel driven algorithms having different advantages and drawbacks from the point of views of importance sampling, efficiency and exploitation of the GPU hardware. In this section, we propose a strategy that can combine these techniques preserving their advantages. The combination is based on Multiple Importance Sampling. For the sake of simplicity, in Subsection 7.2.2 we first assume that detectors are ideally black, ignore positron range and scattering, and consider only phantom attenuation in this section. However, even this simplified approach can be applied in systems of physically plausible simulation if the system sensitivity is factored. On the other hand, the basic idea is extended to incorporate scattering in Subsection 7.2.3.

7.2.1 Previous work on multiple importance sampling

A MC quadrature generates N sample points $\mathbf{z}_i$ randomly with probability density $p(\mathbf{z}_i)$ in the integration domain and divides integrand $f(\mathbf{z}_i)$ evaluated at the sample points by *sample density* $d(\mathbf{z}_i) = Np(\mathbf{z}_i)$:

$$\int f(\mathbf{z})\mathrm{d}\mathbf{z} \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(\mathbf{z}_i)}{p(\mathbf{z}_i)} = \sum_{i=1}^{N}\frac{f(\mathbf{z}_i)}{d(\mathbf{z}_i)}. \tag{7.2}$$

Suppose that we have M different quadrature schemes defined by densities $d_1, \ldots, d_M$ and using $N_1, \ldots, N_M$ number of samples for the same integral. The mixture of samples from all methods is characterized by the following density

$$\hat{d}(\mathbf{z}) = \sum_{k=1}^{M} d_k(\mathbf{z}). \tag{7.3}$$

Thus *Multiple Importance Sampling* (*MIS*), i.e. the integral quadrature using the mixture of individual samples is

$$\int f(\mathbf{z})\mathrm{d}\mathbf{z} \approx \sum_{m=1}^{M}\sum_{i=1}^{N_m}\frac{f(\mathbf{z}_{m,i})}{\hat{d}(\mathbf{z}_{m,i})} = \sum_{m=1}^{M}\sum_{i=1}^{N_m}\frac{f(\mathbf{z}_{m,i})}{\sum_{k=1}^{M} d_k(\mathbf{z}_{m,i})} \tag{7.4}$$

where $\mathbf{z}_{m,i}$ is the ith sample of the mth sampling method. Note that the application of the sample mixture means the addition of the estimators of different quadrature schemes and also the modification of their densities to a single common density that is the sum of individual sampling densities.

MIS can also be interpreted as an additional weighting of samples of the techniques to be combined:

$$\int f(\mathbf{z})\mathrm{d}\mathbf{z} \approx \sum_{m=1}^{M}\sum_{i=1}^{N_m}\lambda_m(\mathbf{z}_{m,i})\frac{f(\mathbf{z}_{m,i})}{d_m(\mathbf{z}_{m,i})} \tag{7.5}$$

where the weighting scheme corresponding to Equation 7.4 is

$$\lambda_m(\mathbf{z}) = \frac{d_m(\mathbf{z})}{\sum_{k=1}^{M} d_k(\mathbf{z})}. \tag{7.6}$$

This weighting is called *balance heuristics*. Combining *unbiased* estimators, the combined estimator will also be unbiased if the sum of weights $\sum_{m=1}^{M}\lambda_m(\mathbf{z})$ is 1 for any sample, which is true for balance heuristics.

Why this combination is worth doing can be understood if we consider the modification of a density where it was small or large in a particular method. If the density around sample $\mathbf{z}_{m,i}$ in a particular method m was great with respect to other combined methods, then this particular method puts samples in this neighborhood densely, and a sample represents a small domain

fairly accurately. Thus, this particular method provides a more reliable estimate here than other methods, which should be preserved despite the estimates of other techniques. Indeed, the combination formula of Equation 7.3 and the weight of Equation 7.6 result in $\hat{d}(\mathbf{z}_{m,i}) \approx d_m(\mathbf{z}_{m,i})$ and $\lambda_m(\mathbf{z}_{m,i}) \approx 1$ if $d_m(\mathbf{z}_{m,i})$ is significantly larger than the densities of other methods, thus contribution $f(\mathbf{z}_{m,i})/d_m(\mathbf{z}_{m,i})$ of the samples of method m does not decrease in this region despite the addition of other estimators. On the other hand, if the density of method m is much smaller than other densities for particular sample $\mathbf{z}_{m,i}$, then the samples of this method in this neighborhood is sparse and the estimate is unreliable. Thus, this contribution should be suppressed, which is achieved by Equation 7.3 or Equation 7.6 resulting in $d_m(\mathbf{z}_{m,i}) \ll \hat{d}(\mathbf{z}_{m,i})$, and thus making $\lambda_m(\mathbf{z}_{m,i}) \approx 0$.

The limits of the method can be understood by considering the objective of importance sampling. The variance of the estimator is small if density $\hat{d}(\mathbf{z})$ mimics integrand $f(\mathbf{z})$ and is as large as possible. If we include more estimators, density $\hat{d}(\mathbf{z})$ will increase, which is a positive effect. However, if an included estimator is so bad that it makes the combined density less proportional to the integrand than other estimators would do, then the variance of the combined estimator may be higher than the original one. When it happens, we can solve the problem of preferring good sampling methods even more than suggested by their relative density. For example, in Equation 7.6 the weights can be defined as

$$\lambda_m(\mathbf{z}) = \frac{d_m^\alpha(\mathbf{z})}{\sum_{k=1}^{M} d_k^\alpha(\mathbf{z})} \tag{7.7}$$

which still guarantees that the sum of weights is equal to 1, but suppresses methods more in regions where they have small density if power α is greater than 1. This weighting is called *power heuristics* [VG95]. When $\alpha = 1$ we get balance heuristics back. The other extreme case, called *maximum heuristics* corresponds to $\alpha = \infty$ when $\lambda_m(\mathbf{z}) = 1$ if $d_m(\mathbf{z})$ is greater than all other densities $d_k(\mathbf{z})$ and zero otherwise.

7.2.2 Proposed MIS-based unscattered contribution computation

Chapter 4 presented two different approaches for geometric projection: a LOR driven and a voxel driven method. While deriving the formulae, the corresponding sampling densities were given, i.e. the sample weights $d^{\mathrm{A1}}(\vec{l},\vec{\omega})$ and $d^{\mathrm{A2}}(\vec{v},\vec{\omega})$ of the LOR and voxel driven approaches, respectively:

$$d^{\mathrm{A1}}(\vec{l},\vec{\omega}) = \frac{N_{\mathrm{ray}}}{D_1 D_2 G(\vec{u},\vec{w})\Delta l}, \quad d^{\mathrm{A2}}(\vec{v},\vec{\omega}) = \frac{N_{\mathrm{v}} x(\vec{v})|\vec{v}-\vec{u}|^2}{\mathcal{X} D_1 \cos\theta_{\vec{u}}}.$$

According to the theory of MIS, when two methods are combined, the sampling algorithms are left unchanged, only the sample weights are modified to include the density of all combined methods. Then, the estimators of different techniques are simply added.

The combined weight is

$$\hat{d}(\vec{v},\vec{\omega}) = \frac{N_{\mathrm{ray}}}{D_1 D_2 G(\vec{u},\vec{w})\Delta l} + \frac{N_{\mathrm{v}} x(\vec{v})|\vec{v}-\vec{u}|^2}{\mathcal{X} D_1 \cos\theta_{\vec{u}}}.$$

With the combined weights, the modified LOR driven projection and voxel driven projection compute the following estimates:

$$\hat{y}_L^{\mathrm{A1}} = \sum_{i=1}^{N_{\mathrm{ray}}} \sum_{j=1}^{N_{\mathrm{march}}} \frac{x(\vec{l}_{ij}) A(\vec{u}_i,\vec{w}_i)/(2\pi)}{\hat{d}(\vec{l}_{ij},\vec{\omega}_i)}, \qquad \hat{y}_L^{\mathrm{A2}} = \sum_{i=1}^{N_{\mathrm{v}}} \frac{x(\vec{v}_i) A(\vec{u}_i,\vec{w}_i)\xi_L(\vec{u}_i,\vec{w}_i)/(2\pi)}{\hat{d}(\vec{v}_i,\vec{\omega}_i)},$$

respectively. The final estimator is the sum of the combined estimators:

$$\tilde{y}_L \approx \hat{y}_L^{\mathrm{A1}} + \hat{y}_L^{\mathrm{A2}}.$$

The implementation of the combined sampling scheme is fairly simple. First, based on the current activity distribution a LOR centric projection is executed, which initializes every LOR value $\hat{y}_L^{\mathrm{A1}}$. In this phase a computation thread is responsible for a LOR. Then, a voxel centric projection is run in parallel, where each thread adds its contribution $\hat{y}_L^{\mathrm{A2}}$ to the affected LOR values. Sampling points $\vec{v}_i$ of the voxel centric method are generated on the CPU, and a separate thread is started for every sampling point to compute the contribution of this point to all LORs meeting here. The two phases together constitute the forward projection. Having computed the ratios of measured and expected hits, back projection is executed. While the LOR driven method is of gathering type in forward projection, it is of scattering type in back projection. Similarly, voxel driven methods are of scattering type in forward projection and of gathering type in back projection. Thus, in the combined sampling it is worth preferring LOR sampling and voxel sampling depending whether we execute forward or back projection.

7.2.3 Application to scattering materials

MIS can be used also for physically more plausible projection models. Here we consider scattering in the measured object. In case of scattering, the system sensitivity and the expected hits are high dimensional integrals, which can be expressed as summing the contributions of paths representing increasing number of scattering events

$$\mathcal{T}(\vec{v} \to L) = \sum_{S=0}^{\infty} \mathcal{T}_S(\vec{v} \to L), \tag{7.8}$$

$$\tilde{y}_L = \sum_{S=0}^{\infty} \tilde{y}_L^{(S)} = \sum_{S=0}^{\infty} \int_{\vec{v}\in\mathcal{V}} x(\vec{v})\mathcal{T}_S(\vec{v} \to L)\mathrm{d}v \tag{7.9}$$

where $\mathcal{T}_S(\vec{v} \to L)$ is the probability that a photon pair born in $\vec{v}$ undergoes exactly S scattering events in total and contributes to LOR L.

We consider two samplers, the first is the already defined LOR driven projector with attenuation, which can compute only the unscattered contribution $\tilde{y}_L^{(0)}$. The second sampler is a Direct Monte Carlo Photon Tracer (PT) that simulates photon paths from the annihilation point to the detectors, and can handle direct contribution $\tilde{y}_L^{(0)}$ as well as single and multiple scattering $\sum_{S=1}^{\infty} \tilde{y}_L^{(S)}$.

Voxel driven Direct Monte Carlo photon tracer

In scattering media, the contribution to a LOR, i.e. Equation 1.7 is an infinite dimensional integral over the photon path space. Direct Monte Carlo Photon Tracing (Section 2.1.2) samples annihilation points $\vec{v}$ and simulates the path of particles according to the laws of physics until their path are terminated by absorption, leave the system, or they are detected in one of the LORs. Upon detection, the affected LOR is given contribution $\mathcal{X}/N_{\mathrm{PT}}$ where $\mathcal{X}$ is the total activity and N_{PT} is the number of simulated paths. The estimator is

$$\tilde{y}_L^{\mathrm{PT}} \approx \frac{\mathcal{X}}{N_{\mathrm{PT}}}\#(\text{hits}) \quad \Longrightarrow \quad d^{\mathrm{PT}} = N_{\mathrm{PT}}\frac{x(\vec{v})\mathcal{T}(\vec{v} \to L)}{\mathcal{X}}.$$

This general formula has simpler form for the case when the number of scattering events is zero:

$$d_0^{\mathrm{PT}}(\vec{u}, \vec{v}, \vec{w}) = N_{\mathrm{PT}}\frac{x(\vec{v})A(\vec{u}_i, \vec{w}_i)/(2\pi)}{\mathcal{X}}.$$

Combined method

We run the two projections with the combined weighting scheme one after the other and add up their contributions. The first method is a LOR driven estimator of the unscattered component with density $d^{\mathrm{A1}}(\vec{l},\vec{\omega})$, which can be controlled by the number of rays per LOR, N_{ray}, and the number of ray marching steps per ray, N_{march}. The second method is the Photon Tracer that estimates paths of arbitrary lengths and has density d^{PT}.

The unscattered contribution is estimated by both methods, so their densities should be added when an unscattered path is obtained. In the combined approach, the LOR driven unscattered estimator becomes

$$\hat{y}_L^{\mathrm{A1}} = \sum_{i=1}^{N_{\mathrm{ray}}} \sum_{j=1}^{N_{\mathrm{march}}} \frac{x(\vec{l}_{ij})A(\vec{u}_i,\vec{w}_i)/(2\pi)}{d^{\mathrm{A1}}(\vec{l}_{ij},\vec{\omega}_i) + d_0^{\mathrm{PT}}(\vec{u}_i,\vec{l}_{ij},\vec{w}_i)},$$

where $\vec{u}_i$ and $\vec{w}_i$ are the intersection points of the ray and the detector surfaces. The PT sampler should separate unscattered paths and add the following contribution to the affected LOR:

$$\hat{y}_L^{\mathrm{PT},(0)} = \sum_{i=1}^{N_{\mathrm{PT}}} \frac{x(\vec{v}_i)A(\vec{u}_i,\vec{w}_i)\xi_L(\vec{u}_i,\vec{w}_i)/(2\pi)}{d^{\mathrm{A1}}(\vec{v}_i,\vec{\omega}_i) + d_0^{\mathrm{PT}}(\vec{u}_i,\vec{v}_i,\vec{w}_i)}$$

where $\vec{\omega}_i$ is the direction between hit points $\vec{u}_i$ and $\vec{w}_i$.

The scattered contribution is computed only by PT, so its estimator is unchanged:

$$\hat{y}_L^{\mathrm{PT},(1+)} = \frac{\mathcal{X}}{N_{\mathrm{PT}}} \#(\text{hits from scattered paths}).$$

The combined estimator is the sum of the estimators of the elementary methods:

$$\tilde{y}_L \approx \hat{y}_L^{\mathrm{A1}} + \hat{y}_L^{\mathrm{PT},(0)} + \hat{y}_L^{\mathrm{PT},(1+)}. \tag{7.10}$$

7.2.4 Results

We use the discussed MIS scheme in the forward projector of the reconstruction algorithm. The back projector is the voxel based method of Subsection 4.2.2 for maximum efficiency, which computes geometric effects but does not involve scatter simulation. The reason of using a simplified back projector is that it increases the initial convergence speed and reduces the time needed for a single iteration cycle (Section 2.2).

Performance in geometric projection

In order to evaluate the performance of the derived method in geometric projection, we follow the methodology of Section 4.3 and compare the combined method to the LOR and voxel driven approaches. LOR space L_2 error of a single projection with respect to the computation time is depicted in Figure 7.10. We consider the formerly discussed LOR driven and voxel driven methods and three MIS versions, including balance heuristics (MIS-Balance), power heuristics with $\alpha = 2$ (MIS-Power), and maximum heuristics (MIS-Max).

In Figure 7.10 we can observe that increasing the computation time and thus the number of MC samples, the error converges to zero in all cases, thus, in addition to the proposed voxel driven and LOR driven projectors, MIS combined methods are all unbiased estimators as well. The combined method is significantly better than both of the other methods for the Derenzo and is similar to the best of the LOR driven and voxel driven approaches when the Homogeneity and the Point are reconstructed. When the performances of LOR driven and voxel driven sampling are similar, then balance heuristics is optimal, but when data strongly favors either voxel driven or LOR driven sampling, maximum or power heuristics has minor advantages.

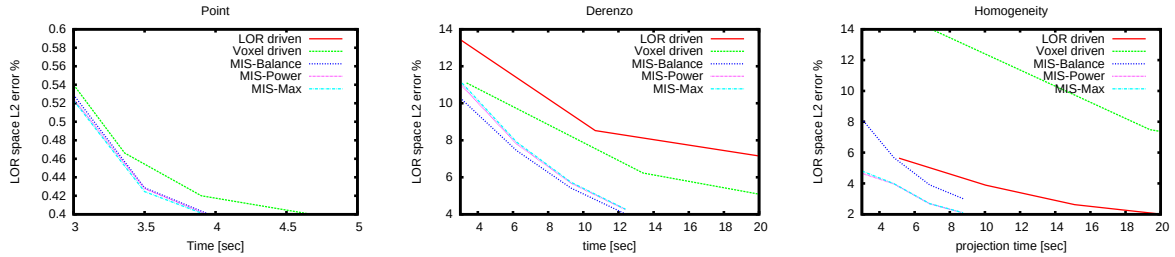


Figure 7.10: LOR space L_2 error of different projectors with respect to the computation time of the projection for the Point (left), Derenzo (middle), and the Homogeneity (right) phantoms. Note that the left figure does not include the curve of the LOR driven sampling because its error is an order of magnitude higher than those of the voxel driven and the combined methods.

In the next phase of evaluation, we include the projectors into a reconstruction algorithm, and use GATE–projected "measurements" of the Derenzo, Point Source and Homogeneity phantoms as input data (Section 4.3). Figure 7.11 shows the voxel space *CC* error of the reconstruction for the three phantoms using different N_{ray}, N_{march} and N_{v} parameters as the function of the iteration number. When N_{ray} is zero, the method is voxel driven. When N_{v} is zero, we run a LOR driven algorithm. The combined approach is characterized by nonzero N_{ray}, N_{march} and N_{v} parameters.

When too few samples are used, the error curve fluctuates and the algorithm may stop converging after certain steps. As we observed before in Section 4.3, the sufficient number of samples depends not only on the resolution of the voxel grid but also on the phantom for LOR driven and voxel driven methods. However, the combined scheme is equally good for all activity distributions. If the number of samples is sufficiently high, then the error curves of different projectors run together when they are drawn with respect to the iteration number (first row of Figure 7.11). The MIS error curves using different heuristics are very similar, so we included only the power heuristics in the figure.

Finally, in the second row of Figure 7.11 we compare the errors as functions of the time in seconds devoted to execute forward projections. Both the voxel driven and the LOR driven methods are outperformed by the combined method for all three phantoms, which allows the reduction of the number of line samples N_{ray} and ray marching steps N_{march}, and adds relatively few N_{v} volume points to compensate the missing samples at important regions. Note that for about 10^6 voxels, only 10^4 added volume samples are sufficient. The random selection and projection of 10^4 volume samples onto 180 million LORs need just 0.3 seconds on the GPU, which is negligible with respect to the times of other processing steps.

Performance in scatter compensation

Scattering in the measured object is significant in human PET, so for the purpose of examining the proposed approach in scatter compensation, we examined the projection and reconstruction of the NEMA NU2-2007 *Human IQ phantom* in Mediso *AnyScan human PET/CT* [Med10a] (Section 1.1.2). The voxel grid has $166^2 \times 75$ resolution and the voxel edge length is 2 mm. We set the energy discrimination window to [100, 750] keV. With such a wide window 35% of the measured events are direct hits, 27% are single scatters and 38% are multiple scatters.

As AnyScan detector modules cover just a small solid angle of the directional sphere, only about 2% of the photons have chance to hit the detectors. To attack this problem, in the Photon Tracer we sample annihilation photon directions non-uniformly, while the density is weighted accordingly. For comparison, we also included the Watson type *Single Scatter Simulation* algorithm of Section 5.2.

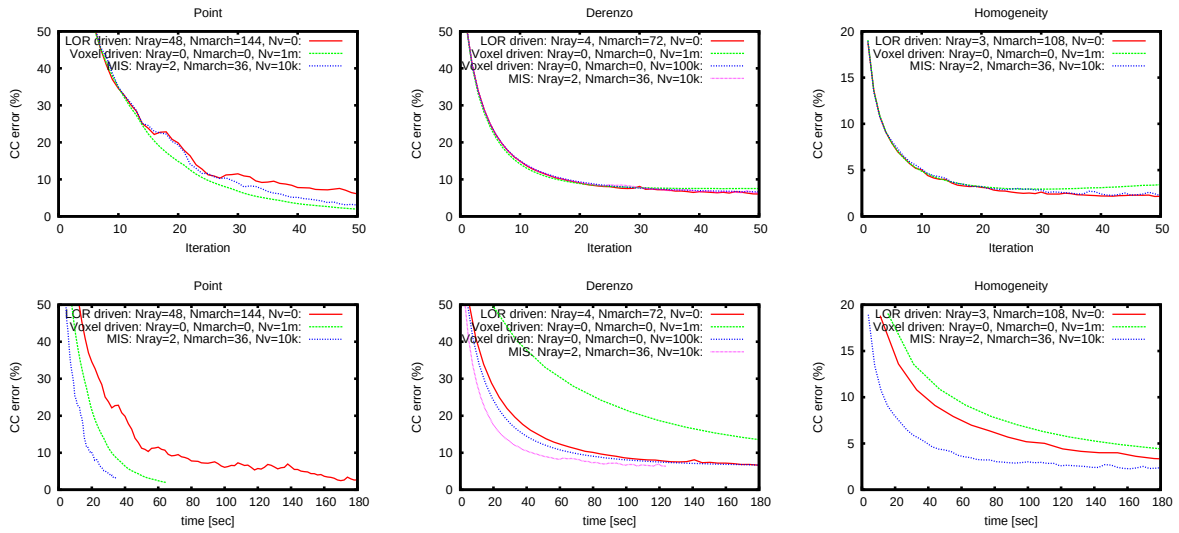


Figure 7.11: Voxel space CC error curves with respect to the iteration number (first row) and to the reconstruction time (second row) of the reconstructed Point (left), Derenzo (middle) and Homogeneity phantoms (right). The error and profile curves were made with different N_{ray}, N_{march} and N_{v} samples. The method is LOR driven when the number of voxel samples N_{v} is zero. The method is voxel driven when the number of LOR samples N_{ray} is zero. Finally, in MIS-combined reconstructions both the number of voxel samples and the number of LOR samples are non zero. We executed full EM iterations in all cases.

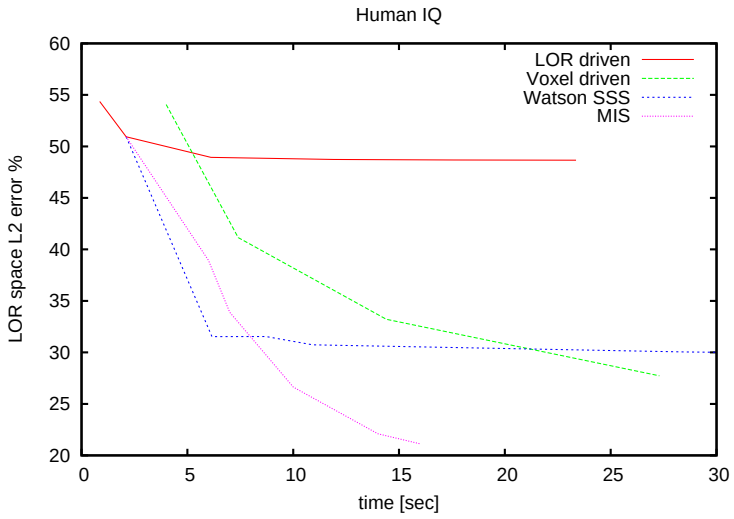


Figure 7.12: LOR space L_2 error of different projectors with respect to the computation time of the projection for the Human IQ phantom.

First, the projectors are validated computing the LOR space L_2 error with respect to a reference projection generated by GATE with 10^{12} annihilation photons (Figure 7.12). We considered different samples that are multiples of $N_{\text{ray}} = 1, N_{\text{march}} = 21$, $N_{\text{scatter}} = 50$, and $N_{\text{PT}} = 10^6$, and the error curves are depicted with respect to the computation time. The LOR driven and the Watson type methods compute only the unscattered term and at most single scattering, respectively, thus they are fast converging at the beginning but for higher number of samples the projection error stops decreasing, i.e. these methods are biased. We used balance heuristics for MIS, which combines the unbiasedness of the Photon Tracer and the speed of LOR driven methods.

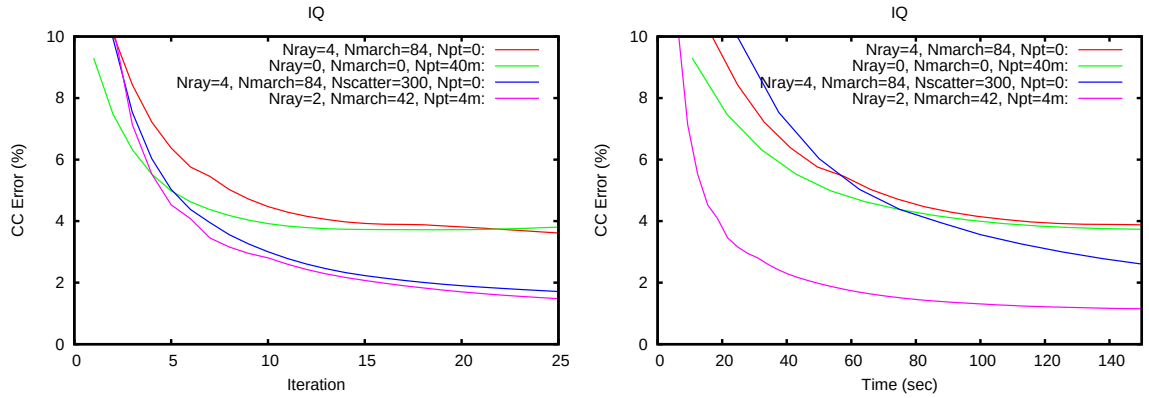


Figure 7.13: CC error curves reconstructing the Human IQ phantom with different N_{ray}, N_{march}, N_{scatter} and N_{PT} sample numbers, depicted as functions of the iteration number (left) and the time devoted to forward projections (right).

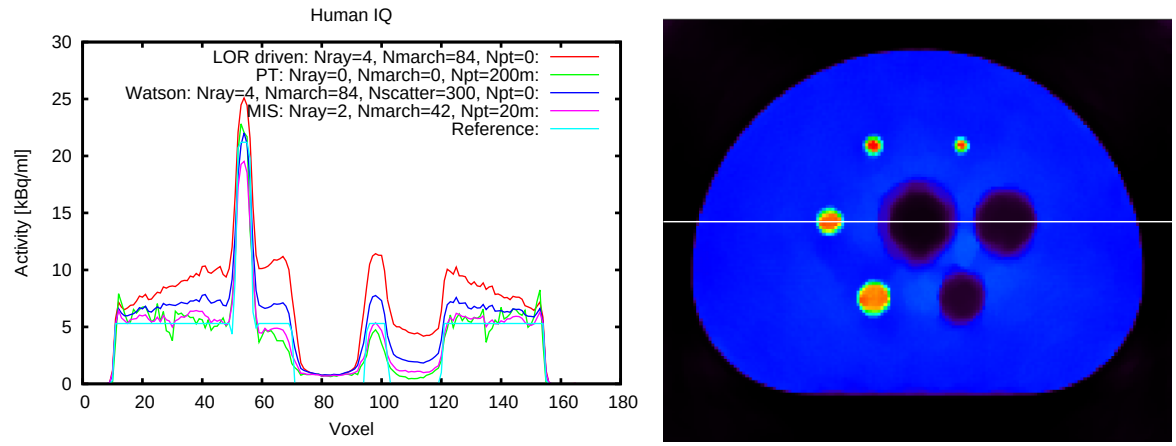


Figure 7.14: Profile curves of the Human IQ phantom along the centerline crossing a hot sphere, the lung, and a cold sphere (left) and transaxial slice obtained with the combined method (right).

The performance of the projectors is also evaluated in ML-EM reconstruction. We get GATE to simulate a 500 sec long measurement of the NEMA NU2-2007 *Human IQ phantom* of 40 MBq activity, which resulted in noisy measurements of 3 SNR. Figure 7.13 shows the error curves with respect to the number of iterations and the total forward projection time. The profile curves on the centerline and also a transaxial slice obtained with the combined method are depicted by Figure 7.14. Note that using a LOR driven method ($N_{\text{ray}} = 4$, $N_{\text{march}} = 84$, $N_{\text{PT}} = 0$) alone, we cannot expect fully accurate reconstruction since this method computes only

the direct contribution and ignores the scattered photon hits. As a consequence, false activity is added that can be observed in the profile curve. The Photon Tracer ($N_{\text{ray}} = 0$, $N_{\text{march}} = 0$, $N_{\text{PT}} = 40 \cdot 10^6$) and the MIS-combined ($N_{\text{ray}} = 2$, $N_{\text{march}} = 42$, $N_{\text{PT}} = 4 \cdot 10^6$) methods involve unbiased multiple scattering estimators, thus they can theoretically lead to accurate reconstructions. The Photon Tracer requires at least 40 million photon pairs per iteration to make the process converge, but even with such number of samples the reconstruction result involves some noise. The combined method is not only more accurate but also much faster since it needs just 4 million photon pairs per iteration to add scattering and to help the LOR centric approach in the computation of the direct contribution.

7.2.5 Conclusions

This section proposed the MIS-based combination of different MC methods, including LOR centric and voxel centric approaches. The individual methods have different advantages and drawbacks from the point of views of numerical accuracy and GPU execution performance. The proposed combination automatically finds an optimal weighting, which keeps the advantages of all techniques. The combined sampling can result in accurate projections using less discrete samples and thus can reduce the time of reconstruction.

We have applied the concept for the computation of geometric projection with attenuation and also for multiple scattering compensation. MIS can also be applied in other MC estimators developed for the same or other physical phenomena. For example, we can consider the combination of more efficient geometric projectors, like the distance driven method, or scattering in the detector crystals can also be simulated with input crystal driven or output crystal driven approaches, whose advantages can be combined with MIS.

7.3 Exploiting samples of previous ML-EM iterations

This section presents modifications of the ML-EM iteration scheme to reduce the reconstruction error due to the on-the-fly MC approximations of forward and back projections. Our goal is to increase the accuracy and the stability of the iterative solution while keeping the number of random samples and therefore the reconstruction time low. As we shall demonstrate in this section, the voxel intensity has a positive bias due to the MC estimate of forward projection. This bias and also the fluctuation of the voxel intensity can be reduced by making the forward projection more accurate.

We propose two solutions that exploit additional samples from previous iteration steps, improving accuracy of the current step without requiring more samples or more processing time: *Averaging iteration* and *Metropolis iteration*. Averaging iteration [SK99, SK00] averages forward projection estimates during the iteration sequence. Metropolis iteration rejects those forward projection estimates that would compromise the reconstruction and also guarantees the unbiasedness of the tracer density estimate, which we shall show formally. We demonstrate that these techniques allow a significant reduction of the required number of samples and thus the reconstruction time.

MC quadrature means that a high-dimensional integral is interpreted as the expected value of a multi-dimensional random variable, and then the expected value is approximated by an average of random samples. As computer library functions can return uniformly distributed pseudo-random values, random variables of other distributions are obtained by transforming random variables that are uniformly distributed in the unit domain. Thus, MC quadrature requires the transformation of the integration domain to a unit cube where coordinates correspond to the independently generated uniform random variables. We call this transformed integration domain the *primary sample space* and denote it by $\mathcal{U}$. In a single iteration step we estimate many high-dimensional integrals, one for each LOR in forward projection and one for each voxel in back projection. To prepare for MC estimation in forward projection, the domain of particle

paths of Equation 1.7 is transformed to the primary sample space:

$$\tilde{y}_L = \int_{\mathcal{V}} x(\vec{v})\mathcal{T}(\vec{v} \to L)\mathrm{d}v = \int_{\mathcal{U}} \hat{y}_L(\mathbf{u})\mathrm{d}\mathbf{u} \tag{7.11}$$

where $\hat{y}(\mathbf{u})$ is the LOR hit estimate associated with the random variable samples in $\mathbf{u}$. The probability density of random variables uniformly distributed in the unit cube is $p(\mathbf{u}) = 1$, thus this integral is the expected value of $\hat{y}_L(\mathbf{u})$, which can be approximated from a single point $\mathbf{u}$ if the fluctuation (variance) of $\hat{y}_L(\mathbf{u})$ is small.

Unfortunately, the unbiasedness of the LOR estimates does not guarantee unbiased voxel estimates in the back projection. Back projection is not a linear function of the computed LOR, but depends inversely proportionally to it via ratios $y_L/\tilde{y}_L$ and it also involves the SM elements in the denominator as a *sensitivity image* $\sum_L A_{LV}$. The sensitivity image has typically small variance since it involves all SM elements corresponding to a single voxel independently of the actual voxel estimates. However, the ratio of measured and computed LOR values can introduce significant fluctuations unless the forward projection is very accurate. Let us consider the expectation of the ratio of measured and computed hits, $y_L/\hat{y}_L$. According to the relation of harmonic and arithmetic means, or equivalently to the Jensen's inequality taking into account that $1/\hat{y}_L$ is a convex function, we obtain:

$$E\left[\frac{y_L}{\hat{y}_L(\mathbf{u})}\right] = \int_{\mathcal{U}} \frac{y_L}{\hat{y}_L(\mathbf{u})}\mathrm{d}\mathbf{u} \geq \frac{y_L}{\int_{\mathcal{U}} \hat{y}_L(\mathbf{u})\mathrm{d}\mathbf{u}} = \frac{y_L}{\tilde{y}_L}. \tag{7.12}$$

This inequality states that $y_L/\tilde{y}_L$ has a random estimator of positive bias. An intuitive graphical interpretation of this result is shown by Figure 7.15. Here we assume that the iteration is already close to the fixed point, so different estimates are around the expected detector hit corresponding to the maximum likelihood. Note that the division in the back projection may amplify forward projection error causing large fluctuations, especially when $\tilde{y}_L$ is close to zero.

We propose two modified iteration schemes to solve this problem, averaging iteration and Metropolis iteration, which are presented in the next sections.

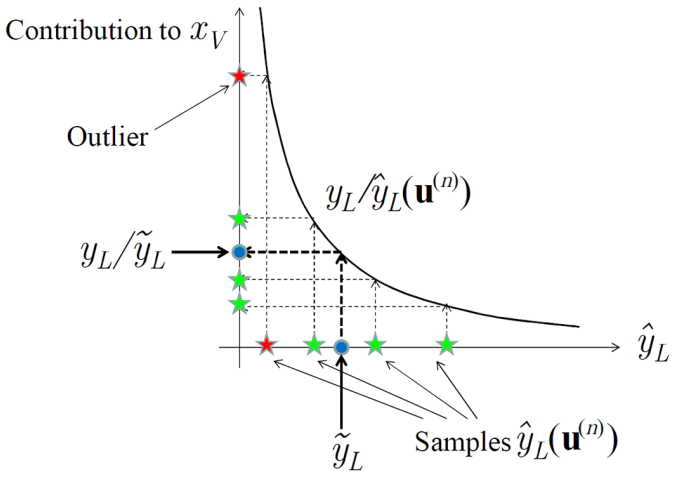


Figure 7.15: Expected LOR hit number $\tilde{y}_L$ is approximated by random samples $\hat{y}_L(\mathbf{u}^{(n)})$ in iteration step n, which have mean $\tilde{y}_L$. These random samples are shown on the horizontal axis. Back projection computes ratio $y_L/\hat{y}_L(\mathbf{u}^{(n)})$ to obtain voxel updates, which is a non-linear, convex function, resulting in voxel values that may be much higher than the correct value $y_L/\tilde{y}_L$. These overshooting samples are responsible for a positive bias and occasionally cause a large random increase in the voxel value.

7.3.1 Averaging iteration

Recall that the MC estimation in forward projection results in computed LOR hit values $\hat{y}_L$ that fluctuate around their exact value $\tilde{y}_L$ (Equation 7.11):

$$\tilde{y}_L = \int_{\mathcal{V}} x(\vec{v})\mathcal{T}(\vec{v} \to L)\mathrm{d}v = \int_{\mathcal{U}} \hat{y}_L(\mathbf{u})\mathrm{d}\mathbf{u}$$

Thus, if MC estimates of subsequent iteration steps use independent random numbers, it is worth averaging the calculated LOR hits obtained in different iteration steps to reduce the scale of the fluctuation.

Formally, we obtain the expected LOR hits $\tilde{y}_L^{(n)}$ in iteration step n as the weighted average of the actual MC estimate $\hat{y}_L(\mathbf{u}^{(n)})$ and its previous value $\tilde{y}_L^{(n-1)}$:

$$\tilde{y}_L^{(n)} = (1 - \tau_n)\, \tilde{y}_L^{(n-1)} + \tau_n \hat{y}_L(\mathbf{u}^{(n)}) \tag{7.13}$$

where τ_n is the decreasing weight of the estimate obtained in the current iteration step. The weighting scheme can be defined, for example, as $\tau_n = \min(\lambda/n, 1)$, where $\lambda \geq 1$ is a user defined parameter describing how quickly averaging iteration forgets earlier results.

The ML-EM algorithm incorporating averaging in forward projection is as follows:

for $n = 1$ **to** m **do** *// iterations*
  for $L = 1$ **to** N_{LOR} **do** *// Forward project + average*
    $\hat{y}_L$ = Forward Project $\mathbf{x}^{(n-1)}$ with a MC algorithm.
    $\tau_n = \min(\lambda/n, 1)$.
    $\tilde{y}_L^{(n)} = (1 - \tau_n)\, \tilde{y}_L^{(n-1)} + \tau_n \hat{y}_L$.
  endfor
  for $V = 1$ **to** N_{voxel} **do** *// Back project*
    $x_V^{(n)}$ = Back Project $y_L / \tilde{y}_L^{(n)}$ with a MC algorithm.
  endfor
endfor

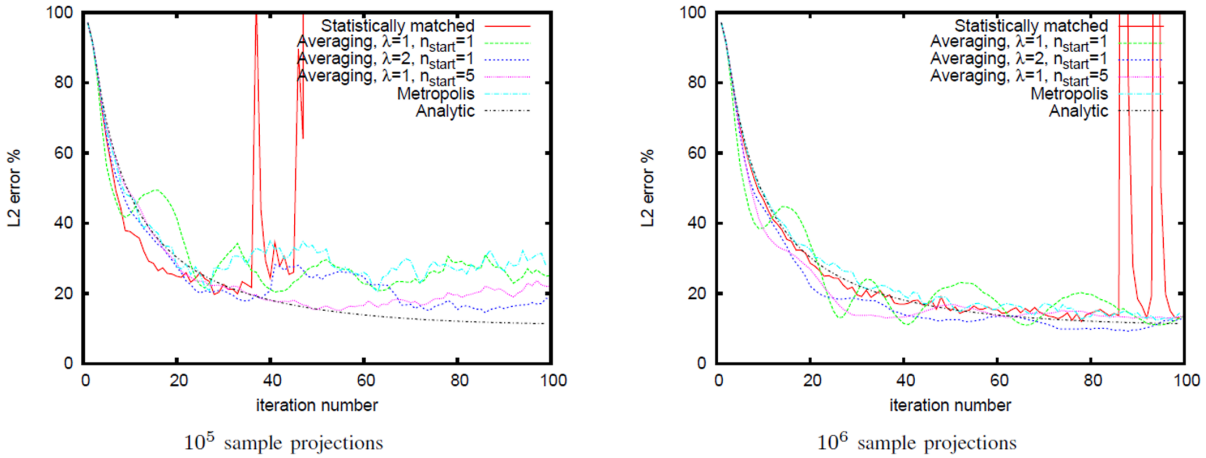


Figure 7.16: Relative L_2 error curves of averaging and Metropolis iterations and their comparison to statistically matched iteration. The waves in the error curve of averaging iteration started at the first step with $\lambda = 1$ is eliminated by either starting averaging just at step $n_{\mathrm{start}} = 5$ or setting $\lambda = 2$.

If we are close to the fixed point x_V^* and execute m additional iterations with $\tau_m = 1/m$, then averaging iteration is similar to iterating with the average of the SMs, i.e. with the matrix

that is computed using m times more samples:

$$E_{\text{avg}}\left[\tilde{y}_L^{(m)}\right] \approx \frac{1}{m}\sum_{n=1}^{m}\tilde{y}_L^{(n)} = \frac{1}{m}\sum_{n=1}^{m}\sum_{V=1}^{N_{\text{voxel}}}\mathbf{F}_{LV}^{(n)}x_V^* = \sum_{V=1}^{N_{\text{voxel}}}\frac{\sum_{n=1}^{m}\mathbf{F}_{LV}^{(n)}}{m}x_V^*,$$

where $\mathbf{F}^{(n)}$ denotes the forward projection matrix of iteration n. However, when we are farther from the fixed point, LOR estimates $\hat{y}_L$ are different not only due to the random fluctuation of the MC sampling but also because of the early evolution of the reconstructed activity $\mathbf{x}^{(n)}$. Averaging iteration reduces random fluctuations, but also slows down the convergence towards the solution having the maximum likelihood especially when there are still significant differences between subsequent iteration steps. This problem can be solved by starting averaging only later in the iteration sequence, or by increasing parameter λ. Note that $\lambda = \infty$ corresponds to statistically matched iteration (Section 2.2.1).

Figure 7.16 compares the relative L_2 error curves of averaging iteration using statistically independent forward and back projections and statistically matched iteration for the 2D example of Section 2.2.1, and Figure 7.17 depicts the reconstructions. Note that averaging iteration is stable unlike statistically matched iteration even for small sample numbers. Its wavy L_2 curve is due to the problem that averaging is not fast enough to forget estimates of the first iteration steps, which can be solved by starting averaging at iteration step $n_{\text{start}} = 5$ or by increasing parameter λ from 1 to 2.

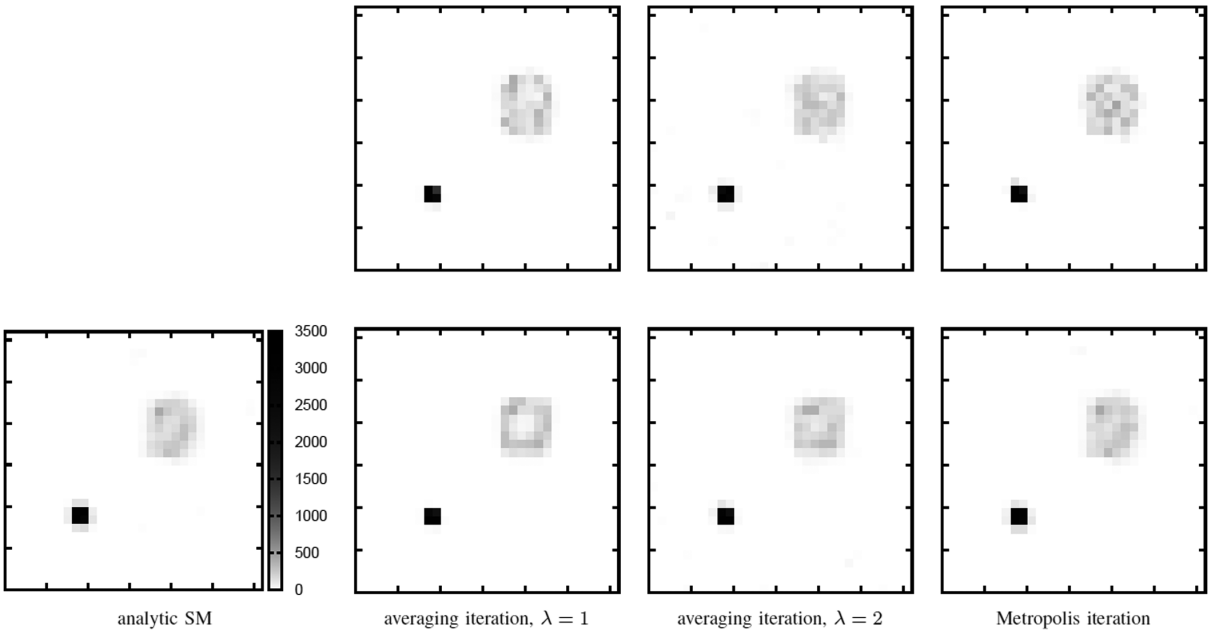


Figure 7.17: Reconstructed activity obtained with analytic SM, and with averaging and Metropolis iterations using 10^5 sample projections (upper row) and 10^6 sample projections (lower row).

7.3.2 Metropolis iteration

First, we present Metropolis iteration intuitively, based on the analysis of Figure 7.15. The problems of positive bias and the large fluctuations are caused by random samples $\hat{y}_L(\mathbf{u})$ that are much lower than their expected value $\tilde{y}_L$ and result in large overshooting values $y_L/\hat{y}_L(\mathbf{u})$ in the voxel contributions. To attack this problem, these overshooting samples (outliers) are rejected. We suppose that the MC algorithm provides us with a sequence of random *tentative samples* during the iteration, from which real samples are generated by rejecting the outliers and replacing them with the last accepted sample. On the one hand, such replacement would

decrease the expectation of the voxel contribution, thus the positive bias can be eliminated. On the other hand, the probability of very large $y_L/\hat{y}_L(\mathbf{u})$ ratios is decreased, so is the probability of fluctuations when these effects are added in different LORs.

A classical MC forward projector obtains samples in the primary sample space with uniform probability density, and transforms these samples as required by the particular algorithm. The added rejection or replacement scheme modifies the uniform probability in the primary sample space. We wish to have a rejection scheme and an associated probability density $p_{\text{Met}}(\mathbf{u})$ that make the updated voxels have unbiased estimates. Let us show that this requirement is met if density $p_{\text{Met}}(\mathbf{u})$ is proportional to forward projection estimate $\hat{y}_L(\mathbf{u})$. The ratio of proportionality is obtained from the requirement that $p_{\text{Met}}(\mathbf{u})$ is a probability density, thus its integral is equal to 1:

$$p_{\text{Met}}(\mathbf{u}) = \frac{\hat{y}_L(\mathbf{u})}{\int\limits_{\mathcal{U}} \hat{y}_L(\mathbf{u})\mathrm{d}\mathbf{u}} = \frac{\hat{y}_L(\mathbf{u})}{\tilde{y}_L}. \tag{7.14}$$

The expectation of random estimate $y_L/\hat{y}_L(\mathbf{u})$ is then indeed equal to the exact ratio:

$$E_{\text{Met}}\left[\frac{y_L}{\hat{y}_L(\mathbf{u})}\right] = \int\limits_{\mathcal{U}} \frac{y_L}{\hat{y}_L(\mathbf{u})} p_{\text{Met}}(\mathbf{u})\mathrm{d}\mathbf{u} = \frac{y_L}{\tilde{y}_L}. \tag{7.15}$$

The only remaining task is the elaboration of a rejection scheme that keeps a sample with probability density $p_{\text{Met}}(\mathbf{u}) \propto \hat{y}_L(\mathbf{u})$. Such tasks can be solved with the *Metropolis method* [MRR+53, KSKAC02]. The sequence of tentative samples $\mathbf{u}^{(n)}$ are uniformly distributed in the primary sample space and are statistically independent. Metropolis sampling establishes a Markov chain $\mathbf{u}_{\text{Met}}^{(n)}$ by randomly rejecting a new tentative element $\mathbf{u}^{(n)}$ based on its contribution $\hat{y}_L(\mathbf{u}^{(n)})$ and on the contribution of the previously accepted sample $\hat{y}_L(\mathbf{u}_{\text{Met}}^{(n-1)})$. The decision uses the *acceptance probability* $a(\mathbf{u}^{(n)})$ that is the ratio of the contributions of the tentative sample and the previously accepted sample.

The state transition probability of the Markov chain is

$$P(\mathbf{u} \to \mathbf{u}') = \min\left(\frac{\hat{y}_L(\mathbf{u}')}{\hat{y}_L(\mathbf{u})}, 1\right). \tag{7.16}$$

Thus, the ratio of state transition probabilities in two directions between two states is

$$\frac{P(\mathbf{u} \to \mathbf{u}')}{P(\mathbf{u}' \to \mathbf{u})} = \frac{\hat{y}_L(\mathbf{u}')}{\hat{y}_L(\mathbf{u})}. \tag{7.17}$$

As tentative samples are generated for each primary sample space point associated with a non-zero contribution, the established Markov chain is ergodic, i.e. it has a unique stationary distribution $p_\infty(\mathbf{u}) = \lim p_n(\mathbf{u})$ which is independent of the initial state. The stationary distribution must satisfy the *balance requirement*, i.e. the probability of outflow from a state equals to the probability of inflow, thus

$$\int\limits_{\mathcal{U}} p_\infty(\mathbf{u}) P(\mathbf{u} \to \mathbf{u}')\mathrm{d}\mathbf{u}' = \int\limits_{\mathcal{U}} p_\infty(\mathbf{u}') P(\mathbf{u}' \to \mathbf{u})\mathrm{d}\mathbf{u}'. \tag{7.18}$$

Using Equation 7.17, it is easy to see that the balance requirement is met when $p_\infty(\mathbf{u}) \propto \hat{y}_L(\mathbf{u})$, and the condition of uniqueness guarantees that the sample density will converge to this distribution.

The ML-EM algorithm incorporating Metropolis sampling in forward projection is as follows:

```
for n = 1 to m do                                          // iterations
    for L = 1 to L = N_LOR do                              // Forward project
        ŷ_L = Forward Project x^(n−1) with a MC algorithm.
```

$a_L = \min\{\hat{y}_L/\tilde{y}_L^{(n)}, 1\}$. // *acceptance probability*
Generate random number r in $[0, 1)$.
if $r < a_L$ **then** $\tilde{y}_L^{(n)} = \hat{y}_L$ // *accept with probability* a_L
else $\tilde{y}_L^{(n)} = \tilde{y}_L^{(n-1)}$
endfor
for $V = 1$ **to** $V = N_{\text{voxel}}$ **do** // *Back project*
$x_V^{(n)}$ = Back Project $y_L/\tilde{y}_L^{(n)}$ with a MC algorithm.
endfor
endfor

In the stationary case, the Markov process generates samples with a density proportional to $\hat{y}_L(\mathbf{u})$, but early samples may be drawn from a different distribution. This may result in a *start-up bias*, which is typically handled by ignoring the first few samples corresponding to the burn-in period while the process is not stationary yet. However, we do not have to ignore early samples because of the following two reasons. As our method generates tentative samples independently of the current sample, the perturbation is as large as the whole primary sample space, thus the start-up bias disappears quickly. On the other hand, instead of computing just a single integral, we execute an iteration where each step requires its own projection integrals. Even if some error is made early in the iteration due to the start-up bias when the activity is only roughly estimated anyway, the error will be corrected by later iteration steps when the start-up bias already vanishes.

The relative L_2 error curves of Metropolis iteration are also included in Figure 7.16 and its reconstruction result is compared to averaging iteration in Figure 7.17. We can observe that the Metropolis method has higher fluctuation than averaging iteration but does not introduce waves in the error curves. The superior stability of averaging iteration is due to the fact that it exploits the samples of all previous iteration steps when the forward projection is estimated while Metropolis iteration effectively combines just the last iteration steps. However, this is also an advantage since the convergence of Metropolis is not slowed down by the effect of earlier samples, and therefore it does not require additional, volume dependent parameters like λ or start of averaging n_{start}.

Method	30% L_2 error	20% L_2 error
Fixed	80	300
Deterministically matched	80	290
Statistically matched	17	37
Averaging ($\lambda = 2$)	2	11
Metropolis	6	19

Table 7.1: Total number of samples in millions needed to take the L_2 error below 20% and 30%, respectively for the 2D analytic test case of Section 1.4.1.

The relative performance of different sampling techniques can be characterized by counting the total number of samples — i.e. the product of the number of samples per iteration and the number of iterations — needed to reduce and keep the error below a given threshold (note that stochastic sampling has fluctuating error curve, so we need to find the number of samples that guarantees that the maximum of the fluctuation is less than the threshold). Table 7.1 shows a parameter study performed by reconstructing the 2D data of Section 1.4.1 with sample numbers per iteration in the range of 10^5–10^7 and finding the minimum of the product of the sample number and the number of iterations. As the reconstruction time is proportional to the total number of samples, this table shows the relative speed of different methods. For example, averaging iteration is 25–40 times faster than fixed iteration (Section 2.2.1), which may be considered as a classical method, and 3–8 times faster than statistically matched stochastic

iteration (Section 2.2.1). Metropolis iteration, on the other hand, is 13 times faster than fixed iteration and 2–3 times faster than statistically matched iteration.

7.3.3 Results

In this section we consider different factored phases of a PET reconstruction algorithm, including geometric projection, scattering in the detector, and scattering in the measured object. However, we note that the proposed scheme can also be used with other projection models, including, for example, not factored MC particle transport algorithms [WCK+09] or processing Time of Flight (ToF) data as well.

Geometric projections and scattering in the detectors

Geometric projections without and with detector scattering calculation are tested with Mediso's small animal *nanoScan-PET/CT* [Med10b] (Section 1.1.2).

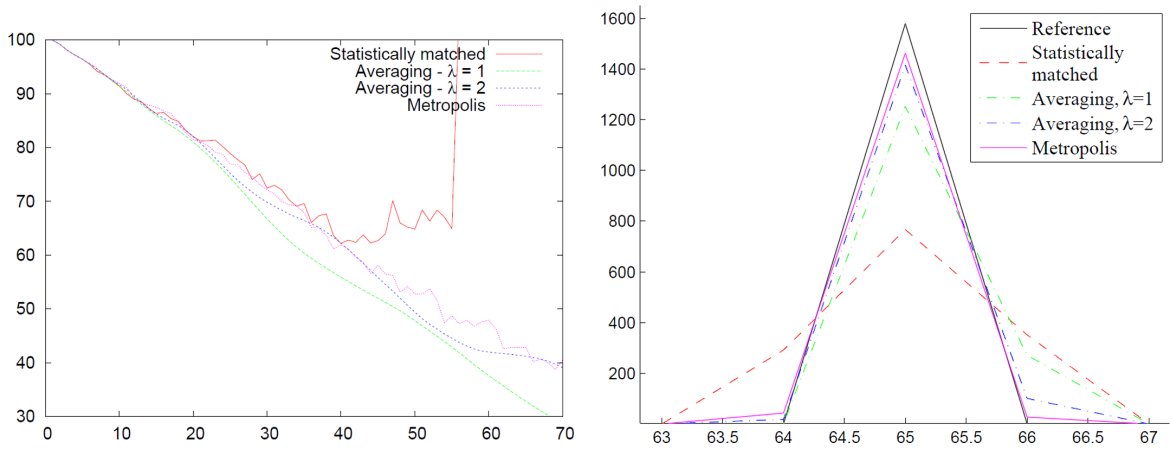


Figure 7.18: Relative L_2 error with respect to the number of iterations (left) and profile curves obtained after 100 iteration steps (right) of the point source reconstructions. The activity density for the profile curve is in Bqs/mm^3, the unit on the horizontal axis is the edge size of a voxel that is equal to 0.185 mm.

To test the efficiency of averaging and Metropolis iterations, first we took an *off-axis point source* of 0.1 MBq activity, placed 2 mm North and 1 mm East from the axis and simulated a 10 sec long measurement with GATE, assuming ideal black detectors, to obtain the input for the reconstruction. We run four reconstructions of the GATE simulation: averaging iteration with two λ factors, Metropolis iteration, and also statistically matched iteration (Section 2.2.1) for comparison. Figure 7.18 (left) shows the relative L_2 error curves of the reconstruction of the point source using 0.185 mm^3 voxels and $N_{\text{ray}} = 4$ line samples per LOR. Right of Figure 7.18 depicts the line profiles of the reconstructed tri-linear activity density. Note that statistically matched iteration exhibits drastic oscillations in the error value and results in a blurred reconstruction, unlike averaging and Metropolis iteration methods. We repeated the statistically matched iteration with $N_{\text{ray}} = 8, 16$ and 24 samples, and compared them to the 4 sample averaging or Metropolis iterations. We concluded that statistically matched iteration gets better than the averaging and Metropolis iterations if it uses more than 16 samples instead of 4. It means that averaging and Metropolis iterations allow 4–5 times faster projections.

We also examined the *Micro Derenzo phantom* with rod diameters $1.0, 1.1, \ldots, 1.5$ mm in different segments. The Derenzo was virtually filled with 1.6 MBq activity and we simulated a 10 sec, i.e. low-dose, and a 1000 sec, i.e. high-dose, measurement with GATE assuming ideal black

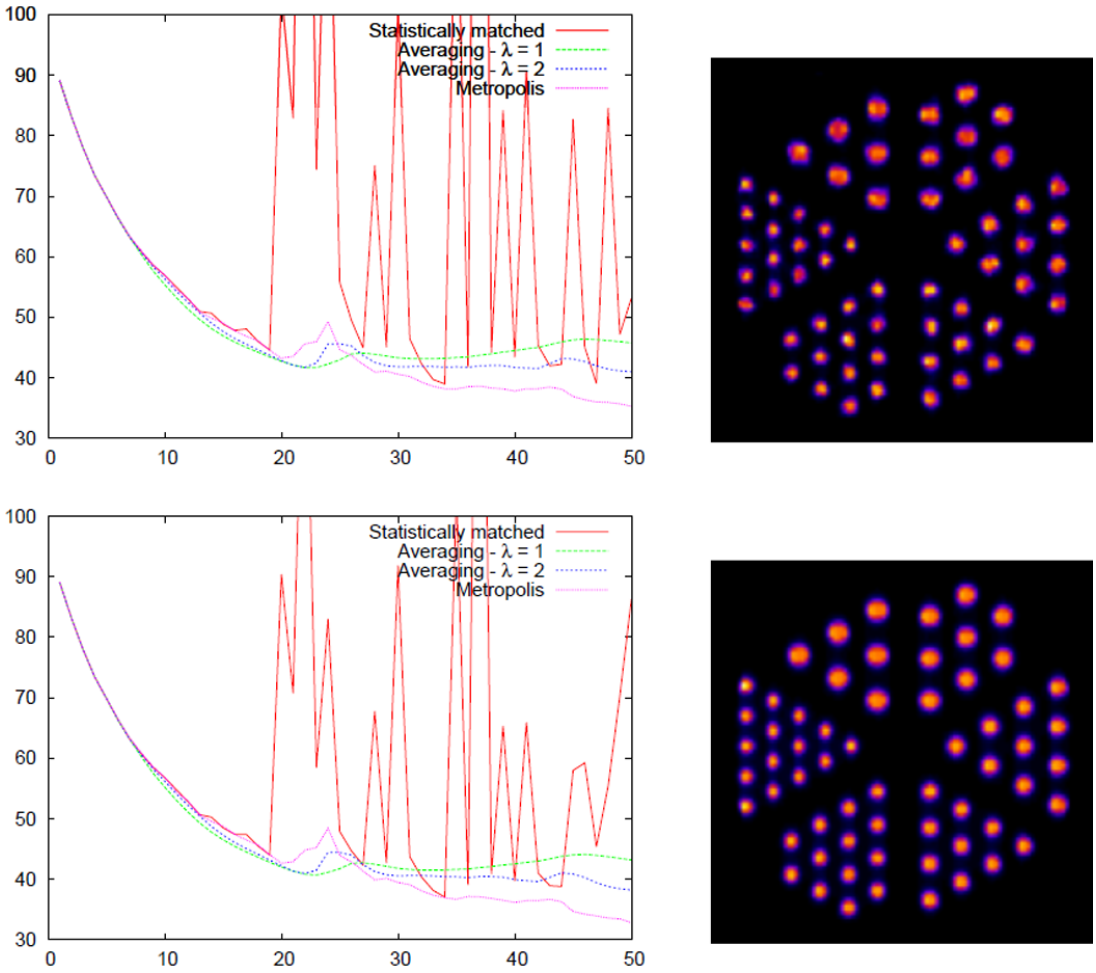


Figure 7.19: Relative L_2 error curves obtained during reconstructing the 10 second (upper row) and the 1000 second (lower row) Derenzo phantoms with $N_{\text{ray}} = 1$ random ray per LOR. The cross section images are obtained with Metropolis iteration.

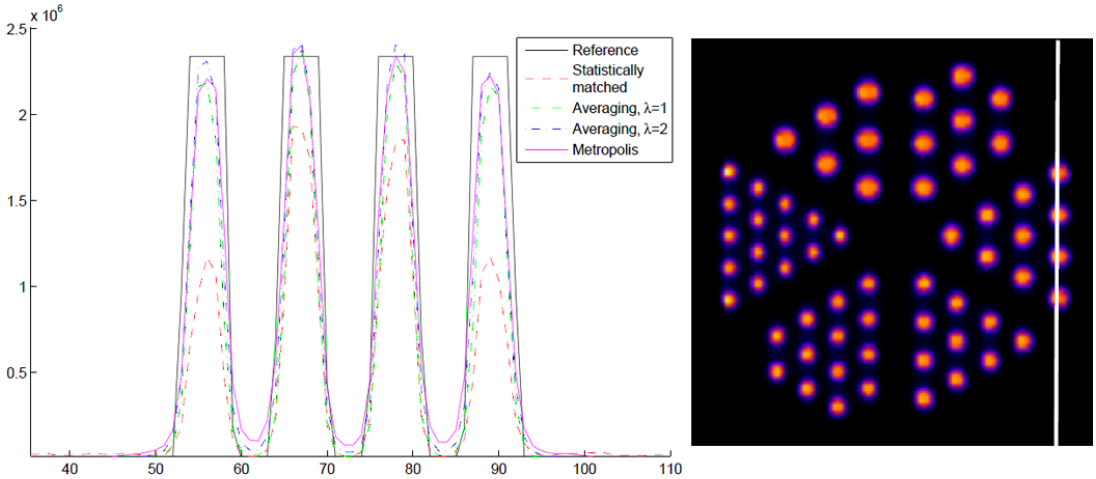
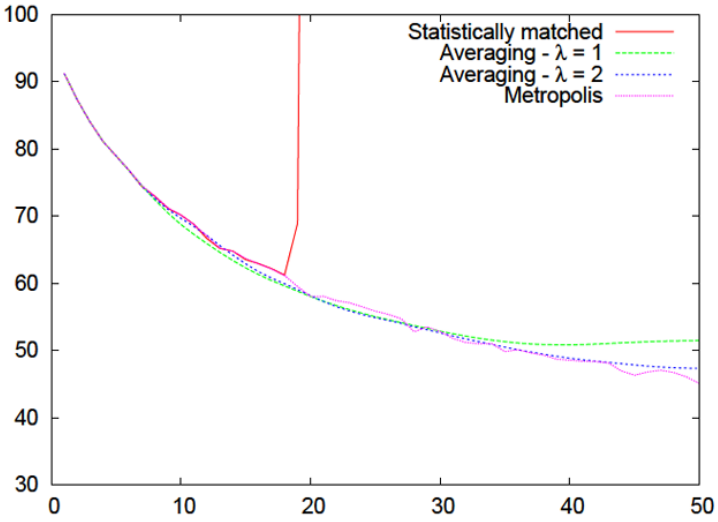


Figure 7.20: Line profiles of the Derenzo 1000 sec phantom reconstructed with statistically matched, averaging and Metropolis iterations. The unit on the horizontal axis is the edge size of a voxel i.e. 0.23 mm.

detectors. The average hits per LOR are 0.05 and 5 in the 10 sec and 1000 sec measurements, respectively. Figure 7.19 depicts the error and the cross section images of the Derenzo phantom reconstruction with $N_{\text{ray}} = 1$ random ray per LOR, and Figure 7.20 shows a line profile of the reconstructed volume. With only $N_{\text{ray}} = 1$ line sample, the statistically matched iteration cannot correctly reconstruct the phantom, but both averaging and Metropolis iterations can since they utilize the MC estimates from more than one iteration step. Statistically matched iteration would require at least 3 line samples to have the same error curve as averaging or Metropolis iteration has with 1 line sample, i.e. averaging or Metropolis iteration offers 3 times faster projection in this case. The evaluation of the low number of samples used in a single iteration is very fast on the GPU, a full averaging or Metropolis forward projection requires 0.9 seconds.



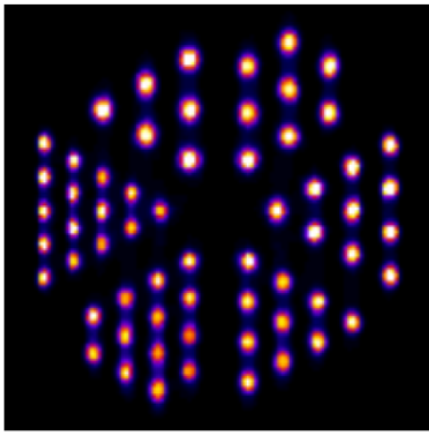


Figure 7.21: Detector scattering compensation with averaging and Metropolis iterations using $N_{\text{ray}} = 1$ ray for geometric projection and $N_{\text{det}} = 64$ random LOR space offsets per LOR for MC simulation of scattering in the detector.

To test the application of averaging and Metropolis iterations in detector scattering compensation, we set LYSO crystals in the GATE simulation projecting the 1000 sec Derenzo phantom and turned on the detector model (Section 6.3) in our system as well. Detector scattering not only blurs the sinogram, but also reduces the average hits per LOR to 0.6 in the 1000 sec simulation due to the possibility that photons fly through or get lost in the detector. The results obtained with $N_{\text{det}} = 64$ 4D LOR space offsets mimicking the probability density of photon transfer in the detectors at the two ends of the LOR are shown by Figure 7.21. With this number of samples, the computation time of detector blurring compensation in a single projection of the full EM iteration needs 7.2 sec.

Scattering in the measured object

Scattering in the measured object is significant in human PET, so for the purpose of examining the proposed iterations in object scatter compensation, we model the *AnyScan human PET/CT* [Med10a] (Section 1.1.2).

We used GATE to produce "measurements" of the human IQ phantom, first setting the energy discrimination window to 400–600 keV. Figure 7.22 shows the reconstruction at $166^2 \times 75$ voxel resolution and the NEMA-NU2-2007 contrast evaluation results with single scatter compensation (Section 5.2) taking only $N_{\text{scatter}} = 5$ random scattering point samples per iteration. The reason of selecting so few scattering point samples is to emphasize the differences of the examined iteration types. We also repeated the reconstruction for data generated by GATE

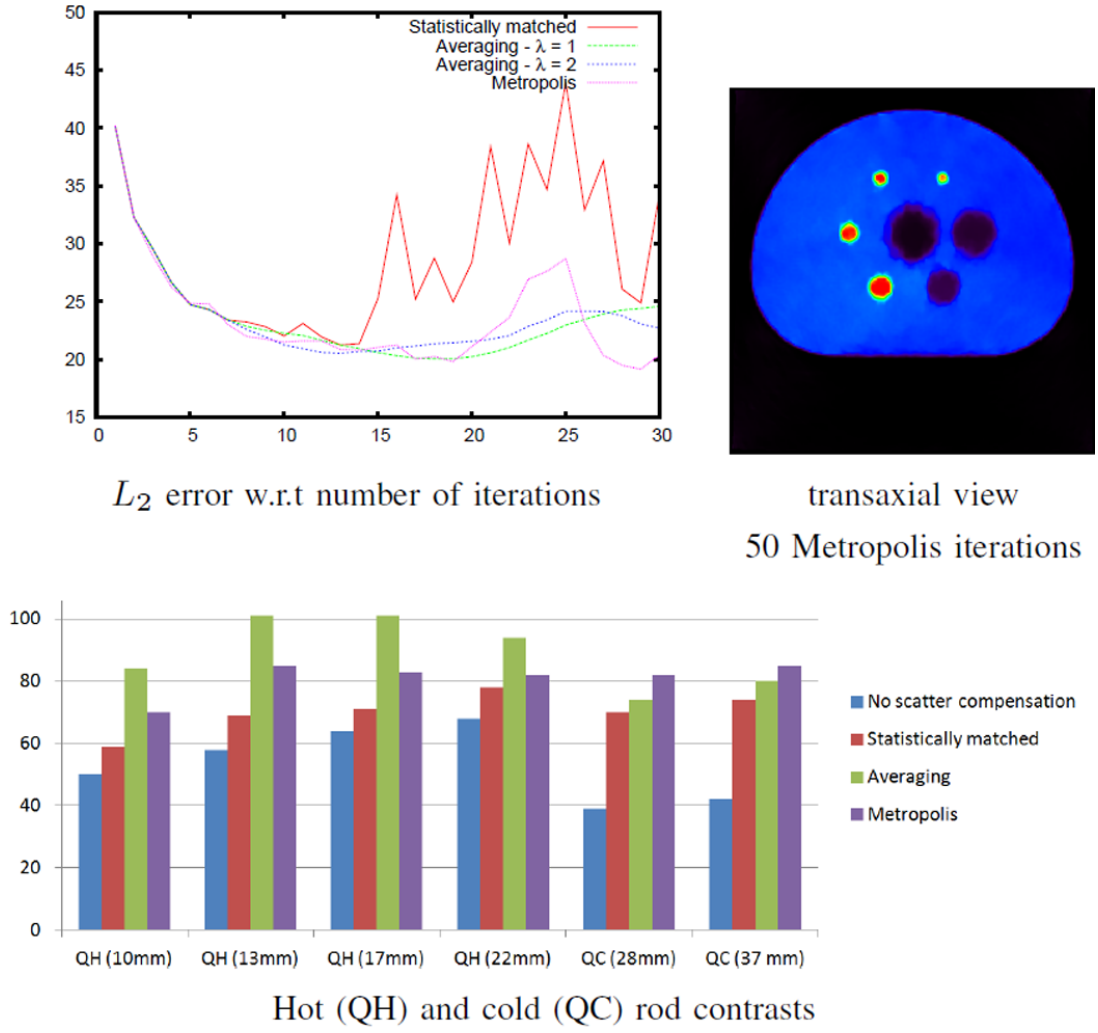
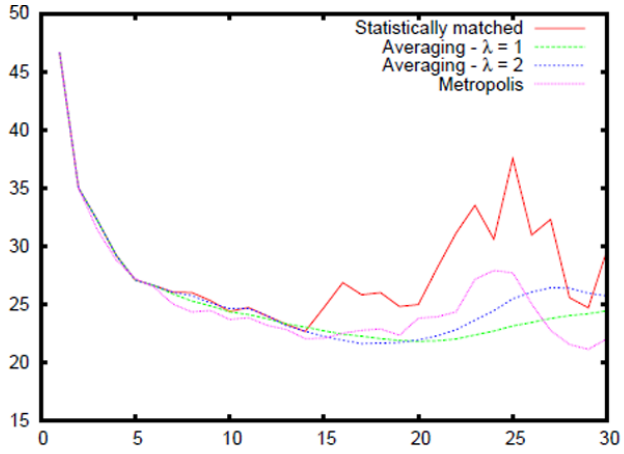
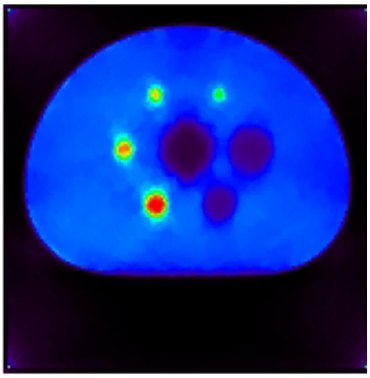


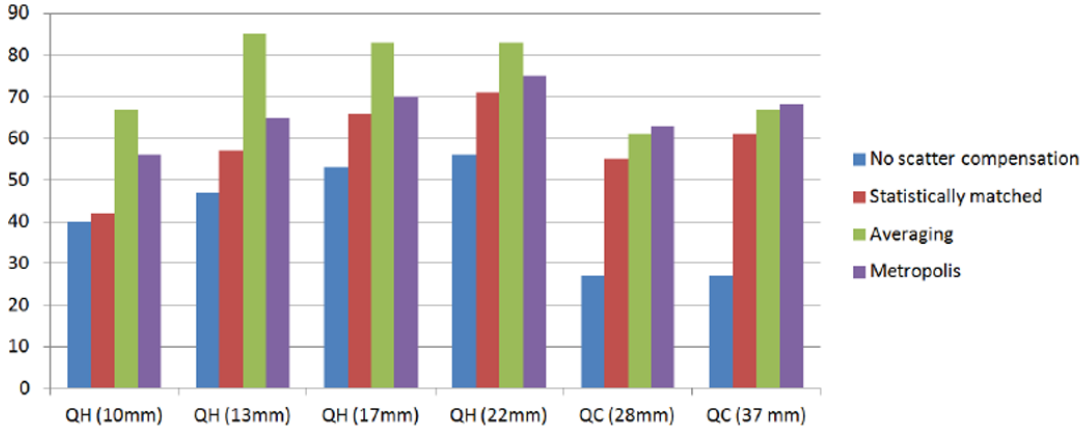
Figure 7.22: Relative L_2 error curves obtained during reconstructing the NEMA human IQ phantom with $N_{\text{scatter}} = 5$ global scattering points in each iteration step, and the NEMA-NU2-2007 hot and cold contrast values after 50 iterations. The "measured data" is produced with GATE with 400-600 keV energy window. Single scatter compensation is executed in every iteration step after the 5th iteration.



L_2 error w.r.t number of iterations



transaxial view
50 averaging iterations



Hot (QH) and cold (QC) rod contrasts

Figure 7.23: Relative L_2 error curves obtained during reconstructing the NEMA human IQ phantom with $N_{\text{scatter}} = 5$ global scattering points in each iteration step. The "measured data" is produced with GATE with 100-700 keV energy window. Multiple scatter compensation is executed in every iteration step after the 5th iteration.

with 100–700 keV energy window, approximately compensating multiple scattering as proposed in Section 5.3.2. Note that all iteration types do a fairly good job in scatter compensation, but the L_2 error and contrast values are better in averaging and Metropolis iterations than in statistically matched iteration, which would require at least 10 samples to provide similar quality. Averaging iteration with $\lambda = 1$ is particularly good at improving the hot contrast. Single scatter compensation with 5 samples needs 1.1 sec in each full EM iteration step.

7.3.4 Conclusions

This section proposed the application of averaging and Metropolis iteration schemes to improve the speed and accuracy of emission tomography reconstruction working with on-the-fly MC estimates. The goal is to distribute the cost of more samples in different iteration steps, thus we get higher accuracy without increasing the computation time or storing any of the SM elements. We demonstrated the application of the method in three factored phases of a binned fully-3D PET reconstruction, including the geometric projections, scattering in the detectors during both forward and back projections, and scattering in the measured object only in forward projection. The method works not only with full EM but also with OSEM and is suitable for GPU implementation.

Chapter 8

Thesis summary

This thesis work concentrates on particle simulation methods for positron emission tomography. We aim at efficient, Monte Carlo techniques that can exploit the features of the GPU.

Thesis Group 1. Positron range simulation

Positron range simulation for heterogeneous materials in the frequency domain

Positron range can be approximated as a material dependent blurring on the estimated positron emission density. In high-resolution small animal PET systems, the average free path length of positrons may be many times longer than the linear size of voxels, which means that the blurring kernel should have a very large support so its voxel space calculation would take prohibitively long.

I proposed a fast GPU-based solution to compensate positron range effects which executes filtering in the frequency domain, thus provides a performance that is independent of the size of the blurring kernel. To handle heterogeneous media, we execute Fast Fourier Transforms for each material type and for appropriately modulated tracer densities and merge these partial results into a density that describes the composed, heterogeneous medium. As Fast Fourier Transform requires the filter kernels on the same resolution as the tracer density is defined, I also presented efficient methods for re-sampling the probability densities of positron range for different resolutions and basis functions [C11].

Thesis Group 2. Geometric projections

LOR driven estimator for the GPU

Existing LOR driven forward projector methods assume piece-wise constant basis functions and use analytic approximations for the five-dimensional integral of the geometric projection. The deterministic error made by the analytic approximations results in a biased estimator and thus modifies the fixed point of the iteration. Additionally, existing methods use varying sample number to evaluate line integrals and thus would assign different computational load to parallel threads causing their divergence if these methods are ported to the GPU.

I proposed an unbiased sampling scheme that offers efficient parallel implementation using the same set of samples for each thread and derived the sample density formulae based on integral transformations. The surfaces of the detectors are re-sampled uniformly in every iteration step, and a random offset is added for the line samples along the line to guarantee that every point that may correspond to a LOR is sampled with a positive probability [J8, C3, C5].

Voxel driven estimator with importance sampling

Efficient parallel implementation requires the geometric projection to be LOR driven in the forward projector and voxel driven in the back projector. However, these approaches may be wasting in the sense that they do not consider the annihilation density during sampling, thus are poor for importance sampling.

I proposed a voxel driven geometric projection scheme that computes the contribution of a voxel to LORs and derived the sample density formulae based on integral transformations. First sample points in the volume of interest are generated mimicking the annihilation density, then for each sample point detector surface points are sampled fairly uniformly. This allows the activity distribution to be taken into account in the forward projection, using importance sampling of the voxels. Furthermore, being a voxel centric approach, it provides an efficient parallel implementation of the back projector [J8, C5].

Thesis Group 3. Scattering in the measured object

Scatter simulation with photoelectric absorption

Watson's method is a popular choice of single scatter simulation and its implementation becomes very efficient with the reuse of line segments. However, it simulates only single photon–material interaction and is not feasible for dense materials since it ignores photoelectric absorption and downsamples the set of detectors.

I proposed several GPU-based improvements for Watson's algorithm. First, in order to make the method suitable for dense materials, I showed how to include photoelectric absorption into the model, without loosing the ability to precompute paths. Second, I proposed the application of importance sampling for the selection of scattering samples. Third, by giving an efficient GPU implementation that includes path reuse, I showed that the method can work in 3D without needing to downsample the detector space [J1, J2, J4, B1, C4, C5, C7, D2].

Multiple forward scattering for free

Due to truncation of the Neumann series where terms represent higher order bounces, particle transport results are underestimated, thus the radiotracer density in the reconstruction becomes overestimated. This negative bias can be eliminated by Russian roulette which is inefficient on the GPU and it trades bias for noise. The contribution of the terms above truncation can also be approximately re-introduced by blurring and scaling the calculated contribution. However, these methods cannot accurately consider patient specific data and have the added computational cost of filtering.

I presented a simple approximate method to improve the accuracy of scatter computation in PET without increasing the computation time. The proposed method exploits the facts that higher order scattering is a low frequency phenomenon and the Compton effect is strongly forward scattering in the energy window of PET. I showed that the directly not evaluated terms of the Neumann series can approximately be incorporated by an appropriate modification of the scattering cross section while the highest considered term is calculated. The correction factor depends just on the geometry of the detector and is robust to the variation of patient specific data [C9, D9].

Thesis Group 4. Detector model

Detector model with Monte Carlo LOR filtering

When modeling inter-crystal scattering to increase the accuracy of PET, we can take advantage of the fact that the structure of the detector is fixed, and most of the corresponding scattering

calculation can be ported to a pre-processing phase. Pre-computing the scattering probabilities inside the crystals, the final system response is the convolution of the geometric response obtained with the assumption that crystals are ideal absorbers and the crystal transport probability matrix. This convolution depends on the incident direction and is four-dimensional which poses complexity problems as the complexity of the naive convolution evaluation grows exponentially with the dimension of the domain.

I proposed a Monte Carlo method to attack the curse of dimension in higher dimensional spatial varying convolution. The method replaces the summation of the signal values weighted with the filter kernel by a random sum of signal values at points sampled with the density of the filter kernel. Decoupling the geometric phase from the detector model, i.e. pre-computing the direct contribution before the convolution is evaluated, I demonstrated that these techniques have just negligible overhead on the GPU [C5, C6, C8, D5].

Thesis Group 5. Sampling techniques

Filtered sampling for PET

On-the-fly system matrix generation, i.e. approximation of high dimensional integrals is usually attacked by Monte Carlo quadrature and importance sampling. Determining the number of samples used by the estimators belongs to the classical tradeoff problem between accuracy and computational time. However, the approximation error mainly comes from the measurement noise and high frequency components of the measured object that cannot be captured by the given sample density. Filtered sampling applies low-pass filter on the integrand before sampling in order to suppress both noise and high frequency details.

I proposed the application of filtered sampling for the forward projection step of iterative ML-EM based PET reconstruction to decrease the variance of the integrand and thus to reduce the error of integral estimation for a given set of samples. The input of the forward projection is filtered using a low-pass filter, which reduces noise and increases the probability that samples do not miss high frequency peaks — e.g. a point source — and requires only negligible overhead on the GPU. I showed that the iteration converges to a modified fixed point, from which the original function can be extracted by applying the same filter [C5, C10].

Multiple importance sampling for PET

Voxel driven methods can focus on point like features while LOR driven approaches are good in reconstructing large, homogeneous regions. Existing methods use voxel and LOR driven approaches exclusively which means that they cannot achieve good performance for every types of input.

I proposed the application of Multiple Importance Sampling (MIS) in fully 3D PET to speed up the iterative reconstruction process. The proposed method combines the results of LOR driven and voxel driven projections keeping their advantages, like importance sampling, performance and parallel execution on GPUs. To make the combined estimator unbiased and of low variance, the densities of all individual methods are determined and the integrand values are compensated by their sum [J8].

Averaging and Metropolis iteration for PET

High dimensional integrals of PET are estimated by Monte Carlo quadrature. If the sample locations are the same in every iteration step of the ML-EM scheme, then the approximation error will lead to a modified reconstruction result. However, when random estimates are statistically independent in different iteration steps, then the iteration may either diverge or fluctuate around the solution. Our goal is thus to increase the accuracy and the stability of the iterative

solution while keeping the number of random samples and therefore the reconstruction time low. One way to achieve this is to exploit additional samples from previous iteration steps.

I proposed two modifications of the Maximum Likelihood, Expectation Maximization (ML-EM) iteration scheme to reduce the reconstruction error due to the on-the-fly Monte Carlo approximations of forward and back projections with negligible additional cost: Averaging iteration and Metropolis iteration. Averaging iteration averages forward projection estimates during the iteration sequence. Metropolis iteration rejects those forward projection estimates that would compromise the reconstruction and also guarantees the unbiasedness of the tracer density estimate. I demonstrated that these techniques make the estimation unbiased and significantly increase the stability of the iteration sequence. As a result, we can obtain accurate reconstructions with less samples, decreasing the reconstruction time [J5, J6, D10].

Summary of the proposed reconstruction loop

Figure 8.2 shows the ML-EM reconstruction loop including the proposed methods.

Performance summary

Table 8.1 summarizes computational complexities of the proposed methods. In Table 8.2, we show the average running times of the proposed algorithms on a single NVIDIA GeForce 690 GTX GPU for a few representative cases, reconstructing a ring phantom simulation in Mediso's nanoScan-PET/CT geometry. Figure 8.1 shows the running times as the percentages of a complete ML-EM iteration step for a typical parameter setting. FP, BP and SSS stand for geometric forward projection, geometric back projection and single scatter simulation, respectively.

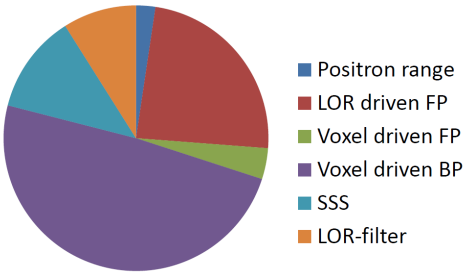


Figure 8.1: Running times as the percentage of a complete iteration for a Ring reconstruction at $N_{\text{voxel}} = 256^3$ voxel resolution and with $1:3$ coincidence mode, when executed on a single NVIDIA GeForce 690 GTX GPU. Parameters of the methods were $m = 3$, $N_{\text{ray}} = 4$, $N_{\text{march}} = 128$, $N_{\text{v}} = 100.000$, $N_{\text{scatter}} = 300$ and $N_{\text{det}} = 64$.

Method	Complexity
Positron range	$\mathcal{O}(mN_{\text{voxel}} \log N_{\text{voxel}})$
LOR driven FP	$\mathcal{O}(N_{\text{LOR}} N_{\text{ray}} N_{\text{march}})$
Voxel driven FP	$\mathcal{O}(N_{\text{Det}} N_{\text{v}})$
Voxel driven BP	$\mathcal{O}(N_{\text{Det}} N_{\text{voxel}})$
SSS	$\mathcal{O}(N_{\text{LOR}} N_{\text{scatter}})$
LOR-filter	$\mathcal{O}(N_{\text{LOR}} N_{\text{det}})$

Table 8.1: Computational complexity of the methods. Note that the voxel driven geometric projection requires atomic writes. The SSS algorithm has an additional computational cost $\mathcal{O}(N_{\text{Det}} N_{\text{scatter}} N_{\text{march}})$ due to the line integral computation between the scattering points and the detectors, however, this is negligible to the overall complexity of the method.

Method	Settings	Time (s)
Positron range	$m = 3$, $N_{\text{voxel}} = 128^3$	0.6
	$m = 3$, $N_{\text{voxel}} = 256^3$	2
LOR driven FP	$N_{\text{ray}} = 1$, $N_{\text{march}} = 64$	3.1
	$N_{\text{ray}} = 4$, $N_{\text{march}} = 128$	20.2
Voxel driven FP	$N_{\text{v}} = 100.000$	3.2
	$N_{\text{v}} = 1.000.000$	22
Voxel driven BP	$N_{\text{voxel}} = 128^3$	6.2
	$N_{\text{voxel}} = 256^3$	41.5
SSS	$N_{\text{scatter}} = 100$	3.4
	$N_{\text{scatter}} = 1000$	31.3
LOR-filter	$N_{\text{det}} = 32$	5.7
	$N_{\text{det}} = 128$	10.5

Table 8.2: Running times in seconds for the ring phantom in Mediso's nanoScan-PET/CT geometry, executed on a single NVIDIA GeForce 690 GTX GPU. The voxel resolution was $N_{\text{voxel}} = 128^3$, except when stated otherwise. Coincidence mode was set to $1 : 3$ in all cases. Note that the voxel centric geometric projection can be executed for about 8–20 times more voxels in a given time budget when it is implemented in an output driven manner (i.e. in back projection), compared to its input driven implementation (i.e. when used as forward projector).

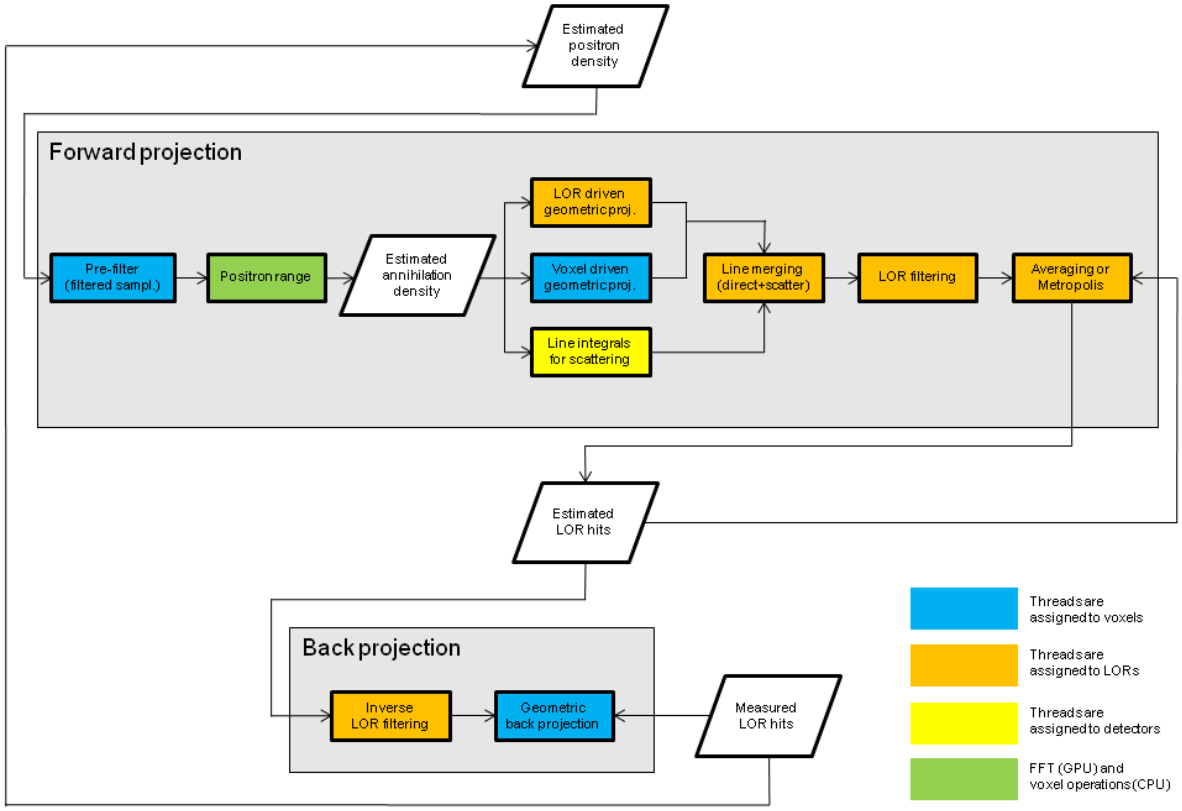


Figure 8.2: Flowchart of the ML-EM reconstruction loop that includes the proposed methods. Unless indicated otherwise, every step is performed on the GPU. Data structures are stored on the GPU.

Own publications

[P1] László Szirmay-Kalos, Milán Magdics, Balázs Tóth. *Method, Computer Readable Medium and System for Tomographic Reconstruction.* US Patent no. PCT/HU2012/000066 (pending), submitted in 2012.

[J1] László Szirmay-Kalos, Balázs Tóth, Milán Magdics, Dávid Légrády, Anton Penzov. *Gamma Photon Transport on the GPU for PET.* LECTURE NOTES IN COMPUTER SCIENCE 5910: pp 435-442., 2010.

[J2] Milán Magdics, László Szirmay-Kalos, Balázs Tóth, Ádám Csendesi, Anton Penzov. *Scatter Estimation for PET Reconstruction.* LECTURE NOTES IN COMPUTER SCIENCE 6046: pp 1-12., 2010.

[J3] Tamás Umenhoffer, László Szécsi, Milán Magdics, Gergely Klár, László Szirmay-Kalos. *Non-photorealistic Rendering for Motion Picture Production.* UPGRADE 10:(6) pp 20-27., 2010.

[J4] László Szirmay-Kalos, Balázs Tóth, Milán Magdics. *Free Path Sampling in High Resolution Inhomogeneous Participating Media.* COMPUTER GRAPHICS FORUM 30:(1) pp 85-97., 2011.

[J5] László Szirmay-Kalos, Milán Magdics, Balázs Tóth, Tamás Bükki. *Averaging and Metropolis Iterations for Positron Emission Tomography.* IEEE TRANSACTIONS ON MEDICAL IMAGING 32:(3) pp. 589-600., 2013.

[J6] Milán Magdics, László Szirmay-Kalos, Balázs Tóth, Anton Penzov. *Analysis and Control of the Accuracy and Convergence of the ML-EM Iteration.* LECTURE NOTES IN COMPUTER SCIENCE 8353 pp. 147-154, 2014.

[J7] Milán Magdics, Rubén Garcia, Mateu Sbert. *Marker-Based Framework for Structural Health Monitoring of Civil Infrastructure.* APPLIED MECHANICS AND MATERIALS 378: pp. 539-545., 2013.

[J8] László Szirmay-Kalos, Milán Magdics, Balázs Tóth, Tamás Bükki. *Multiple Importance Sampling for PET.* IEEE TRANSACTIONS ON MEDICAL IMAGING, to appear.

[J9] Milán Magdics, Ruben Jesus Garcia, Voravika Wattanasoontorn, Mateu Sbert. *Test Installation of a Marker-Based Framework for Structural Health Monitoring of Bridges.* APPLIED MECHANICS AND MATERIALS 477: pp. 813-816., 2014.

[B1] László Szirmay-Kalos, Balázs Tóth, Milán Magdics. *Monte Carlo Photon Transport on the GPU.* GPU Computing Gems (editor: Wen-mei Hwu), MA: Morgan Kaufmann Publishers, Boston, pp 234-255., 2010.

[B2] Milán Magdics, Gergely Klár. *Rule-based Geometry Synthesis in Real-time.* GPU Pro: Advanced Rendering Techniques (editor: Wolfgang Engel), Natick: Kluwer Acad. Publ., pp 41-66., 2010.

[B3] Milán Magdics, László Szirmay-Kalos. *Total Variation Regularization in Maximum Likelihood Estimation.* Optimization in Computer Engineering, Scientific Research Publishing, Delaware, USA, pp 155-168., 2011.

[B4] Voravika Wattanasoontorn, Milán Magdics, Imma Boada, Mateu Sbert. *A Kinect-Based System for Cardiopulmonary Resuscitation Simulation: A Pilot Study.* Serious Games Development and Applications: 4th International Conference, SGDA 2013, Trondheim, Norway. Proceedings. Berlin ; Heidelberg: Springer, pp. 51-63., 2013.

[C1] Milán Magdics. *Real-time Evaluation of L-system Scene Models in Online Multiplayer Games.* MIPRO 2009: 32nd International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, pp 346-351., 2009.

[C2] Milán Magdics. *Real-time Generation of L-system Scene Models for Rendering and Interaction.* Spring Conference on Computer Graphics SCCG 2009: Conference Materials and Posters, Budmerice, Slovakia, pp 77-84., 2009.

[C3] Balázs Tóth, Milán Magdics, László Szirmay-Kalos. *Fast System Matrix Generation on a GPU Cluster.* MIPRO 2009: 32nd International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, pp 319-324., 2009.

[C4] László Szirmay-Kalos, Milán Magdics, Balázs Tóth, Balázs Csébfalvi. *Efficient Free Path Sampling in Inhomogeneous Media.* Eurographics 2010 Posters, Norwood, Sweden, pp 1-2., 2010.

[C5] Milán Magdics, László Szirmay-Kalos, Ákos Szlávecz, Gábor Hesz, Balázs Benyó, Áron Cserkaszky, Judit Lantos, Dávid Légrády, Szabolcs Czifrus, Andás Wirth, Béla Kári, Gergely Patay, Dávid Völgyes, Tamás Bükki, Péter Major, Gábor Németh and Balázs Domonkos. *TeraTomo project: a fully 3D GPU based reconstruction code for exploiting the imaging capability of the NanoPET^{TM}/CT system.* 2010 World Molecular Imaging Congress, Kyoto, Japan, pp 1., 2010.

[C6] Milán Magdics, László Szirmay-Kalos. *Crystal Scattering Simulation for PET on the GPU.* Eurographics 2011 Short papers, Bangor, Great Britain, pp 61-64., 2011.

[C7] Milán Magdics, László Szirmay-Kalos, Balázs Tóth, Dávid Légrády, Áron Cserkaszky, László Balkay, Balázs Domonkos, Dávid Völgyes, Gergely Patay, Péter Major, Judit Lantos, and Tamás Bükki. *Performance Evaluation of Scatter Modeling of the GPU-based "Tera-Tomo" 3D PET Reconstruction.* IEEE Nuclear Science Symposium and Medical Imaging Conference, pp. 4086-4088, 2011.

[C8] Milán Magdics, László Szirmay-Kalos, Ákos Szlávecz, Gábor Hesz, Balázs Benyó, Áron Cserkaszky, Judit Lantos, Dávid Légrády, Szabolcs Czifrus, Andás Wirth, Béla Kári, Gergely Patay, Dávid Völgyes, Tamás Bükki, Péter Major, Gábor Németh and Balázs Domonkos, László Szécsi, Balázs Tóth. *Detector modeling techniques for pre-clinical 3D PET reconstruction on the GPU.* The 11th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine, Potsdam, Germany, pp 375-378., 2011.

[C9] Milán Magdics, László Szirmay-Kalos, Balázs Tóth, Tamás Bükki and Balázs Csébfalvi. *Higher order scattering estimation for PET.* In IEEE Nuclear Science Symposium and Medical Imaging Conference, pp. 2288-2294., 2012.

[C10] Milán Magdics, László Szirmay-Kalos, Balázs Tóth and Tamás Umenhoffer. *Filtered Sampling for PET.* In IEEE Nuclear Science Symposium and Medical Imaging Conference, pp 2509-2514, 2012.

[C11] László Szirmay-Kalos, Milán Magdics, Balázs Tóth, Tamás Umenhoffer, Judit Lantos and Gergely Patay. *Fast Positron Range Calculation in Heterogeneous Media for 3D PET Reconstruction.* In IEEE Nuclear Science Symposium and Medical Imaging Conference, pp. 2150-2155, 2012.

[C12] Rubén Garcia, Milán Magdics, Antonio Rodríguez and Mateu Sbert. *Visual Realism in 3D Serious Games for Learning: A Case Study.* In 2013 International Conference on Information Science and Technology Applications (ICISTA-2013). Macau, China, pp. 128-132., 2013.

[C13] Voravika Wattanasoontorn, Milán Magdics and Mateu Sbert. *Modifying a game interface to take advantage of advanced I/O devices: A case study.* In 15th International Conference on Enterprise Information Systems (ICEIS 2013): Workshop on Interaction Design in Educational Environments. Angers, France, pp 1-9., 2013.

[C14] Antonio Rodríguez, Rubén Garcia, Juan Manuel Garcia, Milán Magdics and Mateu Sbert. *Implementation of a videogame: Legends of Girona.* In Spanish Conference on Informatics (CEDI 2013): Symposium on Entertainment Computing (SEED 2013). Madrid, Spain, pp. 96-107., 2013.

[C15] Milán Magdics, Catherine Sauvaget, Rubén Garcia, and Mateu Sbert. *Post-Processing NPR Effects for Video Games.* In: 12th ACM International Conference on Virtual Reality Continuum and Its Applications in Industry: VRCAI 2013. Hong-Kong, China, New York: ACM, pp. 147-156, 2013.

[D1] Milán Magdics. *Formal Grammar Based Geometry Synthesis on the GPU Using the Geometry Shader.* 7th Conference of Hungarian Association for Image Processing and Pattern Recognition (KEPAF 2009), Budapest, Hungary. pp 1-9., 2009.

[D2] Milán Magdics, Balázs Tóth, Ádám Csendesi. *Iterative 3D Reconstruction with Scatter Compensation for PET-CT on the GPU.* V. Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, pp 159-168., 2010.

[D3] Kristóf Ralovich, Milán Magdics. *Recursive Ray Tracing in Geometry Shader.* V. Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, pp 19-26., 2010..

[D4] Balázs Tóth, Milán Magdics. *Monte Carlo Radiative Transport on the GPU.* V. Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, pp 177-184., 2010.

[D5] Balázs Tóth, Milán Magdics, László Szirmay-Kalos, Anton Penzov. *Detector Modeling with 4D Filtering in PET.* 8th Conference of Hungarian Association for Image Processing and Pattern Recognition (KEPAF 2011), Szeged, Hungary, pp 27-39., 2011.

[D6] Milán Magdics, Balázs Tóth, Balázs Kovács, László Szirmay-Kalos. *Total Variation Regularization in PET Reconstruction.* 8th Conference of Hungarian Association for Image Processing and Pattern Recognition (KEPAF 2011), Szeged, Hungary, pp 40-53., 2011.

[D7] Tamás Umenhoffer, Milán Magdics, Károly Zsolnai. *Procedural Generation of Hand-drawn like Line Art.* 8th Conference of Hungarian Association for Image Processing and Pattern Recognition (KEPAF 2011), Szeged, Hungary, pp 502-514., 2011.

[D8] Balázs Tóth, Milán Magdics, László Szirmay-Kalos. *Többszörös szóródás szimuláció nagyfelbontású voxeltömbbel definiált közegben.* 8th Conference of Hungarian Association for Image Processing and Pattern Recognition (KEPAF 2011), Szeged, Hungary. pp 488-501., 2011.

[D9] Milán Magdics, Balázs Tóth, László Szirmay-Kalos. *Multiple Forward Scattering Computation for Free.* VI. Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, pp. 114-122., 2012.

[D10] Milán Magdics, Balázs Tóth. *Stochastic Iteration in PET Reconstruction.* VI. Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, pp. 132-138., 2012.

Bibliography

[ACLO10] N.N. Agbeko, Ju-Chieh Cheng, R. Laforest, and A. O'Sullivan. Positron range correction in PET using an alternating EM algorithm. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2010 IEEE*, pages 2875–2878, 2010.

[AK06] Adam Alessio and Paul Kinahan. Pet image reconstruction. *Nuclear Medicine*, 1, 2006.

[AM08] A. Alessio and Lawrence MacDonald. Spatially variant positron range modeling derived from CT for PET image reconstruction. In *Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE*, pages 3637–3640, 2008.

[Ass07] National Electrical Manufacturers Association. NEMA Standards Publication NU 2-2007. Technical report, NEMA, Rosslyn, VA, USA, 2007. Performance measurements of positron emission tomographs.

[Ass08] National Electrical Manufacturers Association. NEMA Standards Publication NU 4-2008. Technical report, NEMA, Rosslyn, VA, USA, 2008. Performance measurements for small animal positron emission tomographs.

[AST+10] A.M. Alessio, C.W. Stearns, Shan Tong, S.G. Ross, S. Kohlmyer, A. Ganin, and P.E. Kinahan. Application and evaluation of a measured spatially variant system model for pet image reconstruction. *Medical Imaging, IEEE Transactions on*, 29(3):938–949, 2010.

[BEB+83] M. Bergström, L. Eriksson, C. Bohm, G. Blomqvist, and J. Litton. Correction for scattered radiation in a ring detector positron camera by integral transformation of the projections. *Journal of computer assisted tomography*, 7(1):42–50, February 1983.

[BHS+98] M.J. Berger, J.H. Hubbell, S.M. Seltzer, J. Chang, J.S. Coursey, R. Sukumar, D.S. Zucker, and K. Olsen. Xcom: Photon cross sections database, 1998. NIST Physical Measurement Laboratory, http://www.nist.gov/pml/data/xcom/index.cfm.

[BKL+05] D. Brasse, P.E. Kinahan, C. Lartizien, C. Comtat, M. Casey, and C. Michel. Correction methods for random coincidences in fully 3d whole-body pet: Impact on data and image quality. *J Nucl Med*, 46:859–867, 2005.

[BM94] D L Bailey and S R Meikle. A convolution-subtraction scatter correction method for 3d pet. *Physics in Medicine and Biology*, 39(3):411, 1994.

[BM99] R.D. Badawi and P.K. Marsden. Self-normalization of emission data in 3d pet. *Nuclear Science, IEEE Transactions on*, 46(3):709–712, 1999.

[BMM08] Nicolai Bissantz, B.A. Mair, and A. Munk. A statistical stopping rule for mlem reconstructions in pet. In *Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE*, pages 4198–4200, 2008.

[BR67] R. N. Bracewell and A. C. Riddle. Inversion of fan-beam scans in radio astronomy. *Astronomical Journal*, 150(2):427–434, 1967.

[BRL+03] Bing Bai, A. Ruangma, R. Laforest, Y.-C. Tai, and R.M. Leahy. Positron range modeling for statistical PET image reconstruction. In *Nuclear Science Symposium Conference Record, 2003 IEEE*, volume 4, pages 2501–2505 Vol.4, 2003.

[BS06] Bing Bai and A.M. Smith. Fast 3D Iterative Reconstruction of PET Images Using PC Graphics Hardware. In *Nuclear Science Symposium Conference Record, 2006. IEEE*, volume 5, pages 2787–2790, 2006.

[BTD09] W.C. Barker, S. Thada, and W. Dieckmann. A GPU-accelerated implementation of the MOLAR PET reconstruction package. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, pages 4114–4119, 2009.

[Bur09] Don J. Burdette. A study of the effects of strong magnetic fields on the image resolution of pet scanners, 2009.

[BVVC12] Belzunce, Verrastro, Venialgo, and Cohen. Cuda parallel implementation of image reconstruction algorithm for positron emission tomography. *Open Medical Imaging Journal*, 6:108–118, 2012.

[CGHE+09] J. Cal-Gonzalez, J. L. Herraiz, S. Espana, M. Desco, J.J. Vaquero, and J.M. Udias. Positron range effects in high resolution 3d pet imaging. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, pages 2788–2791, 2009.

[CGN96] M. E. Casey, H. Gadagkar, and Danny Newport. A component based method for normalization in volume PET. In Pierre Grangeat and J. L. Amans, editors, *Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, pages 66–71. Kluwer Academic, 1996.

[CGR12] J. Cabello, J. E. Gillam, and M. Rafecas. High performance 3d pet reconstruction using spherical basis functions on a polar grid. *Journal of Biomedical Imaging*, 2012:5:5–5:5, January 2012.

[Che01] Simon R. Cherry. Fundamentals of positron emission tomography and applications in preclinical drug development. *The Journal of Clinical Pharmacology*, 41(5):482–491, 2001.

[Chr03] Per H. Christensen. Adjoints and importance in rendering: An overview. *IEEE Transactions on Visualization and Computer Graphics*, 09(3):329–340, 2003.

[CK04] Balázs Csébfalvi and József Koloszár. Vector quantization for feature-preserving volume filtering. In *Vision, Modeling, and Visualization Proceedings*, pages 363–370, Stanford, 2004.

[CK07] Mark Colbert and Jaroslav Křivánek. Real-time shading with filtered importance sampling. In *ACM SIGGRAPH 2007 sketches*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.

[CLBT+12] J. Clerk-Lamalice, M. Bergeron, C. Thibaudeau, R. Fontaine, and R. Lecomte. Evaluation of easily implementable inter-crystal scatter recovery schemes in high-resolution pet imaging. In *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2012 IEEE*, pages 2196–2199, 2012.

[CM03] Ken Chidlow and Torsten Möller. Rapid emission tomography reconstruction. In *Proceedings of the 2003 Eurographics/IEEE TVCG Workshop on Volume graphics*, VG '03, pages 15–26, New York, NY, USA, 2003. ACM.

[CMH93] S R Cherry, S R Meikle, and E J Hoffman. Correction and characterization of scattered events in three-dimensional pet using scanners with retractable septa. *J Nucl Med*, 34(4):671–8, 1993.

[CR12] J. Cabello and M. Rafecas. Comparison of basis functions for 3D PET reconstruction using a Monte Carlo system matrix. *Physics in Medicine and Biology*, 57(7):1759–1777, 2012.

[Csé05] B. Csébfalvi. Prefiltered Gaussian reconstruction for high-quality rendering of volumetric data sampled on a body-centered cubic grid. In *Proceedings of IEEE Visualization*, pages 311–318, 2005.

[Csé10] B. Csébfalvi. An evaluation of prefiltered b-spline reconstruction for quasi-interpolation on the body-centered cubic lattice. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):499–512, 2010.

[CWJ+05] K.-S. Chuang, J. Wu, M.-L. Jan, S. Chen, and C.-H. Hsu. Novel scatter correction for three-dimensional positron emission tomography by use of a beam stopper device. *Nuclear Instruments and Methods in Physics Research A*, 551:540–552, October 2005.

[Der86] S. E. Derenzo. Mathematical removal of positron range blurring in high resolution tomography. *IEEE Trans. Nucl. Sci.*, 33:546–549, 1986.

[DFF+07] C. Degenhardt, K. Fiedler, T. Frach, W. Rutten, T. Solf, and A. Thon. Impact of intercrystal crosstalk on depth-of-interaction information in pet detectors. *Nuclear Science, IEEE Transactions on*, 54(3):427–432, 2007.

[DJ09] Balázs Domonkos and Gábor Jakab. A Programming Model for GPU-based Parallel Computing with Scalability and Abstraction. In *Spring Conference on Computer Graphics*, pages 115–122, 2009.

[DJS12] M. Dawood, X. Jiang, and K. Schafers. *Correction Techniques in Emission Tomography.* Series in Medical Physics and Biomedical Engineering. CRC Press, 2012.

[Gai10] Anastasios Gaitanis. Development of stopping rule methods for the mlem and osem algorithms used in pet image reconstruction, November 2010.

[GBB+12] Andrew L Goertzen, Qinan Bao, Mélanie Bergeron, Eric Blankemeyer, Stephan Blinder, Mario Cañadas, Arion F Chatziioannou, Katherine Dinelle, Esmat Elhami, Hans-Sonke Jans, Eduardo Lage, Roger Lecomte, Vesna Sossi, Suleman Surti, Yuan-Chuan Tai, Juan José; Vaquero, Esther Vicente, Darin A Williams, and Richard Laforest. NEMA NU 4-2008 Comparison of Preclinical PET Imaging Systems. *J Nucl Med*, 53(8):1300–1309, 2012.

[GBH70] Richard Gordon, Robert Bender, and Gabor T. Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology*, 29(3):471 – 481, 1970.

[Gil72] Peter Gilbert. Iterative methods for the three-dimensional reconstruction of an object from projections. *Journal of Theoretical Biology*, 36(1):105 – 117, 1972.

[GLRZ93] G. Gindi, M. Lee, A. Rangarajan, and I. G. Zubal. Bayesian reconstruction of functional images using anatomical information as priors. *Medical Imaging, IEEE Transactions on*, 12(4):670–680, December 1993.

[GMDH08] Nicolas Gac, Stphane Mancini, Michel Desvignes, and Dominique Houzet. High speed 3D tomography on CPU, GPU, and FPGA. *EURASIP Journal on Embedded Systems*, 2008. Article ID 930250.

[GO94] A.S. Goggin and J.M. Ollinger. A model for multiple scatters in fully 3d pet. In *Nuclear Science Symposium and Medical Imaging Conference, 1994., 1994 IEEE Conference Record*, volume 4, pages 1609–1613 vol.4, 1994.

[Goi72] M. Goitein. Three-dimensional density reconstruction from a series of two-dimensional projections. *Nuclear Instruments and Methods*, 101(3):509 – 518, 1972.

[Gor74] Richard Gordon. A tutorial on art (algebraic reconstruction techniques). *IEEE Trans. Nucl. Sci. NS-21*, (3):78–93, 1974.

[Gre90a] P. J. Green. Bayesian reconstructions from emission tomography data using a modified EM algorithm. *IEEE Trans. Med. Im.*, 9(1):84–93, 1990.

[Gre90b] Peter J. Green. Bayesian reconstructions from emission tomography data using a modified em algorithm. *IEEE Trans. Med. Imag*, pages 84–93, 1990.

[GSJ+91] S. Grootoonk, T.J. Spinks, T. Jones, C. Michel, and A. Bol. Correction for scatter using a dual energy window technique with a tomograph operated without septa. In *Nuclear Science Symposium and Medical Imaging Conference, 1991., Conference Record of the 1991 IEEE*, pages 1569–1573 vol.3, 1991.

[HCK+07] I. K. Hong, S. T. Chung, H. K. Kim, Y. B. Kim, Y. D. Son, and Z. H. Cho. Ultra fast symmetry and simd-based projection-backprojection (ssp) algorithm for 3-d pet image reconstruction. *IEEE Trans. Med. Imaging*, 26(6):789–803, 2007.

[HDU90] S. F. Haber, S.E. Derenzo, and D. Uber. Application of mathematical removal of positron range blurring in positron emission tomography. *Nuclear Science, IEEE Transactions on*, 37(3):1293–1299, 1990.

[HEG+09] J.L. Herraiz, S. Espaa, S. Garcia, R. Cabido, A.S. Montemayor, M. Desco, J.J. Vaquero, and J.M. Udias. GPU acceleration of a fully 3D iterative reconstruction software for PET using CUDA. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, pages 4064–4067, 2009.

[HEV+06] J L Herraiz, S Espaa, J J Vaquero, M Desco, and J M Udas. FIRST: Fast Iterative Reconstruction Software for (PET) tomography. *Physics in Medicine and Biology*, 51(18):4547, 2006.

[HHPK81] Edward J Hoffman, Sung-Cheng Huang, Michael E Phelps, and David E Kuhl. Quantitation in positron emission computed tomography: 4. effect of accidental coincidences. *Journal of computer assisted tomography*, 5(3):391–400, 1981.

[HL94] H.M. Hudson and R.S. Larkin. Accelerated image reconstruction using ordered subsets of projection data. *Medical Imaging, IEEE Transactions on*, 13(4):601–609, 1994.

[HTC+05] B. Hesse, K. Tgil, A. Cuocolo, C. Anagnostopoulos, M. Bardis, J. Bax, F. Bengel, E. Busemann Sokole, G. Davies, M. Dondi, L. Edenbrandt, P. Franken, A. Kjaer, J. Knuuti, M. Lassmann, M. Ljungberg, C. Marcassa, P.Y. Marie, F. McKiddie, M. OConnor, E. Prvulovich, R. Underwood, and B. Eck-Smit. Eanm/esc procedural guidelines for myocardial perfusion imaging in nuclear cardiology. *European Journal of Nuclear Medicine and Molecular Imaging*, 32(7):855–897, 2005.

[IMS07] M. Iatrou, R.M. Manjeshwar, and C.W. Stearns. Comparison of two 3d implementations of tof scatter estimation in 3d pet. In *Nuclear Science Symposium Conference Record, 2007. NSS '07. IEEE*, volume 5, pages 3474–3477, 2007.

[JDB08] Gábor Jakab, Balázs Domonkos, and Tamás Bükki. Practical implementation of helical cone-beam CT imaging using multiple GPUs. NVISION, Poster, San Jose, CA, United States, August 2008.

[Jea04] S. Jan and et al. GATE: A simulation toolkit for PET and SPECT. *Physics in Medicine and Biology*, 49(19):4543–4561, 2004. http://www.opengatecollaboration.org.

[JKB+12] Abhinav K. Jha, Matthew A. Kupinski, Harrison H. Barrett, Eric Clarkson, and John H. Hartman. Three-dimensional neumann-series approach to model light transport in nonuniform media. *J. Opt. Soc. Am. A*, 29(9):1885–1899, Sep 2012.

[Jos82] Peter M. Joseph. An improved algorithm for reprojecting rays through pixel images. *IEEE Transactions on Medical Imaging*, 1(3):192–196, nov. 1982.

[JRM+09] G. Jakab, A. Racz, P. Major, T. Bukki, and G. Nemeth. Fully GPU based real time corrections and reconstruction for cone beam micro CT. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, pages 4068–4071, 2009.

[JSC+97] C. A. Johnson, J. Seidel, R. E. Carson, W. R. Gandler, A. Sofer, M. V. Green, and M. E. Daube-Witherspoon. Evaluation of 3D reconstruction algorithms for a small animal PET camera. *IEEE Transactions on Nuclear Science*, 44:1303–1308, June 1997.

[JSK13] Gábor Jakab and László Szirmay-Kalos. Hybrid Monte Carlo CT simulation on GPU. *Lecture Notes of Computer Science*, 2013.

[KCC+09] Sean L Kitson, Vincenzo Cuccurullo, Andrea Ciarmiello, Diana Salvo, and Luigi Mansi. Clinical Applications of Positron Emission Tomography (PET) Imaging in Medicine: Oncology, Brain Diseases and Cardiology. *Current Radiopharmaceuticals*, 2(4):224–253, 2009.

[Kip02] M. S. Kipper. Clinical applications of positron emission tomography. *Applied Radiology*, 31(11):41–48, 2002.

[KJY03] J. Koloszár and Y. Jae-Young. Accelerating virtual endoscopy. *Journal of WSCG*, 11(2), 2003.

[KKB07] M. Kachelries, M. Knaup, and O. Bockenbach. Hyperfast parallel-beam and cone-beam backprojection using the CELL general purpose hardware. *Medical Physics*, 34(4):1474–1486, 2007.

[Klá08] Gergely Klár. Level of detail flow simulation. In *Eurographics 2008 - Short Papers*, pages 127–130, 2008.

[Kol08] József Koloszár. *Virtual Colonoscopy.* PhD thesis, BME VIK, Budapest, 2008.

[KSKAC02] Csaba Kelemen, László Szirmay-Kalos, György Antal, and Ferenc Csonka. A simple and robust mutation strategy for the Metropolis light transport algorithm. *Comput. Graph. Forum*, 21(3):531–540, 2002.

[KSKTJ06] József Koloszár, László Szirmay-Kalos, Zsolt Tarján, and Dávid Jocha. Shape based computer aided diagnosis and automated navigation in virtual colonoscopy. In *Spring Conference on Computer Graphics*, pages 113–119, Budmerice, 2006.

[KY11a] Kyung Sang Kim and Jong Chul Ye. Fully 3d iterative scatter-corrected osem for hrrt pet using a gpu. *Physics in Medicine and Biology*, 56(15):4991, 2011.

[KY11b] Kyung Sang Kim and Jong-Chul Ye. Ultra-fast hybrid cpu-gpu monte carlo simulation for scatter correction in 3d pets. In *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2011 IEEE*, pages 2749–2752, 2011.

[Lak75] A.V. Lakshminarayanan. *Reconstruction from divergent ray data.* Technical report. State University of New York, Department of Computer Science, 1975.

[LCLC10] J. Lantos, Sz. Czifrus, D. Légrády, and A. Cserkaszky. Detector response function of the NanoPET/CT system. In *IEEE Nuclear Science Symposium and Medical Imaging Conference.*, pages 3641–3643, 2010.

[LCM+02] Miriam Leeser, Srdjan Coric, Eric Miller, Haiqian Yu, and Marc Trepanier. Parallel-beam backprojection: an FPGA implementation optimized for medical imaging. In *Proc. Tenth Int. Symposium on FPGA*, pages 217–226, 2002.

[Lew92] R M Lewitt. Alternatives to voxels for image representation in iterative reconstruction algorithms. *Physics in Medicine and Biology*, 37(3):705, 1992.

[LH99] C. S. Levin and E. J. Hoffmann. Calculation of positron range and its effect on the fundamental limit of positron emission tomography system spatial resolution. *Phys. Med. Biol.*, 44:781–799, 1999.

[LKF+03] Kisung Lee, P.E. Kinahan, J.A. Fessler, R.S. Miyaoka, and T.K. Lewellen. Pragmatic image reconstruction for the mices fully-3d mouse imaging pet scanner. In *Nuclear Science Symposium Conference Record, 2003 IEEE*, volume 4, pages 2566–2570 Vol.4, 2003.

[LSK10] L. Szécsi L. Szirmay-Kalos. General purpose computing on graphics processing units. In A. Iványi, editor, *Algorithms of Informatics*, pages 1451–1495. MondArt Kiadó, Budapest, 2010. http://sirkan.iit.bme.hu/szirmay/gpgpu.pdf.

[LT00] D S Lalush and B M Tsui. Performance of ordered-subset reconstruction algorithms under conditions of extreme attenuation and truncation in myocardial spect. *J Nucl Med*, 41(4):737–44, 2000.

[MB04] Bruno De Man and Samit Basu. Distance-driven projection and backprojection in three dimensions. *Physics in Medicine and Biology*, 49(11):2463, 2004.

[MDB+08] S. Moehrs, M. Defrise, N. Belcari, A. D. Guerra, A. Bartoli, S. Fabbri, and G. Zanetti. Multi-ray-based system matrix generation for 3D PET reconstruction. *Physics in Medicine and Biology*, 53:6925–6945, 2008.

[Med10a] Mediso. Anyscan PET/CT, 2010. http://www.mediso.com/products.php?fid=1,9&pid=73.

[Med10b] Mediso. Nanoscan-PET/CT, 2010. http://www.mediso.com/products.php?fid=2,11&pid=86.

[ML96] S Matej and R M Lewitt. Practical considerations for 3-d image reconstruction using spherically symmetric volume elements. *IEEE Trans Med Imaging*, 15(1):68–78, 1996.

[MLCH96] E.U. Mumcuoglu, R.M. Leahy, S.R. Cherry, and E. Hoffman. Accurate geometric and physical response modelling for statistical image reconstruction in high resolution pet. In *Nuclear Science Symposium, 1996. Conference Record., 1996 IEEE*, volume 3, pages 1569–1573 vol.3, 1996.

[MR06] C. Mora and M. Rafecas. Polar pixels for high resolution small animal pet. In *Nuclear Science Symposium Conference Record, 2006. IEEE*, volume 5, pages 2812–2817, 2006.

[MRPG12] Paul M. Matthews, Eugenii A. Rabiner, Jan Passchier, and Roger N. Gunn. Positron emission tomography molecular imaging for drug development. *British Journal of Clinical Pharmacology*, 73(2):175–186, 2012.

[MRR+53] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[NVI07] NVIDIA. http://developer.nvidia.com/cuda. In *The CUDA Homepage*, 2007.

[NVI13] NVIDIA Corporation. *NVIDIA CUDA C Programming Guide*, June 2013.

[OJ93] J.M. Ollinger and G.C. Johns. Model-based scatter correction for fully 3d pet. In *Nuclear Science Symposium and Medical Imaging Conference, 1993., 1993 IEEE Conference Record.*, pages 1264–1268, 1993.

[Oll96] John M Ollinger. Model-based scatter correction for fully 3d pet. *Physics in Medicine and Biology*, 41(1):153, 1996.

[Oll97] J.M. Ollinger. Analytic correction for scatter in fully 3d pet: statistical issues. In *Nuclear Science Symposium, 1997. IEEE*, volume 2, pages 1386–1389 vol.2, 1997.

[PB92] M. R. Palmer and G. L. Brownell. Annihilation density distribution calculations for medically important positron emitters. *IEEE Trans. Med. Imag.*, 11:373–378, 1992.

[PBE01] M. Persson, D. Bone, and H. Elmqvist. Total variation norm for threedimensional iterative reconstruction in limited view angle tomography. *Physics in Medicine and Biology*, 46(3):853–866, 2001.

[PKMC06] V.Y. Panin, F. Kehren, C. Michel, and M. Casey. Fully 3-D PET reconstruction with system matrix derived from point source measurements. *IEEE Transactions on Medical Imaging*, 25(7):907 –921, july 2006.

[PL09] G. Pratx and C. S. Levin. Bayesian reconstruction of photon interaction sequences for high-resolution pet detectors, (selected as feature article of the month, american institute of physics, august 2009). *Physics in Medicine and Biology*, 54(17):5073–5094, 2009.

[PL11] G. Pratx and C. S. Levin. Online detector response calculations for high-resolution pet image reconstruction. *Physics in Medicine and Biology*, 56(13):4023–4040, 2011.

[PX11] Guillem Pratx and Lei Xing. GPU computing in medical physics: A review. *Medical Physics*, 38:2685, 2011.

[QL06] J. Qi and R. Leahy. Iterative reconstruction techniques in emission computed tomography. *Physics in Medicine and Biology*, 51(15):541–578, 2006.

[QLC+98] J. Qi, R. M. Leahy, S. R. Cherry, A. Chatziioannou, and T. H. Farquhar. High-resolution 3D Bayesian image reconstruction using the microPET small-animal scanner. *Physics in Medicine and Biology*, 43(4):1001, 1998.

[QMT10] Hua Qian, R. Manjeshwar, and K. Thielemans. A comparative study of multiple scatter estimations in 3d pet. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2010 IEEE*, pages 2700–2702, 2010.

[Rad17] J. Radon. Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Akad. Wiss.*, 69:262–277, 1917.

[Rad86] J. Radon. On the Determination of Functions from Their Integral Values along Certain Manifolds. *IEEE Transactions on Medical Imaging*, 5(4):170–176, 1986.

[RLCC98] Jinyi Qi Richard, Richard M Leahy, Simon R Cherry, and Arion Chatziioannou. High-resolution 3d bayesian image reconstruction using the micropet small-animal scanner. *Med. Biol*, 43:1001–1013, 1998.

[RLT+08] A. Rahmim, M.A. Lodge, J. Tang, S. Lashkari, and M.R. Ay. Analytic system matrix resolution modeling in PET: An application to Rb-82 cardiac imaging. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 1307–1310, 2008.

[RMD+04] M. Rafecas, B. Mosler, M. Dietz, M. Pogl, A. Stamatakis, D.P. McElroy, and S.I. Ziegler. Use of a monte carlo-based probability matrix for 3-d iterative reconstruction of madpet-ii data. *Nuclear Science, IEEE Transactions on*, 51(5):2597–2605, 2004.

[Ros84] A. Rosenfeld. *Multiresolution Image Processing and Analysis*. Springer series in information sciences. Springer-Verlag, 1984.

[RSC+06] Andrew J Reader, Florent C Sureau, Claude Comtat, Rgine Trbossen, and Irne Buvat. Joint estimation of dynamic pet images and temporal basis functions using fully 4d ml-em. *Physics in Medicine and Biology*, 51(21):5455, 2006.

[RZ07] A. J. Reader and H. Zaidi. Advances in PET image reconstruction. *PET Clinics*, 2(2):173–190, 2007.

[SBM+11] S Stute, D Benoit, A Martineau, N S Rehfeld, and I Buvat. A method for accurate modelling of the crystal response function at a crystal sub-level applied to pet reconstruction. *Physics in Medicine and Biology*, 56(3):793, 2011.

[Ser06] Alain Seret. The number of subsets required for osem reconstruction in nuclear cardiology. *European Journal of Nuclear Medicine and Molecular Imaging*, 33(2):231–231, 2006.

[SFK94] Lingxiong Shao, R. Freifelder, and J.S. Karp. Triple energy window scatter correction technique in pet. *Medical Imaging, IEEE Transactions on*, 13(4):641–648, 1994.

[SH05] C. Sigg and M. Hadwiger. Fast third-order texture filtering. In *GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation*, pages 313–329. Matt Pharr(ed.), Addison-Wesley, 2005.

[Sid85] R. L. Siddon. Fast calculation of the exact radiological path for a three-dimensional ct array. *Medical Physics*, 12(2):252–257, 1985.

[SK91] Lingxiong Shao and J.S. Karp. Cross-plane scattering correction-point source deconvolution in pet. *Medical Imaging, IEEE Transactions on*, 10(3):234–239, 1991.

[SK99] L. Szirmay-Kalos. Stochastic iteration for non-diffuse global illumination. *Computer Graphics Forum*, 18(3):233–244, 1999.

[SK00] L. Szirmay-Kalos. *Photorealistic Image Synthesis with Ray-Bundles*. Hungarian Academy of Sciences, D.Sc. Dissertation, Budapest, 2000. http://www.iit.bme.hu/~szirmay/ Thesis-SzKL.htm.

[SK08] L. Szirmay-Kalos. *Monte-Carlo Methods in Global Illumination — Photo-realistic Rendering with Randomization*. VDM, Verlag Dr. Müller, Saarbrücken, 2008.

[SKS09] L. Szirmay-Kalos and L. Szécsi. Deterministic importance sampling with error diffusion. *Computer Graphics Forum (EG Symposium on Rendering)*, 28(4):1056–1064, 2009.

[SKSS08] L. Szirmay-Kalos, L. Szécsi, and M. Sbert. *GPU-Based Techniques for Global Illumination Effects*. Morgan and Claypool Publishers, San Rafael, USA, 2008.

[Sob91] I. Sobol. *Die Monte-Carlo Methode*. Deutscher Verlag der Wissenschaften, 1991.

[SPC+00] V.V. Selivanov, Y. Picard, J. Cadorette, S. Rodrigue, and R. Lecomte. Detector response models for statistical iterative image reconstruction in high resolution pet. *Nuclear Science, IEEE Transactions on*, 47(3):1168–1175, 2000.

[SRA+02] D.W. Shattuck, J. Rapela, E. Asma, A. Chatzioannou, J. Qi, and R.M. Leahy. Internet2-based 3D PET image reconstruction using a PC cluster. *Physics in Medicine and Biology*, 47:2785–2795, 2002.

[SSD+03] D Strul, R B Slates, M Dahlbom, S R Cherry, and P K Marsden. An improved analytical detector response function model for multilayer small-diameter pet scanners. *Physics in Medicine and Biology*, 48(8):979, 2003.

[SSG12] Krishnendu Saha, KennethJ. Straus, and StephenJ. Glick. Iterative reconstruction with monte carlo based system matrix for dedicated breast pet. In AndrewD.A. Maidment, PredragR. Bakic, and Sara Gavenonis, editors, *Breast Imaging*, volume 7361 of *Lecture Notes in Computer Science*, pages 157–164. Springer Berlin Heidelberg, 2012.

[SSKEP13] L. Szécsi, L. Szirmay-Kalos, Gy. Egri, and G. Patay. Binned Time-of-Flight Positron Emission Tomography. In *Proceedings of KÉPAF 2013*, pages 340–350, Bakonybél, Hungary, January 2013.

[SSSK04] L. Szécsi, M. Sbert, and L. Szirmay-Kalos. Combined correlated and importance sampling in direct light source computation and environment mapping. *Computer Graphics Forum (Eurographics 04)*, 23(3):585–604, 2004.

[SV82] L. Shepp and Y. Vardi. Maximum likelihood reconstruction for emission tomography. *IEEE Trans. Med. Imaging*, 1:113–122, 1982.

[TAR+05] C. Tsoumpas, P. Aguiar, D. Ros, N. Dikaios, and K. Thielemans. Scatter simulation including double scatter. In *Nuclear Science Symposium Conference Record, 2005 IEEE*, volume 3, pages 5 pp.–1619, 2005.

[Tho88] C.J. Thompson. The effect of collimation on scatter fraction in multi-slice pet. *Nuclear Science, IEEE Transactions on*, 35(1):598–602, 1988.

[TM98] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 839–, Washington, DC, USA, 1998. IEEE Computer Society.

[TQ09] Michel S Tohme and Jinyi Qi. Iterative image reconstruction for positron emission tomography based on a detector response function estimated from point source measurements. *Physics in Medicine and Biology*, 54(12):3709, 2009.

[TU09] Balázs Tóth and Tamás Umenhoffer. Real-time Volumetric Lighting in Participating Media. pages 57–60, Munich, Germany, 2009. Eurographics Association.

[VDdW+01] S. Vandenberghe, Y. D'Asseler, R. Van de Walle, T. Kauppinen, M. Koole, L. Bouwens, K. Van Laere, I. Lemahieu, and R.A. Dierckx. Iterative reconstruction algorithms in nuclear medicine. *Computerized Medical Imaging and Graphics*, 25(2):105 – 111, 2001.

[VG95] E. Veach and L. Guibas. Optimally combining sampling techniques for Monte Carlo rendering. In *Computer Graphics Proceedings, Annual Conference Series, 1995 (ACM SIGGRAPH '95 Proceedings)*, pages 419–428, 1995.

[VL87] Eugene Veklerov and Jorge Llacer. Stopping rule for the mle algorithm based on statistical hypothesis testing. *Medical Imaging, IEEE Transactions on*, 6(4):313–319, 1987.

[Wat00] C.C. Watson. New, faster, image-based scatter correction for 3d pet. *Nuclear Science, IEEE Transactions on*, 47(4):1587–1594, 2000.

[Wat07] C.C. Watson. Extension of single scatter simulation to scatter correction of time-of-flight pet. *Nuclear Science, IEEE Transactions on*, 54(5):1679–1686, 2007.

[WBD+02] Alexander Werling, Olaf Bublitz, Josef Doll, Lars-Eric Adam, and Gunnar Brix. Fast implementation of the single scatter simulation algorithm and its use in iterative image reconstruction of pet data. *Physics in Medicine and Biology*, 47(16):2947, 2002.

[WCK+09] A. Wirth, A. Cserkaszky, B. Kári, D. Legrády, S. Fehér, S. Czifrus, and B. Domonkos. Implementation of 3D Monte Carlo PET reconstruction algorithm on GPU. In *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, pages 4106–4109, 24 2009-nov. 1 2009.

[WCMB04] C.C. Watson, M.E. Casey, C. Michel, and B. Bendriem. Advances in scatter correction for 3d pet/ct. In *Nuclear Science Symposium Conference Record, 2004 IEEE*, volume 5, pages 3008–3012, 2004.

[WNC96] C.C. Watson, D. Newport, and M.E. Casey. *Three-Dimensional Image Reconstruction in Radiation and Nuclear Medicine*, chapter A single scattering simulation technique for scatter correction in 3D PET, pages 255–268. Kluwer Academic Publishers, 1996.

[WSK06] M.E. Werner, S. Surti, and J.S. Karp. Implementation and evaluation of a 3d pet single scatter simulation with tof modeling. In *Nuclear Science Symposium Conference Record, 2006. IEEE*, volume 3, pages 1768–1773, 2006.

[XM07] F. Xu and K. Mueller. Real-time 3d computed tomographic reconstruction using commodity graphics hardware. *Physics in Medicine and Biology*, pages 3405–3419, 2007.

[Yan08] C. N. Yang. The Klein-Nishina formula & quantum electrodynamics. *Lect. Notes Phys.*, 746:393–397, 2008.

[YHO+05] Taiga Yamaya, Naoki Hagiwara, Takashi Obi, Masahiro Yamaguchi, Nagaaki Ohyama, Keishi Kitamura, Tomoyuki Hasegawa, Hideaki Haneishi, Eiji Yoshida, Naoko Inadama, and Hideo Murayama. Transaxial system models for jPET-D4 image reconstruction. *Physics in Medicine and Biology*, 50(22):5339, 2005.

[YYO+08] T. Yamaya, E. Yoshida, T. Obi, H. Ito, K. Yoshikawa, and H. Murayama. First human brain imaging by the jpet-d4 prototype with a pre-computed system matrix. *Nuclear Science, IEEE Transactions on*, 55(5):2482–2492, 2008.

[ZG00] G. Zeng and G. Gullberg. Unmatched projector/backprojector pairs in an iterative reconstruction algorithm. *IEEE Transactions on Medical Imaging*, 19(5):548–555, 2000.

[ZM07] Habib Zaidi and Marie-Louise Montandon. Scatter compensation techniques in pet. *PET Clinics*, 2(2):219 – 234, 2007. PET Instrumentation and Quantification.

[ZSD+08] Stephen Eric Zingelewicz, Austars Raymond Schnore, Walter Vincent Dixon, Samit Kumar Basu, Bruno De Man, and William D. Smith. Method and apparatus for reconstruction of 3d image volumes from projection images. *US patent No. 0095300 A1*, 2008.

Index

Nomenclature

ϵ_0	relative photon energy
$\hat{\mathbf{T}}$	attenuation matrix
$\mathbf{d}$	detector
$\mathbf{A}$	system matrix
$\mathbf{G}$	geometric projection matrix
$\mathbf{L}$	detector model matrix
$\mathbf{P}$	positron range matrix
$\mathbf{S}$	scattering matrix
$\mathbf{x}$	voxelized positron emission density
$\mathbf{x}^a$	estimated annihilation density
$\mathbf{y}$	measured coincidences
$\mathcal{D}$	total surface of detectors
$\mathcal{L}$	likelihood function
$\mathcal{T}$	scanner sensitivity
$\mathcal{X}$	total activity
$\mathscr{F}$	Fourier transform
$\mu_{\mathbf{d}}$	detector sensitivity
$\nu(\vec{\omega})$	detection probability
Ω	domain of directions
σ_a	absorption cross section
σ_s	scattering cross section
σ_t	extinction parameter
θ	angle
$\tilde{\mathbf{y}}$	estimated coincidences
$\vec{\omega}$	direction vector
$\vec{s}$	scattering point
$\vec{v}$	point in the volume of interest
$\vec{z}$	detector surface point
$\xi_m(\vec{v})$	material indicator function
$A_\epsilon(\vec{z}_1, \vec{z}_2)$	attenuation factor
$b_V(\vec{v})$	basis function
$B_\epsilon(\vec{z}_1, \vec{z}_2)$	absorption factor
$d\omega$	differential solid angle
E_t	transport function
$G(\vec{z}_1, \vec{z}_2)$	geometry factor
L	radiant intensity
L^e	emission density
$m(\vec{v})$	material index
N_{Det}	total number of detectors
N_{det}	number of detector samples
N_{LOR}	total number of LORs
N_{march}	step number of raymarching
N_{path}	number of path samples
N_{PT}	number of simulated photon paths
N_{ray}	number of ray samples
N_{scatter}	number of scattering point samples
N_{voxel}	total number of voxels
N_{v}	number of voxel samples
$P(\vec{\omega}', \vec{\omega})$	phase function
$p_{\mathbf{i}\to\mathbf{d}}(\vec{\omega})$	crystal transport probability
$T_\epsilon(\vec{z}_1, \vec{z}_2)$	out-scattering factor
$x(\vec{v})$	positron emission density
$x^a(\vec{v})$	annihilation density
$\mathcal{V}$	volume of interest