Multidimensional *B*-spline parameterization of the detection probability of PET systems to improve the efficiency of Monte Carlo simulations

IOP Publishing Physics in Medicine and Biology

Phys. Med. Biol. **55** (2010) 3339–3361 doi:10.1088/0031-9155/55/12/006

Multidimensional *B*-spline parameterization of the detection probability of PET systems to improve the efficiency of Monte Carlo simulations

Niklas S Rehfeld[1], **Sébastien Vauclin**[2], **Simon Stute**[1] **and Irène Buvat**[1]

[1] IMNC-UMR 8165 CNRS, Paris 7 University, Paris 11 University, 15 rue Georges Clémenceau, 91406 Orsay Cedex, France
[2] LITIS EA Laboratory, University of Rouen, Rouen, France

Received 1 October 2009, in final form 20 April 2010
Published 26 May 2010
Online at stacks.iop.org/PMB/55/3339

Abstract

Accurate modeling of system response and scatter distribution is crucial for image reconstruction in emission tomography. Monte Carlo simulations are very well suited to calculate these quantities. However, Monte Carlo simulations are also slow and many simulated counts are needed to provide a sufficiently exact estimate of the detection probabilities. In order to overcome these problems, we propose to split the simulation into two parts, the detection system and the object to be imaged (the patient). A so-called 'virtual boundary' that separates these two parts is introduced. Within the patient, particles are simulated conventionally. Whenever a photon reaches the virtual boundary, its detection probability is calculated analytically by evaluating a multi-dimensional *B*-spline that depends on the photon position, direction and energy. The unknown *B*-spline knot values that define this *B*-spline are fixed by a prior 'pre-' simulation that needs to be run once for each scanner type. After this pre-simulation, the *B*-spline model can be used in any subsequent simulation with different patients. We show that this approach yields accurate results when simulating the Biograph 16 HiREZ PET scanner with Geant4 Application for Emission Tomography (GATE). The execution time is reduced by a factor of about 22× (scanner with voxelized phantom) to 30× (empty scanner) with respect to conventional GATE simulations of same statistical uncertainty. The pre-simulation and calculation of the *B*-spline knots values could be performed within half a day on a medium-sized cluster.

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Geant4 Application for Emission Tomography (GATE) (Jan *et al* 2004) is a widespread, well validated and very versatile application for Monte Carlo simulations in emission tomography.

0031-9155/10/123339+23$30.00 © 2010 Institute of Physics and Engineering in Medicine Printed in the UK

It allows detailed simulations of positron emission tomography (PET) or single photon emission computed tomography (SPECT) scanners using the general purpose Monte Carlo code Geant4 (Agostinelli *et al* 2003, Allison *et al* 2006). This includes the exact positioning of the constituting parts (like crystals, blocks, supporting structure, bed), correct physical interactions, and the modeling of the signal processing of the detected events. GATE is also capable to simulate the propagation of light (low energy photons) in the crystals (Takoudis *et al* 2008).

Iterative PET or SPECT image reconstruction algorithms as opposed to analytical algorithms are nowadays the methods of choice due to their superior statistical performance and flexibility (Qi and Leahy 2006). The system matrix that plays a central role in every iterative reconstruction algorithm describes the system response of the detection system, i.e., the probability that a positron emitted at a certain location in the patient be detected by a given detection element. Any improvement that can be achieved with respect to the quality of this matrix will therefore directly translate into more quantitative reconstructed images (Qi and Huesman 2005). It has been shown that Monte Carlo simulations could be used for accurate system matrix calculation (Rafecas *et al* 2004, El Bitar *et al* 2006, Lazaro *et al* 2004, 2005).

A major drawback of Monte Carlo simulations is their slow performance. Usually, realistic simulation tools like GATE tally integer counts to mimic correctly the statistics and the signal processing. For matrix simulations non-integer detection probabilities need to be computed. It would be therefore desirable to replace the binary response of the detectors by the detection probability.

In this work, we propose to accurately model the detector response function by a multidimensional *B*-spline. The *B*-spline weights are fixed using a Monte Carlo simulation that precedes the actual Monte Carlo simulation. This ‘pre-simulation’ has to be performed only once for a given scanner type. Once the detection probability function of the detectors is parameterized, it could be used for any subsequent Monte Carlo simulation involving the same scanner type. It could then be useful for the efficient calculation of model-based scatter correction or an accurate patient-dependent system matrix to replace the detailed Monte Carlo propagation within the scanner by the parameterized detector response and to use the full Monte Carlo tracking only to account for the particle tracking within the patient.

In the past, approaches to model the detection response have been proposed to accelerate Monte Carlo simulations of SPECT. These approaches are based on a tabulated angular response function (Song *et al* 2005) or on convolution-based forced detection (de Jong *et al* 2001, Beekman *et al* 2002, de Beenhouwer *et al* 2008). The presented general *B*-spline method is applied to PET Monte Carlo simulations.

2. Theory

2.1. Principle

The proposed method assumes that the simulation can be divided into two parts. In the first part the particles are started in the phantom/simulated patient and tracked in a conventional manner using a Monte Carlo code (in this example GATE). Eventually, in the conventional simulation, the particles will either be detected in the detection system or be discarded, because they are absorbed, or their energy drops below a certain energy threshold, or they leave the simulated geometry. In the presented method a virtual border between the interior of the gantry and the detection system is introduced. A particle that crosses this border leaves the Monte Carlo simulation and the probability of detection is then calculated analytically. The analytical

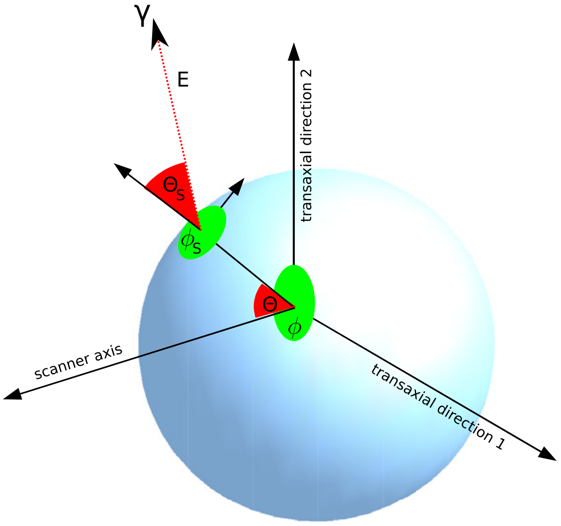


Figure 1. The variables used to describe the photon state on the virtual boundary.

description is itself based upon an accurate Monte Carlo simulation that was performed in a so-called ‘pre-simulation’.

Our analytical model relies on the following assumptions:

(i) Other particles (e.g., electrons, positrons) than photons that are crossing the virtual border do not reach the detectors.

(ii) Photons cross the border two times at most. This means that a photon cannot leave, re-enter, and leave again. The interior part of the geometry does not influence the detection probability once the photon crossed the virtual border.

(iii) The detection probability itself is not flux dependent, i.e. dead time effects are not taken into account.

(iv) Furthermore, although the detection probability can change non-continuously from one detector to the next, this probability will change smoothly for a given detector: small changes in the photon state at the virtual border will only lead to small changes in detection probability.

(v) If many detectors are involved (for example several tens of thousands as in PET), it is assumed that for a photon in a given state the detection probability is only non-negligible for a small subset of the detectors.

In our approach we used a spherical virtual border of fixed radius. When neglecting spin, the photon state **x** on this border can be represented by five variables (see figure 1):

$$\mathbb{R}^N \ni \mathbf{x} = \begin{pmatrix} \text{position inclination angle } \theta \\ \text{position azimuthal angle } \phi \\ \text{momentum direction angle of incidence } \theta_\mathrm{S} \\ \text{momentum direction azimuthal angle } \phi_\mathrm{S} \\ \text{energy } E \end{pmatrix}$$

$$N \equiv \text{ number of coordinates (here } N = 5)$$

Here θ is the angle between the scanner axis (z-axis) and the line connecting the scanner center (origin) with the point where the photon hits the virtual border. The angle ϕ defines the position of the latter point in the transaxial plane. These angles are global coordinates. The momentum direction angles are local coordinates. Angle θ_S is the angle of incidence with respect to the virtual border and ϕ_S is the angle in the tangential plane.

We assumed that photons cannot be detected when they do not reach the ring with detectors. Photons with a position located outside this specific area of the virtual border were thus discarded. In order to improve statistics, rotational symmetry as well as global axial symmetry and azimuthal intra-block symmetry can be used (for example, in total 192 fold for the Biograph HiREZ). In the pre-simulation, photons were therefore only directed such that they reach a small non-redundant subarea of the virtual border.

For each of the detectors (for example, $M = 24\,366$ detectors in the case of the PET/CT scanner Biograph HiREZ) a five-dimensional B-spline with the state $\mathbf{x}$ as argument is used to described the detection probability. It can be expected that for a fixed photon state $\mathbf{x}$, the B-spline is only non-negligible for a small subset of the detectors ($\ll$24 366 for the Biograph HiREZ).

For a given detector, the B-spline $f(\mathbf{x})$ describing the probability of detection for a photon in state $\mathbf{x}$ at the border is defined for an interval $\mathbf{x} \in I = [\mathbf{x}^{\min}, \mathbf{x}^{\max}]$ using a uniform spacing $\mathbf{s} \in \mathbb{R}^N$ of the B-spline knots for each of the N dimensions and $x_i^{\max} = x_i^{\min} + q_i s_i$ with $q_i \in \mathbb{N}$:

$$f(\mathbf{x}) = \sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}})\varphi_{\mathbf{k}+\hat{\mathbf{k}}}. \tag{1}$$

Here $B_{\mathbf{k}}(\widetilde{\mathbf{x}})$ is the $\mathbf{k}$th tensor B-spline basis function depending on the running index vector $\mathbf{k}$ and $\widetilde{\mathbf{x}}$ (see (2) and (3)).

$$B_{\mathbf{k}}(\widetilde{\mathbf{x}}) \equiv \prod_{i=0}^{N-1} B_{k_i}^{(n_i)}(\widetilde{x}_i) \tag{2}$$

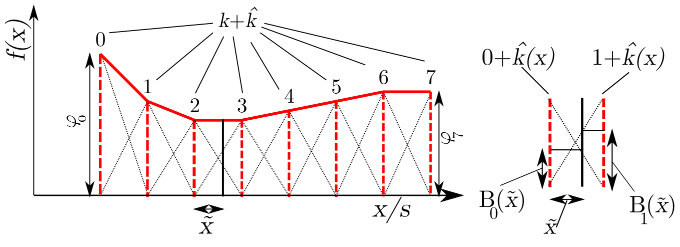
$$\begin{aligned} B_{k_i}^{(n_i)}(\cdots) &\equiv k_i\text{th } B\text{-spline basis function of order } n_i \\ \hat{k}_i &= \left\lfloor \frac{x_i - x_i^{\min}}{s_i} \right\rfloor \\ \widetilde{x}_i &= \frac{x_i - x_i^{\min}}{s_i} - \hat{k}_i. \end{aligned} \tag{3}$$

Here $\lfloor\cdots\rfloor$ returns the integer part of the argument (`std::floor` function in C/C++). The order vector $\mathbf{n}$ defines the order of the B-spline basis functions for each dimension of $\mathbf{x}$. In addition to the shape of the B-spline basis functions, this vector $\mathbf{n}$ defines also the set Ω (see (4)) and influences therefore the number of one-dimensional B-spline basis function evaluations that are required to calculate the approximate detection probability $f(\mathbf{x})$ for a given state $\mathbf{x}$ (see (1)):

$$\Omega \equiv \mathbb{N}^0_{\leqslant n_0} \times \mathbb{N}^0_{\leqslant n_1} \times \cdots \times \mathbb{N}^0_{\leqslant n_{N-1}}. \tag{4}$$

In equation (5), the B-spline basis functions of orders 0, 1, 2 are shown for illustrative purpose:

$$\begin{array}{lll} B_0^{(0)}(x) = 1; & B_0^{(1)}(x) = 1 - x; & B_0^{(2)}(x) = \frac{1}{2}(1-x)^2 \\ B_1^{(1)}(x) = x; & B_1^{(2)}(x) = -x^2 + x + \frac{1}{2} & \\ B_2^{(2)}(x) = \frac{1}{2}x^2. & & \end{array} \tag{5}$$



(a) (b)

Figure 2. One-dimensional example of a *B*-spline of order $n = 1$. (a) The *B*-spline and (b) B_0 and B_1.

Figure 2(a) shows a one-dimensional example of a *B*-spline of order $n = 1$ to visualize the variables defined in equation (3). The continuous thick red line is the *B*-spline. The dotted thin lines represent the weighted *B*-spline basis functions.

The *B*-spline knot values $\varphi_{\mathbf{k}+\hat{\mathbf{k}}}$ are the free parameters of the model. Therefore,

$$\#Q = \prod_{i=0}^{N-1} (q_i + n_i) \tag{6}$$

(yet unknown) *B*-spline knot values $\varphi_{\mathbf{k}+\hat{\mathbf{k}}}$ are needed for the calculation of f in I.

For M detectors, the methodology is straightforwardly generalized by constructing a vector function $\mathbf{F}$ that consists of M scalar functions f:

$$\underbrace{\begin{pmatrix} f^0(\mathbf{x}) \\ f^1(\mathbf{x}) \\ \cdots \\ f^{M-1}(\mathbf{x}) \end{pmatrix}}_{\equiv\mathbf{F}(\mathbf{x})} = \sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}}) \underbrace{\begin{pmatrix} \varphi^0_{\mathbf{k}+\hat{\mathbf{k}}} \\ \varphi^1_{\mathbf{k}+\hat{\mathbf{k}}} \\ \cdots \\ \varphi^{M-1}_{\mathbf{k}+\hat{\mathbf{k}}} \end{pmatrix}}_{\equiv\boldsymbol{\Psi}_{\mathbf{k}+\hat{\mathbf{k}}}}$$

$$\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}}) \boldsymbol{\Psi}_{\mathbf{k}+\hat{\mathbf{k}}} \tag{7}$$

2.2. *Calculation of the B-spline knot values in* $\Psi_{\mathbf{k}+\hat{\mathbf{k}}}$

Since the detection process is complex, the detection probability values $\varphi^i_{\mathbf{k}+\hat{\mathbf{k}}}$ of the *B*-spline knots cannot be calculated directly. Instead, we propose to use a detailed Monte Carlo simulation to set these values. We call this simulation a 'pre-'simulation. The values $\varphi^i_{\mathbf{k}+\hat{\mathbf{k}}}$ are therefore replaced by Monte Carlo estimated values. For each *B*-spline knot, a weight vector $\boldsymbol{\omega}_{\mathbf{k}+\hat{\mathbf{k}}}$ of the same length as $\boldsymbol{\Psi}_{\mathbf{k}+\hat{\mathbf{k}}}$ is initialized with zero values and a variable $W_{\mathbf{k}+\hat{\mathbf{k}}}$ is set to zero:

$$\boldsymbol{\omega}_{\mathbf{k}+\hat{\mathbf{k}}} = \begin{pmatrix} w^0_{\mathbf{k}+\hat{\mathbf{k}}} \\ w^1_{\mathbf{k}+\hat{\mathbf{k}}} \\ \cdots \\ w^{M-1}_{\mathbf{k}+\hat{\mathbf{k}}} \end{pmatrix} \tag{8}$$

$$W_{\mathbf{k}+\hat{\mathbf{k}}} = \text{summed weight.} \tag{9}$$

During the pre-simulation, each photon is tagged with the initial randomly chosen starting state $\mathbf{x} \in I$ on the virtual border. The photon is either detected by exactly one of the detectors (i = detector number) or not at all. Using equations (3) the values $\hat{\mathbf{k}}(\mathbf{x})$ and $\widetilde{\mathbf{x}}(\mathbf{x})$ can be calculated and the values of $\boldsymbol{\omega}_{\mathbf{k}+\hat{\mathbf{k}}}$ and $W_{\mathbf{k}+\hat{\mathbf{k}}}$ increased for all $\mathbf{k} \in \Omega$ according to the rule

$$\begin{array}{ll} \text{if detected by } i & w^i_{\mathbf{k}+\hat{\mathbf{k}}} \leftarrow w^i_{\mathbf{k}+\hat{\mathbf{k}}} + B_{\mathbf{k}}(\widetilde{\mathbf{x}}) \\ \text{always} & W_{\mathbf{k}+\hat{\mathbf{k}}} \leftarrow W_{\mathbf{k}+\hat{\mathbf{k}}} + B_{\mathbf{k}}(\widetilde{\mathbf{x}}). \end{array} \tag{10}$$

With a sufficient number of simulated photons, this leads to an approximation of $\boldsymbol{\Psi}_{\mathbf{k}+\hat{\mathbf{k}}}$:

$$\boldsymbol{\Psi}_{\mathbf{k}+\hat{\mathbf{k}}} \approx \tilde{\boldsymbol{\Psi}}_{\mathbf{k}+\hat{\mathbf{k}}} = \frac{\boldsymbol{\omega}_{\mathbf{k}+\hat{\mathbf{k}}}}{W_{\mathbf{k}+\hat{\mathbf{k}}}}. \tag{11}$$

The update rule guarantees that

(i) The total detection probability of a photon is conserved, since $\sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}}) \equiv 1$ for all $\widetilde{\mathbf{x}} \in [0, 1] \times \cdots \times [0, 1] \subset \mathbb{R}^N$ (partition of unity).

(ii) The weighting of a B-spline knot for a given starting state/detection outcome pair corresponds directly to the weight of the B-spline knot used later in the actual Monte Carlo simulations.

(iii) The calculation of the B-spline values is sufficiently fast (linear in the number of simulated particles, independent of the number of B-spline knots).

This is exemplified in figure 2(b) for a one-dimensional B-spline of order $n = 1$. For a given photon state x, and according to the update rule (10), $W_{0+\hat{k}}$ is increased by $B_0(\tilde{x})$ and $W_{1+\hat{k}}$ is increased by $B_1(\tilde{x})$. The closer x to the knot $k + \hat{k}$ the larger will the knot value be increased. Note that $B_0(\tilde{x}) + B_1(\tilde{x}) \equiv 1$. The photon is therefore 'distributed' and attributed to the neighboring B-spline knots. When the photon is detected by detector i the same update rule is applied to $w^i_{0+\hat{k}}$ and $w^i_{1+\hat{k}}$.

2.3. *Usage of the B-spline model during tracking in the actual Monte Carlo simulations*

During tracking in the actual Monte Carlo simulation, the particles are tracked conventionally within the volume that is surrounded by the virtual border (usually the patient geometry). Photons that reach the virtual border are removed from the simulation and the B-spline function $\mathbf{F}(\mathbf{x})$ is evaluated with $\mathbf{x}$ being the photon state on the virtual border (12):

$$F_i(\mathbf{x}) = \sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}})\varphi^i_{\mathbf{k}+\hat{\mathbf{k}}} \approx \sum_{\mathbf{k}\in\Omega} B_{\mathbf{k}}(\widetilde{\mathbf{x}})\frac{\omega^i_{\mathbf{k}+\hat{\mathbf{k}}}}{W_{\mathbf{k}+\hat{\mathbf{k}}}}. \tag{12}$$

The set of M detectors $\mathcal{M} = \{0, \ldots, M - 1\}$ together with $\{F_i(\mathbf{x})\}_{i\in\mathcal{M}}$ represents the possible detection outcome ('which detector detects the photon . . . ?') and its detection probability (' . . . and with which probability?'), respectively.

Since the number of detectors is usually high and there are many B-spline knots per detector due to its $N = 5$ dimensions, it is not desirable or perhaps even not possible to keep all B-spline knot values for all detectors in memory. Therefore two ways to store the values in a sparse manner are presented that are used at different stages of the method.

Both storage schemes have in common that only those *B*-spline knots that have non-zero values are stored and both schemes require that in addition to the value $\tilde{\varphi}^{m(i)}_{\mathbf{k}+\hat{\mathbf{k}}} = \omega^{m(i)}_{\mathbf{k}+\hat{\mathbf{k}}} \big/ W_{\mathbf{k}+\hat{\mathbf{k}}}$ the detector number $m(i)$ must be stored:

$$[\tilde{\boldsymbol{\Psi}}_{\mathbf{k}+\hat{\mathbf{k}}}] = \begin{bmatrix} \tilde{\varphi}^{m(0)}_{\mathbf{k}+\hat{\mathbf{k}}}; & m(0) \\ \tilde{\varphi}^{m(1)}_{\mathbf{k}+\hat{\mathbf{k}}}; & m(1) \\ \dots; & \dots \\ \tilde{\varphi}^{m(M^{+}-1)}_{\mathbf{k}+\hat{\mathbf{k}}}; & m(M^{+}-1) \end{bmatrix}. \tag{13}$$

In equation (13), it is shown how the values are stored. For given $\mathbf{k} + \hat{\mathbf{k}}$, only those $M^{+} < M$ values that are larger than zero are stored. Here $m(i)$ is a bijective function from $\{0, \dots, M^{+} - 1\}$ to the set of detector numbers with non-zero detection probability. The two storing schemes differ only in the choice of the bijective function $m(i)$. In the first scheme (bijective function $m_{\mathrm{D}}(i)$) the entries are ordered with ascending detector numbers and in the second scheme (bijective function $m_{\mathrm{P}}(i)$) with descending detection probability. The first ordering is used whenever fast access for a given detector number is needed (e.g., when building the *B*-spline in the pre-simulation), the second ordering when it is advantageous to know that a hit in any of the following detectors (in memory) is less probable.

During the actual simulation, both ordering schemes are used. One vector $\mathbf{v}$ ordered by the detection number and of variable size serves to add and store all possible outcomes and detection probabilities for a given photon state $\mathbf{x}$. The *B*-spline obtained by the pre-simulation is stored using probability ordering. In this way it can be avoided to evaluate values that are very small and that would not change significantly the result. The read-out algorithm can be seen in algorithm 1.

All contributions to (12) with values smaller than $p_{\min}$ are therefore not included into the sum and the while loop is often aborted before i reaches $M^{+} - 1$ (see algorithm 1, line 9). This helps reducing the execution time spent on the calculation of very unlikely outcomes. The bijective function m_{D} maps from the entry number in $\mathbf{v}$ to the detector number and must be therefore updated each time a new detector is added to $\mathbf{v}$. In our implementation this was achieved by using a `std::map` of the standard template library (STL).

To save memory, not only zero but also small probability values $\tilde{\varphi}^{i}_{\mathbf{k}+\hat{\mathbf{k}}} < p_{\mathrm{mem}}$ can be suppressed already in the stored *B*-spline after the pre-simulation and prior to the actual simulations. In this case M^{+} is replaced by $M^{\star} < M^{+}$ and the set of stored elements for the *B*-spline is a subset of the elements shown in (13). Usually, $M^{\star} \ll M$ can be achieved (see assumption v in section 2.1). Therefore, two tuning parameters $p_{\min}$ and p_{mem} are available to reduce execution time (both parameters) and memory requirements (p_{mem}) at the cost of some accuracy.

To further reduce the number of evaluated small values that contribute only weakly to the result but normally to the execution time, a Russian roulette type exit condition can be introduced (see algorithm 1, lines 11–20). Whenever $B_{\mathbf{k}}(\tilde{\mathbf{x}})\varphi^{m}_{\mathbf{k}+\hat{\mathbf{k}}}$ is below a parameter p_{roulette} the detection probability for the subsequent detectors is only increased (and therefore evaluated) for every $1/a_{\mathrm{roulette}}$th photon on average. In order to account for this reduced probability, the added weights are multiplied by a_{roulette} in this case.

In order to preserve the total detection probability when the *B*-spline knot values smaller than p_{mem} are removed, a corrective factor was applied to the remaining non-zero *B*-spline knot values prior to the actual simulation:

$$a_{\mathbf{k}+\hat{\mathbf{k}}} = \frac{\sum_{i=0}^{M-1} w^{i}_{\mathbf{k}+\hat{\mathbf{k}}}}{\sum_{i=0}^{M^{\star}-1} w^{i}_{\mathbf{k}+\hat{\mathbf{k}}}}. \tag{14}$$

The factors $a_{\mathbf{k}+\hat{\mathbf{k}}}$ were usually very close to 1. Preliminary tests showed that the quality of the approximation could be slightly increased when using this factor.

Algorithm 1. Read out during actual simulation

```
calculate random number $r \in [0, 1]$
calculate $\tilde{\mathbf{x}}$ and $\hat{\mathbf{k}}$ according to (3)
create empty detection outcome vector $\mathbf{v}$
**for** $\mathbf{k} \in \Omega$ **do**
    $i \leftarrow 0$
    $m \leftarrow m_{\mathrm{P}}(i)$

    — Leave while loop to omit very small values:—
    **while** $\left(B_{\mathbf{k}}(\widetilde{\mathbf{x}})\varphi^{m}_{\mathbf{k}+\hat{\mathbf{k}}} > p_{\min}\right) \wedge (i < M^{+})$ **do**

        —Russian roulette for small values:—
        **if** $\left(B_{\mathbf{k}}(\widetilde{\mathbf{x}})\varphi^{m}_{\mathbf{k}+\hat{\mathbf{k}}} < p_{\mathrm{roulette}}\right)$ **then**
            **if** $r < 1/a_{\mathrm{roulette}}$ **then**
                $b \leftarrow a_{\mathrm{roulette}}$
            **else**
                **break** (leave while loop)
            **end if**
        **else**
            $b \leftarrow 1$
        **end if**

        —Add possible detector outcome:—
        **if** entry for detector m exists in $\mathbf{v}$ **then**
            $\mathbf{v}\left[m_{\mathrm{D}}^{-1}(m)\right] \leftarrow \mathbf{v}\left[m_{\mathrm{D}}^{-1}(m)\right] + b\, B_{\mathbf{k}}(\widetilde{\mathbf{x}})\tilde{\varphi}^{m}_{\mathbf{k}+\hat{\mathbf{k}}}$
        **else**
            insert entry $\mathbf{v}\left[m_{\mathrm{D}}^{-1}(m)\right] = b\, B_{\mathbf{k}}(\widetilde{\mathbf{x}})\tilde{\varphi}^{m}_{\mathbf{k}+\hat{\mathbf{k}}}$
            Update bijective function $m_{\mathrm{D}}^{-1}$
        **end if**

        $i \leftarrow i + 1$
        $m \leftarrow m_{\mathrm{P}}(i)$
    **end while**
**end for**
return possible detection outcome in vector $\mathbf{v}$
```

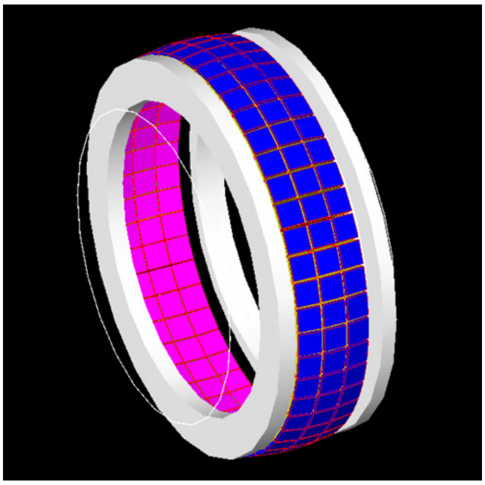


Figure 3. The modeled PET scanner Biograph 16 HiREZ.

3. Methods

3.1. Modeled tomograph

In figure 3, the used GATE model of the PET part of the Biograph 16 HiREZ PET/CT scanner of Siemens Medical Solutions can be seen. This model has been previously validated (Lazaro *et al* 2005, Michel *et al* 2006, Vauclin *et al* 2008). The scanner consists of three rings with 48 crystal blocks. Each crystal block contains 13 × 13 crystal detectors. The blocks of the two outer rings are tilted in such a way that the crystals approximate the (sub-)surface of a sphere. Dead time was not modeled. In our approach, we used the 48-fold rotational symmetry as well as the symmetry in axial direction and the transaxial intra-block symmetry (192 fold in total).

3.2. Computer resources

The used computer cluster consisted of 10 computers with two processors (Intel E5520 at 2.27 GHz). Each of the processors was equipped with four cores. For the simulations/ calculations, only half of the cores per computer was used (40 cores in total). The front end computer had an Intel i7 9200 CPU running at 2.67 GHz.

3.3. Pre-simulation and choice of B-spline parameters

In the pre-simulation, around 2.5 billion photons were started by randomly choosing starting position, direction and energy from the interval $\mathcal{I}^\star$. Due to the symmetry of the scanner, the photons were only directed to a sub-space $\mathcal{I}^\star \subset \mathcal{I}$ (see table 1). All values except energy were sampled uniformly. Since usually around half of the detected photons in the actual simulations are expected to be unscattered, it is desirable to have a more accurate detector model for energies around 511 keV and therefore the initial energy of 50% of the photons in the pre-simulation was set to 511 keV. The energy of the other photons was distributed uniformly between 350 keV and 512 keV. In table 1, the borders of the interval $\mathcal{I}^\star$ as well as the number of *B*-spline knots and the order of *B*-spline can be seen for each dimension. The

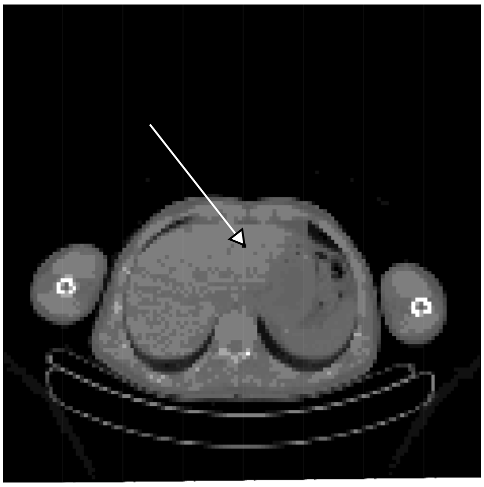


Figure 4. The central slice of the segmented CT images used for simulation. The segmented values go from air (black) to rib bone (white) with intermediate values for lung, soft tissue, spine bone, and 20 compositions to interpolate between lung and soft tissue. The arrow points to position $\mathbf{r}_a$.

Table 1. Chosen parameters and the interval $\mathcal{I}^\star$ that were used for the modeling.

i	Variable	Unit	$x_i^{\min}$	$x_i^{\max}$	q_i	n_i
0	θ	1	$\frac{\pi}{2} - 11.1\frac{\pi}{180}$	$\frac{\pi}{2}$	40	1
1	ϕ	1	0	$\frac{2\pi}{96}$	13	1
2	$\sin^2\theta_S$	1	0	$\sin^2\left(\frac{45\pi}{180}\right)$	20	1
3	ϕ_S	1	0	2π	20	1
4	E	keV	350[a]	512[a]	8	1

[a] see text.

radius of the sphere describing the virtual border was set to the mean crystal distance relative to the center of the scanner ($r = 435.23$ mm). Photons were started in the empty scanner such that they reach the virtual border within the desired position (θ, ϕ), inclination angle (θ_S, ϕ_S) and energy interval. In the actual simulation, the intervals for θ and ϕ could be widened to $\left[\pi - 11.1\frac{\pi}{180}, \pi + 11.1\frac{\pi}{180}\right]$ and $[0, 2\pi]$, respectively, by exploiting the aforementioned symmetries. B-spline values with values below $p_{\text{mem}} = 5 \times 10^{-4}$ were suppressed to save memory (see also section 2.3).

Photons were started such that they would hit the virtual border in a straight line at the randomly chosen positions and incident angles. In the actual simulation the detection system is replaced by the B-spline on the virtual border. Photons that leave the voxelized patient are transported to the virtual border in a straight line (no interaction is assumed between the voxelized patient and the virtual border). There, the B-spline is evaluated.

3.4. Actual simulation

The accuracy of the detector model was investigated by placing a single voxel of dimension $4 \times 4 \times 4$ mm^3 and filled with ^{18}F either in the empty scanner (filled with air) or in a scanner

Table 2. Calculation of the weights of the coincidences between the detectors *i* and *j* resulting from one back-to-back photon pair ('pink' and 'blue' photon). In the case of the *B*-spline simulation multiple detection outcomes are possible.

Conventional	*B*-spline
$w(i, j) = w^{\text{pink}}(i)\, w^{\text{blue}}(j)$	$w(i_0, i_0) = w^{\text{pink}}(i_0)\, w^{\text{blue}}(j_0)$ $w(i_0, j_1) = w^{\text{pink}}(i_0)\, w^{\text{blue}}(j_1)$ $\ldots$ $w(i_{n-1}, j_{m-1}) = w^{\text{pink}}(i_{n-1})\, w^{\text{blue}}(j_{m-1})$
Weights $\in \{0, 1\}$	Weights $\in [0, 1]$

with a voxelized patient that is derived from a CT scan (see figure 4). Three different voxel positions were chosen: (a) at the center $\mathbf{r}_\text{a} = (0, 0, 0)$ cm of the scanner, (b) at an off-central position $\mathbf{r}_\text{b} = (3, 14, 5)$ cm within the field-of-view (FOV), (c) at an off-central position $\mathbf{r}_\text{c} = (15, 15, 13)$ cm out of FOV. Each of the simulations was performed by running 40 × 1 s simulation of 100 MBq (^{18}F, positron range = 0 mm, no accollinarity) with 40 different random generator seeds using the random generator Mersenne Twister. Since dead time was not simulated and the time $t = 1\,\text{s} \times 40 \ll T_{\frac{1}{2}}(^{18}\text{F}) = 110\,\text{min}$, the resulting singles of the 40 simulations were simply added to obtain the equivalent of a 40 s scan. Photons coming from out of FOV that could potentially interact with the shielding due to their position and momentum direction were handed over back to GATE and the detection probability was determined in the conventional way. All other photons that eventually hit the virtual boundary were handled by the *B*-spline. The used tuning parameters were $p_\text{roulette} = 1 \times 10^{-3}$, $p_\text{min} = 2 \times 10^{-6}$, and $a_\text{roulette} = 10$.

In the conventional simulations that were performed for validation purposes, the same physical models and geometrical setup were used, but instead of using the *B*-spline for detection, the photons are tracked conventionally by GATE. In the conventional simulations and in the simulation using the *B*-spline, fictitious interaction tracking was used for the voxelized patient (Rehfeld *et al* 2009).

We also tested the accuracy of the sinograms derived from the *B*-spline-based simulations as follows. In order to obtain high statistics, 224 conventional simulations of 5 s scans of a 8 × 8 × 8 cm^3 box positioned at $\mathbf{r}_\text{b}$ and filled with 100 MBq (^{18}F, positron range = 0 mm, no accollinearity) were performed. Ten 0.2 s scans with 10 GBq (otherwise same setup) were simulated using the *B*-spline approach. Since in both cases the simulated times (5 s and 0.2 s) were much smaller than the decay time of ^{18}F it could be safely assumed that in total $5.6 = 224 \times 5 \times 10^8/10 \times 0.2 \times 10^{10}$ times more photons were simulated in the conventional simulations than in the *B*-spline simulations and that the corresponding sinograms should differ ideally by a factor of 5.6. These sinograms were created directly from the singles by only counting coincidences that came from the same decay. In the conventional way this was straightforward (see table 2, first column). Only when both photons *i* and *j* are detected ($w^{\text{pink}}(i) = 1$ and $w^{\text{blue}}(j) = 1$) a coincidence of weight $w(i, j) = 1$ is formed. In the *B*-spline simulation multiple combinations are possible due to multiple possible outcomes once a photon hits the virtual border. In this case all possible combinations were formed. In table 2 (second column), it is shown how the $m \times n$ coincidences were formed and weights were calculated when one of the original back-to-back photons could be detected by *n* detectors with non-zero detection probability and the other photon by *m* detectors.

3.5. Evaluation of the results

3.5.1. Accuracy. The results were evaluated by simulating detector crystal maps that show the detected singles (two-dimensional arrays ⟨number of crystals per ring⟩ × ⟨number of crystals in axial direction⟩ with the detection probability per crystal for each entry. In this way, it could be verified if the modeled detectors yield the same detection probability and cause the same block detector effects as the original detectors used in the conventional simulation. The detector maps were evaluated qualitatively by displaying the maps as gray value images and quantitatively by comparing the total number of detected photons and diagonal profiles through the detector maps with the results of the conventional simulation.

In order to investigate the impact of the *B*-spline approximation on the spatial noise properties of the simulated detector maps, the autocorrelation $\rho(\mathbf{\Delta};\mathbf{t})$ of the central detector crystal $\mathbf{t}$ was calculated for neighboring crystals $\mathbf{t}+\mathbf{\Delta}$:

$$\rho(\mathbf{\Delta};\mathbf{t}) = \frac{\mathbb{E}\left[(Y(\mathbf{t})-\mu(\mathbf{t}))\,(Y(\mathbf{t}+\mathbf{\Delta})-\mu(\mathbf{t}+\mathbf{\Delta}))\right]}{\sigma(\mathbf{t})\sigma(\mathbf{t}+\mathbf{\Delta})}. \tag{15}$$

Here $Y(\mathbf{t})$ stands for the number of detected photons by crystal $\mathbf{t}$. The mean $\mu(\mathbf{t})$ and the standard deviation $\sigma(\mathbf{t})$ of the counts for detector crystal $\mathbf{t}$ as well as the expectation $\mathbb{E}[\cdots]$ were calculated using the maps obtained from the 40 simulations of the computer cluster.

The simulated sinograms were evaluated graphically by showing two 2D sinograms and a line profile through these sinograms.

3.5.2. Computational efficiency. In order to investigate the computational efficiency, the total number of detected particles n that is needed to reach the same level of relative error in the detector crystals is determined for the conventional simulation ($n = n_{\mathrm{C}}$) and for the simulation using the *B*-spline detector model ($n = n_{\mathrm{B}}$). The acceleration factor α that is achieved by the *B*-spline approximation can then be calculated as

$$\alpha = \beta\frac{T_{\mathrm{C}}}{T_{\mathrm{B}}} \qquad \text{with} \qquad \beta = \frac{n_{\mathrm{C}}}{n_{\mathrm{B}}}, \tag{16}$$

where T_{C} and T_{B} are the time needed for the conventional simulation and the *B*-spline simulation, respectively.

We used the coefficient of variation to estimate the relative error. This is done by not only storing the number of detected weights $\mu_0(i)$ and the sum of detected weights $\mu_1(i)$ for each detector crystal i (with $w_j^{(i)} \equiv 1$ for the conventional simulation and $w_j^{(i)} \in \mathbb{R}$ for the modeled simulation), but also by storing $\mu_2(i)$:

$$\mu_1(i) = \sum_{j=1}^{\mu_0} w_j^{(i)}, \qquad \mu_2(i) = \sum_{j=1}^{\mu_0} \left(w_j^{(i)}\right)^2. \tag{17}$$

While the detection in the conventional simulation is a Poisson process and the coefficient of variation is easily approximated (equation (18)), the sum of the detected weighted counts in the simulation using the modeled detectors can be approximated by a weighted Poisson process and the coefficient of variation can be estimated according to equation (19) (de Vries *et al* 1990, Rehfeld 2007):

$$c_{\mathrm{C}} \approx \frac{\sqrt{\mathrm{Var}}}{\mu_1} \approx \frac{1}{\sqrt{\mu_0}} \tag{18}$$

$$c_{\mathrm{B}} \approx \frac{\sqrt{\mathrm{Var}}}{\mu_1} \approx \frac{1}{\mu_1}\sqrt{\frac{\mu_0}{\mu_0-1}\left(\mu_2-\frac{\mu_1^2}{\mu_0}\right)+\frac{\mu_1^2}{\mu_0}}. \tag{19}$$

Table 3. Total numbers of detected counts. The numbers in brackets indicate the relative deviation to the corresponding conventional simulation.

Simulation	No. of detected counts
Empty, $\mathbf{r}_a$, conv.	7.772×10^8
Empty, $\mathbf{r}_a$, *B*-spline	7.782×10^8 (+0.14%)
Empty, $\mathbf{r}_b$, conv.	8.533×10^8
Empty, $\mathbf{r}_b$, *B*-spline	8.493×10^8 (−0.47%)
Empty, $\mathbf{r}_c$, conv.	7.819×10^8
Empty, $\mathbf{r}_c$, *B*-spline	7.786×10^8 (−0.42%)
w/patient, $\mathbf{r}_a$, conv.	3.490×10^8
w/patient, $\mathbf{r}_a$, *B*-spline	3.488×10^8 (−0.08%)
w/patient, $\mathbf{r}_b$, conv.	4.915×10^8
w/patient, $\mathbf{r}_b$, *B*-spline	4.890×10^8 (−0.51%)
w/patient, $\mathbf{r}_c$, conv.	5.996×10^8
w/patient, $\mathbf{r}_c$, *B*-spline	5.970×10^8 (−0.43%)

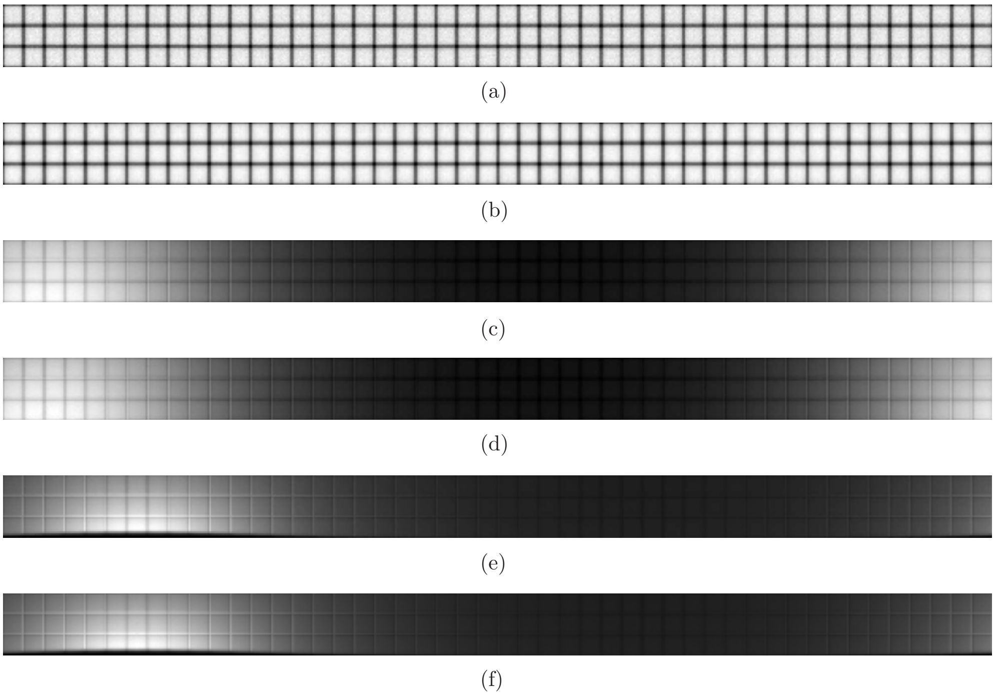
In two additional simulations (central voxel, empty scanner) the coefficient of variation was calculated for the conventional and the *B*-spline simulations. The mean coefficient of variation $\langle c_C \rangle$ and $\langle c_B \rangle$ using 13 neighboring detectors (taking the average to reduce statistical errors) was stored for every 5000 detected photons. In this way it was possible to plot n_C against $\langle c_C \rangle$ and n_B against $\langle c_B \rangle$ and $\beta = \beta(\langle c \rangle) = n_C(\langle c \rangle)/n_B(\langle c \rangle)$ could be estimated graphically by evaluating the two graphs at the same position $\langle c \rangle$.

4. Results

4.1. Accuracy

Figures 5 and 7 show the number of detected singles per detector displayed as gray value coded crystal maps. Visually, there is no difference between crystal maps of the conventional and the *B*-spline simulations. In figures 6 and 8, the corresponding diagonal profiles through these crystal maps can be seen. These profiles confirm the visual impression of the crystal maps, no deviation between the conventional simulations and the simulations using the approximated *B*-spline detector response can be found. Only in the case of the simulation of the voxel out of FOV, a small discrepancy probably due to an approximate model for the shielding (see figures 5 and 7, detector number $d < 50$ and $d > 600$) can be seen. Table 3 shows the total number of detected photons for each of the 12 simulations. The total number of detected photons always agreed within 0.51%. For both the empty scanner and the scanner with the voxelized phantom, the deviation for the central voxel was smaller than for the two other voxels. This indicates that the latter values could perhaps be improved by using a finer sampling of the *B*-spline for θ_S and ϕ_S.

In figure 9, the autocorrelation of the detector crystal maps for the conventional and *B*-spline simulation can be seen (corresponding to figures 5(a) and (b)). The gray values represent the autocorrelation of the central detector crystal with its $37 \times 37 - 1$ neighbors (black: negative correlation, white: positive correlation). Visually in both simulations similar values are obtained. This visual impression is confirmed by the gray value histograms (see figure 10). However, it should be noticed that the noise in figure 9(b) is clearly different (coarser) from that in figure 9(b). Figure 10 shows that the autocorrelation values are distributed similarly in


(a)

(b)

(c)

(d)

(e)

(f)

Figure 5. Gray images showing the detected single counts in crystal maps for the simulations with the empty scanner. White pixels represent the highest count values, black pixels the lowest count values, and gray pixels intermediate count values. Here the transaxial direction and the axial direction are displayed horizontally and vertically, respectively. (a) Voxel at position $\mathbf{r}_\mathrm{a}$, conventional simulation, (b) voxel at position $\mathbf{r}_\mathrm{a}$, simulation with *B*-spline approximation, (c) voxel at position $\mathbf{r}_\mathrm{b}$, conventional simulation, (d) voxel at position $\mathbf{r}_\mathrm{b}$, simulation with *B*-spline approximation, (e) voxel at position $\mathbf{r}_\mathrm{c}$, conventional simulation and (f) voxel at position $\mathbf{r}_\mathrm{c}$, simulation with *B*-spline approximation.

both cases when looking at a larger number of surrounding detector pixels (here $37 \times 37 - 1$). These results together with the direct comparison of the singles detector crystal maps indicate that is safe to use the proposed method to calculate mean values, but that the method should not be used to calculate the spatial correlation between single detector pixels or between two small groups of detector pixels.

Figures 11(a) and (b) show direct (not oblique) 2D sinograms resulting from the conventional sinogram simulation and *B*-spline sinogram simulation, respectively. The sinograms agree very well qualitatively and block detector effects can be observed in both cases and at the same positions. Despite the smaller number of simulated photons, the sinogram of the *B*-spline simulation is less noisy. In figure 12 a profile through both sinograms is plotted in order to compare them quantitatively. It can be seen that both simulations agree well, the *B*-spline results follow closely the results of the conventional simulation, the steep slope around profile position 200 is reproduced correctly, and the block detector effects are placed correctly. However, it seems that in few cases the heights of peaks due to block detector effects are slightly underestimated. Coincidences are more sensitive to deviations in the modeling of the angular detection probability than singles, and we expect therefore that these deviation can be reduced by increasing the number of angular *B*-spline nodes.

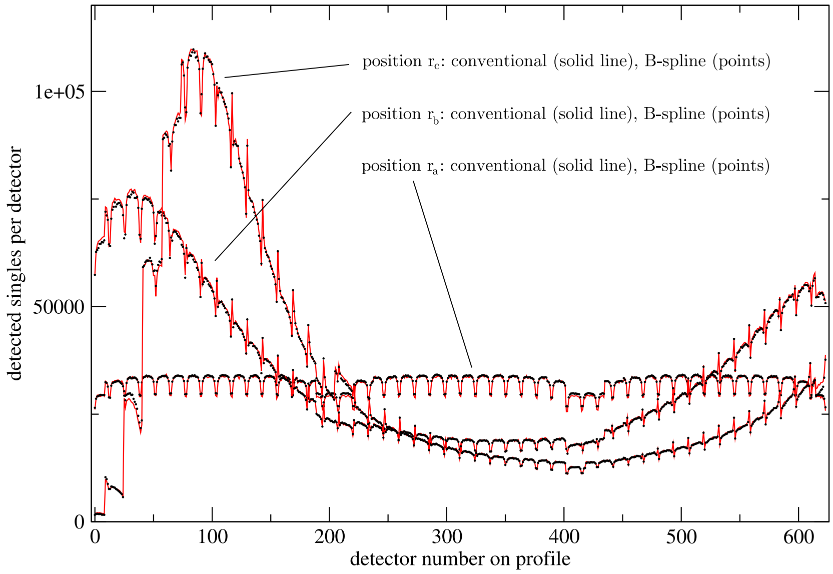


Figure 6. Diagonal profiles (lower left to upper right corner) through the detector crystal maps for the simulation with the empty scanner (see figure 5).

Table 4. CPU-time *t* needed for the pre-simulation including calculation of the *B*-spline values and for the actual simulations. Time $t^\star$ is the estimated CPU time needed to obtain the same accuracy as in the conventional simulation (assuming $\beta = 6$).

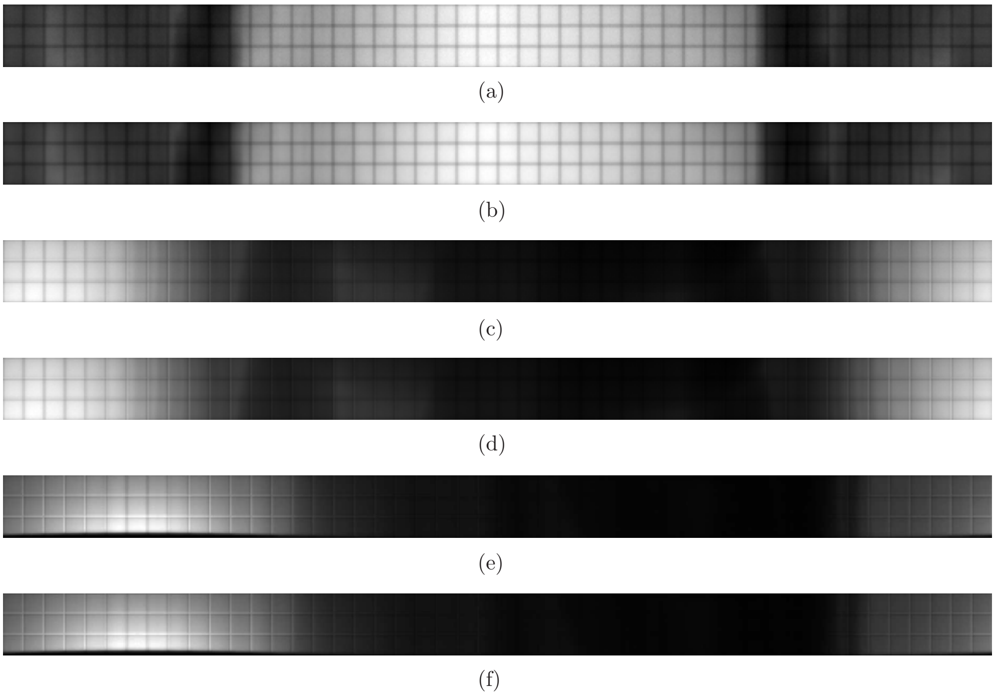
	t	$t^\star$
Pre-simulation+processing	185 h	
Conv. sim. (empty)	203 h	
B-spline sim. (empty)	39.3 h	6.6 h
Conv. sim. (w/patient)	173 h	
B-spline sim. (w/patient)	46.0 h	7.7 h

4.2. Computational efficiency

The time needed for the pre-simulation was almost 200 min on a cluster with 40 cores (corresponding to 132 h on a single CPU). The processing of the data and the creation of the *B*-spline took half an hour on 40 cores (processing per core) + 2.5 h on 10 cores (merging and processing per computer) + 8 h on a single core (merging and sorting using the front-end computer of the cluster). The processing time corresponds thus to 53 h on a single CPU, but is not fully parallelizable. Taking everything into account, the *B*-spline was created within less than $200\,\text{min} + 30\,\text{min} + 150\,\text{min} + 480\,\text{min} \approx 14\,\frac{1}{2}$ h on the computer cluster. In the presented implementation, the *B*-spline modeling required around 800 MB of memory.

In the actual simulation (position $\mathbf{r}_\text{b}$, empty scanner, 40 s scan of 100 MBq) the equivalent of around 200 single CPU hours was needed for the conventional simulation (see table 4). The *B*-spline-based simulation could be performed within less than 40 h.

Since a simulated photon in the *B*-spline simulation can lead to several non-integer detection probabilities in different detector crystals in contrast to a photon in the conventional



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7. Gray images showing the detected single counts in crystal maps for the simulations with the voxelized patient. White pixels represent the highest count values, black pixels the lowest count values, and gray pixels intermediate count values. (a) Voxel at position $\mathbf{r}_\mathrm{a}$, conventional simulation, (b) voxel at position $\mathbf{r}_\mathrm{a}$, simulation with *B*-spline approximation, (c) voxel at position $\mathbf{r}_\mathrm{b}$, conventional simulation, (d) voxel at position $\mathbf{r}_\mathrm{b}$, simulation with *B*-spline approximation, (e) voxel at position $\mathbf{r}_\mathrm{c}$, conventional simulation and (f) voxel at position $\mathbf{r}_\mathrm{c}$, simulation with *B*-spline approximation.

simulation that leads only to a hit or miss in a single crystal, it can be expected that factor β (see section 3.5.2) is greater than 1. Two shorter simulations were run to determine β. In figure 13, the results of these simulations are plotted. Figure 13 shows the required number of detected particles that are needed to obtain a given mean coefficient of variation. This required number is plotted for the conventional simulation (continuous line) and for the *B*-spline simulation (circles). The dashed line shows the correction factor β (ratio of the two curves). For small coefficients of variation the factor β is approximately constant, while for large coefficients of variation β increases. In the latter case, very few photons are simulated and every detected photon significantly improves the estimate even if the attributed weights are small as in the *B*-spline simulation. It can be seen that for a reasonably small coefficient of variation the factor β is around 6. This factor is used in the following.

Therefore about six times fewer particles need to be simulated with the simulation using the *B*-spline to obtain the same statistical accuracy as the conventional simulation. Thus, the equivalent execution time is reduced by a factor $\beta = 6$. Using this factor, we estimate that around $6.6 = 39.3/6$ h on a single CPU are sufficient to obtain the same statistical accuracy. The overall acceleration factor with respect to the conventional simulation can therefore assumed to be 30× ($\approx 203/6.6$). The acceleration factor does not take into account the time that is needed for the pre-simulation.

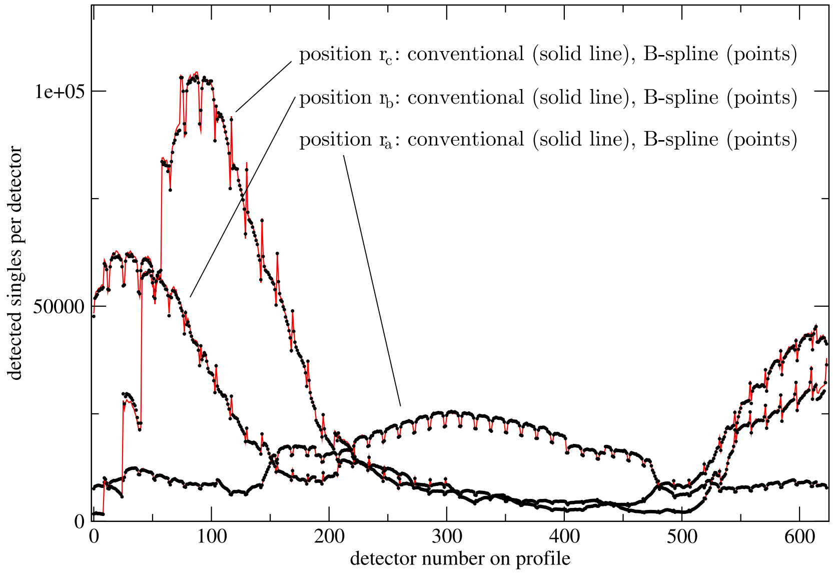


Figure 8. Diagonal profiles (lower left to upper right corner) through the detector crystal maps for the simulation with the scanner including the voxelized patient (see figure 7).

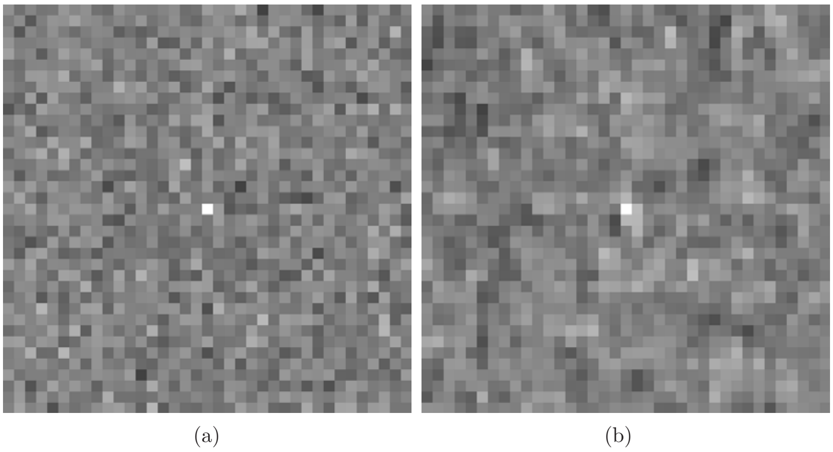


Figure 9. Autocorrelation of the central detector pixels of figures 5(a) and (b) with their $37 \times 37 - 1$ neighboring pixels. The gray values are scaled such that white and black correspond to 1 and -1, respectively. (a) Conventional and (b) *B*-spline.

The last two lines of table 4 refer to a simulation with the voxelized patient. Here, the acceleration factor ($\approx 22\times$) is lower, since the simulation of the particles within the voxelized patient results in an increase of execution time. Interestingly, less CPU time is needed for this simulation than for the conventional simulation with the empty scanner. This indicates that the reduction in CPU time due to the reduction of the number of simulated photons in the detection system (more photons are absorbed by the voxelized patient or removed from the simulation because their energy is too small after Compton interactions in the patient) outreaches the increase in CPU time due to the complex voxelized patient.

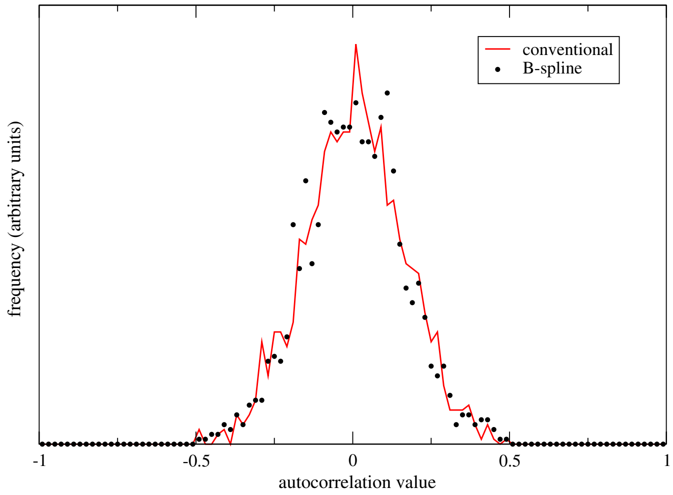


Figure 10. Histogram of the autocorrelation values of images 9(a) and (b).

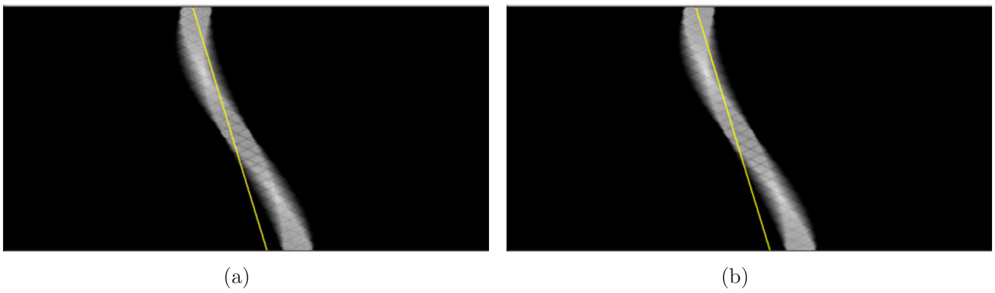


Figure 11. Direct 2D sinogram of the full 3D sinogram simulated with the conventional and with the *B*-spline method. LORs are only resorted and therefore not arc corrected. The yellow line indicates the positioning of the line profile shown in figure 12. (a) Conventional method and (b) *B*-spline.

For very large coefficients of variations (or a small number of simulated photons) the correction factor is even greater. This is reasonable, because in this case most detectors have not detected any photons and the *B*-spline simulation leads to multiple detections for a single simulated photon while the conventional simulation can lead to maximal one detected photon.

5. Discussion

With the presented *B*-spline approximation, it is possible to accurately model the detection process in a fast manner. Several parameters can be adjusted to reduce the memory consumption, execution time, pre-simulation time and increase the accuracy. While there is no closed form expression that can be used to determine the optimal set of parameters for a given detection system, the qualitative dependence on the parameters is quite clear (see table 5). The more *B*-spline knot values are used to represent the detection probability, the

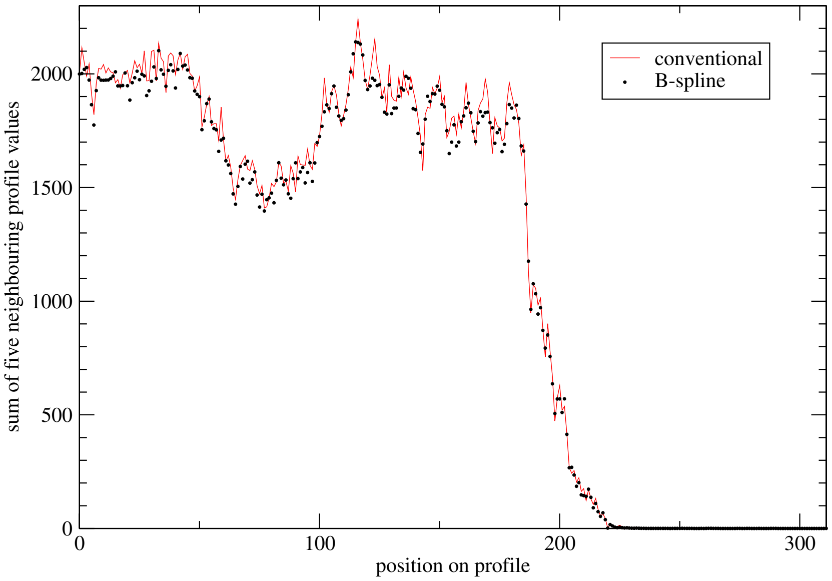


Figure 12. A ‘thick line’ profile (sum of the profile line shown in figures 11(a) and (b) and four neighboring profile lines). The results for the *B*-spline approach are multiplied by the factor 5.6 and should therefore ideally coincide with the results of the conventional simulation.

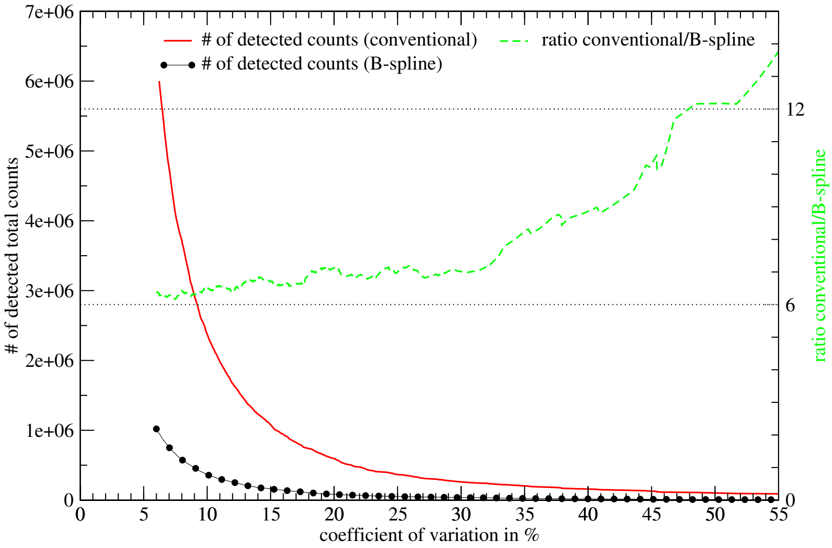


Figure 13. The total number of detected counts (voxel source at $\mathbf{r}_b$) that is needed to obtain a specific coefficient of variation (see equations (18) and (19)). The dashed line shows the ratio β of needed detected counts for the conventional (n_C, continuous line) and the *B*-spline (n_B, circles) simulation to obtain the same coefficient of variation $\langle c \rangle$.

higher is the demand in memory and the more particles need to be simulated during the pre-simulation in order to obtain sufficiently exact knot values. The accuracy increases with the number of knots, especially when the detection probability is a fast varying function. The number of *B*-spline knot values

Table 5. Influence of the parameters on the memory reduction, the execution speed, the accuracy and the pre-simulation speed. A plus sign indicates a positive dependence (increase of the parameter leads to an increase of the desirable property), a negative sign to a negative dependence.

Parameter	Memory reduction	Execution time reduction	Pre-simulation time reduction	Accuracy
q	−		−	+
n		−	+	+(−)
p_{mem}	+	+		−
p_{min}		+		−
p_{roulette}, a_{roulette}		+		−

$$\#\Omega = \prod_{i=0}^{N-1}(n_i + 1) \tag{20}$$

that is needed to evaluate the B-spline function at a given state $\mathbf{x}$ directly influences the execution time during the actual simulation. Thus, the higher the B-spline order the more time is needed. During the pre-simulation, the opposite is true: the larger $\#\Omega$, the more B-spline knot values are influenced by a simulated particle (see equation (10)) and consequently, for larger $\mathbf{n}$, fewer particles are needed to obtain the same statistical accuracy. The impact of the B-spline order on the final accuracy in the actual simulation depends on the detection probability function. In most cases B-spline of higher order should be more suitable to approximate the probability function, but in some cases the opposite might be true. The parameters p_{mem} and p_{min} can be used to reduce the execution time at the cost of some accuracy. The parameter p_{mem} can also be used to reduce the memory consumption and the parameter pair $p_{\mathrm{roulette}}/a_{\mathrm{roulette}}$ can be used to reduce the execution time at the cost of statistical accuracy.

In the presented application of the methodology to a PET scanner, the values presented in table 1 and the four parameters p_{mem}, p_{roulette}, a_{roulette} and p_{min} were chosen rather ad hoc. In initial tests of the program, the B-spline order $n = 2$ was used for the ϕ_{S} and E dependence of the approximated detection probability. Since the change $n = 2 \longrightarrow n = 1$ did not lead to worse results and an execution time reduction was anticipated, the order $n = 1$ was used in all dimensions in the subsequent simulations. Early tests with lower statistics also revealed that the results were not very sensitive to changes in B-spline parameters (table 5).

The number of photons to be simulated in the pre-simulation depends on the chosen number of B-spline knots, on the B-spline order, and on the desired accuracy of the B-spline knot values. In the case of a five-dimensional B-spline of order 1 (as used here), the standard deviation σ_Ψ of the simulated detection probability Ψ can be estimated by $\sigma_\Psi \lesssim 1.03\,\langle l\rangle^{-1/2}$ with $\langle l\rangle$ being the average number of photons that contribute to a B-spline knot (see the appendix, equation (A.8)). Here $\langle l\rangle = 2.5 \times 10^9 \times \#\Omega/\#Q \approx 3.51 \times 10^4$, where $\#Q$ is the total number of B-spline knots and $\#\Omega$ is the number of B-spline knots that are updated when a photon hits the virtual border in the pre-simulation (see equation (10)). This leads to an error of less than 0.55%. It should be noted that in our case, this error is smaller for knots that correspond to photons with energy of 511 keV since half of the simulated photons were emitted at this energy. On the other hand, the error for the other energies can be larger.

The spherical shape of the virtual border was motivated by the spherical arrangement of the detector blocks of this specific scanner. In the case of other detector arrangements other shapes can be chosen (e.g., planar or cylindrical).

Most problematic was the inclusion of the PET shielding into the simulations. Attempts to include the PET shielding into the *B*-spline detector response function were not successful. Further investigations revealed that the correct approximation of the penumbra that is caused by the shielding would require fine sampling of the angles (θ_S, ϕ_S) especially at the penumbra/no penumbra transition. Since this transition can be at almost any angle depending on the position of the source, this would require a fine sampling of a large solid angle. In PET, no collimators are used and the solid angle must be chosen very large (here $\theta_S = 45^\circ$). The inclusion of the shielding into the *B*-spline detector response function was therefore prohibitive due to memory constraints.

For sources inside the FOV, a possible error in the modeling of the shielding is not very important. The proposed method should however not be used for sources outside the FOV without increasing the number of *B*-spline knots (when enough memory is available). In case of doubt a conventional tracking method should be used. The *B*-spline method could be used for PET matrix simulations where voxels outside the FOV and dead time need usually not to be simulated. Randoms cannot be simulated directly using this method.

The proposed *B*-spline approximation methodology is very general and can therefore also be applied to other imaging modalities. Generally, the more time consuming the simulation of the detection system, the larger is the achieved acceleration factor. The presented method can be directly applied to PET simulations that include the simulation of scintillation photons. Due to the large number of such photons, these simulations are very time consuming. Consequently, much larger acceleration factors should be obtained for such simulations.

We expect that the methodology can also be successfully applied to SPECT simulations. Since the collimators in SPECT restrict the possible incident angle θ_S to a rather small solid angle, we expect no memory problems (unlike with the PET shielding). Since only a very small fraction of the photons is detected in conventional SPECT simulations, much larger acceleration factors of 1000–10000 are anticipated. Further, the methodology might be also applicable to CT simulations.

In principle, the methodology could also be used to generate the response function from experiments. Successful attempts to describe the detector response of imaging systems by using experimental data (but other methodologies) have been reported in the past (Formiconi *et al* 1989, Panin *et al* 2006). Finally, the methodology can be easily generalized to *B*-spline of higher dimensions. For example, in this way the spin of the photon could be easily incorporated.

All simulations used the quite complicated and general source handling of GATE. The acceleration factors can therefore be assumed to be even larger if the GATE source handling was replaced by a simple mechanism that is only working with single voxel-shaped sources.

6. Conclusions

We presented a novel methodology to approximate the response of a detection system with many detectors by a multidimensional *B*-spline that is based on a first Monte Carlo simulation (the 'pre'-simulation). In all subsequent Monte Carlo simulations using the same detection system but different imaging objects ('patients'), this approximate response can then be used to reduce computation time. The methodology was applied to the PET scanner Biograph 16 HiREZ and a 20–30 fold reduction in Monte Carlo simulation execution time could be achieved. The pre-simulation and creation of the *B*-spline could be performed within around half a day on a medium-sized computer cluster. Due to its very general form, the presented methodology could be probably also applied to other problems such as the simulation of PET scanners including the simulation of scintillation photons or other systems such as SPECT scanners. In both cases, an even much bigger reduction in simulation time is anticipated.

Acknowledgments

We thank Christian Michel for his help and the efficiency optimization group of the OpenGATE collaboration and Assen S Kirov for discussions. The work was funded by the European Union, Framework Program Seven (FP7), Marie Curie Intra-European Fellowships (IEF), call FP7-PEOPLE-2007-2-1-IEF, grant no. 221464-ACRIPET.

Appendix. Estimation of the statistical accuracy of the *B*-spline knot values

According to equations (10) and (11), a *B*-spline knot value is calculated via

$$\Psi \approx \frac{w}{W} = \frac{D}{D+C} = \frac{\sum_{s=0}^{l_1} b_s}{\sum_{s=0}^{l_1} b_s + \sum_{s=0}^{l_0} b_s}, \tag{A.1}$$

with l_1 being the number of detected photons that contribute to the formation of the *B*-spline knot value Ψ, $b_s = B^{(n)}(\tilde{\mathbf{x}}_s)$ is the contribution of the *s*th photon and l_0 is the number of photons not being detected that contribute to the denominator of equation (A.1). The knot indices $\mathbf{k}$ and $\mathbf{k} + \hat{\mathbf{k}}$ are dropped for convenience. Using in total L photons to simulate the complete *B*-spline, $\langle l \rangle = \frac{\#\Omega \times L}{\#Q}$ photons are used on average to calculate the *B*-spline knot value (see equations (20) and (6)). The variance σ_Ψ^2 of the detection probability Ψ can be estimated using Gauss' law of error propagation

$$\sigma_\Psi^2 \approx \left(\frac{\partial \Psi}{\partial C}\right)^2 \Bigg|_{\langle C \rangle} \sigma_C^2 + \left(\frac{\partial \Psi}{\partial D}\right)^2 \Bigg|_{\langle D \rangle} \sigma_D^2 \tag{A.2}$$

by using the variances σ_C^2 and σ_D^2 of C and D, respectively. Since the photons are started at random positions $\tilde{\mathbf{x}}$, both C and D are the sums of random weight values $b = b_s \in [0, 1]$ and are thus the result of a weighted Poisson process. The variance of the two sums can be estimated by

$$\begin{aligned} \sigma_{\mathrm{C}}^2 &= \langle l_0 \rangle \sigma_b^2 + \sigma_{l_0}^2 \langle b \rangle^2 = (1-\Psi)\langle l \rangle \langle b^2 \rangle \\ \sigma_{\mathrm{D}}^2 &= \langle l_1 \rangle \sigma_b^2 + \sigma_{l_1}^2 \langle b \rangle^2 = \Psi \langle l \rangle \langle b^2 \rangle \end{aligned} \tag{A.3}$$

using the mean values $\langle l_0 \rangle = (1-\Psi)\langle l \rangle$, $\langle l_1 \rangle = \Psi \langle l \rangle$ and $\langle b \rangle$, and the respective variance values $\sigma_{l_0}^2 = \langle l_0 \rangle$, $\sigma_{l_1}^2 = \langle l_1 \rangle$ and $\sigma_b^2 = \langle b^2 \rangle - \langle b \rangle^2$. Inserting equations (A.3) into equation (A.2) and using $\langle C \rangle = \langle l_0 \rangle \langle b \rangle$ and $\langle D \rangle = \langle l_1 \rangle \langle b \rangle$ leads to

$$\begin{aligned} \sigma_\Psi^2 &\approx \left(\frac{\Psi}{\langle l \rangle \langle b \rangle}\right)^2 (1-\Psi)\langle l \rangle \langle b^2 \rangle + \left(\frac{1-\Psi}{\langle l \rangle \langle b \rangle}\right)^2 \Psi \langle l \rangle \langle b^2 \rangle \\ &= \frac{\Psi(1-\Psi)}{\langle l \rangle \langle b \rangle^2} \langle b^2 \rangle \leqslant \frac{\frac{1}{4}}{\langle l \rangle \langle b \rangle^2} \langle b^2 \rangle. \end{aligned} \tag{A.4}$$

The statistical error σ_Ψ of the simulated detection probability Ψ can therefore be estimated as a function of the average number $\langle l \rangle$ of simulated particles per knot

$$\sigma_\Psi \lesssim \sqrt{\frac{\langle b^2 \rangle}{4\langle l \rangle \langle b \rangle^2}}. \tag{A.5}$$

For *B*-splines of order $\mathbf{n} = \mathbf{1}$, the first and second moment $\langle b \rangle$ and $\langle b^2 \rangle$ of the weights $b = b(\tilde{\mathbf{x}}) = b(\tilde{x}_0)b(\tilde{x}_1)b(\tilde{x}_2)b(\tilde{x}_3)b(\tilde{x}_4)$ can be calculated using the uniform product distribution probability function $p_5(u)$ in five dimensions

$$p_5(u) = \int_0^1 \cdots \int_0^1 \delta\left(\prod_{i=1}^{5} x_i - u\right) \mathrm{d}x_1 \cdots \mathrm{d}x_5 = \frac{(-\ln u)^4}{4!} \tag{A.6}$$

$$\langle b \rangle = \int_0^1 u p_5(u)\,\mathrm{d}u = 2^{-5} = \frac{1}{\#\Omega} \qquad \langle b^2 \rangle = \int_0^1 u^2 p_5(u)\,\mathrm{d}u = \frac{1}{243}. \tag{A.7}$$

With the values for the two moments, the statistical error can thus be estimated by

$$\sigma_\Psi(\mathbf{n} \equiv \mathbf{1}) \lesssim \frac{1.03}{\sqrt{\langle l \rangle}}. \tag{A.8}$$

References

Agostinelli S *et al* 2003 Geant4: a simulation toolkit *Nucl. Instrum. Methods Phys. Res.* A **506** 250–303

Allison J *et al* 2006 Geant4 developments and applications *Trans. Nucl. Sci.* **53** 270–8

Beekman F J, de Jong H W A M and van Geloven S 2002 Efficient fully 3-D iterative SPECT reconstruction with Monte Carlo-based scatter compensation *IEEE Trans. Med. Imaging* **21** 867–77

de Beenhouwer J, Staelens S, Vandenberghe S and Lemahieu I 2008 Acceleration of GATE SPECT simulations *Med. Phys.* **35** 1476–85

de Jong H W A M, Slijpen E T P and Beekman F J 2001 Acceleration of Monte Carlo SPECT simulation using convolution-based forced detection *IEEE Trans. Nucl. Sci.* **48** 58–64

de Vries D J, Moore S C, Zimmermann R E, Mueller S P, Friedland B and Lanza R C 1990 Development and validation of Monte Carlo simulation of photon transport in a camera *IEEE Trans. Med. Imaging* **9** 430–8

El Bitar Z, Lazaro D, Breton V, Hill D and Buvat I 2006 Fully 3D Monte Carlo image reconstruction in SPECT using functional regions *Nucl. Instrum. Methods Phys. Res.* A **569** 399–403

Formiconi A R, Pupi A and Passeri A 1989 Compensation of spatial system response in SPECT with conjugate gradient reconstruction technique *Phys. Med. Biol.* **34** 69–84

Jan S *et al* 2004 GATE: a simulation toolkit for PET and SPECT *Phys. Med. Biol.* **49** 4543–61

Lazaro D, Breton V and Buvat I 2004 Feasibility and value of fully 3D Monte Carlo reconstruction in single-photon emission computed tomography *Nucl. Instrum. Methods Phys. Res.* A **527** 195–200

Lazaro D, El Bitar Z, Breton V, Hill D and Buvat I 2005 Fully 3D Monte Carlo reconstruction in SPECT: a feasibility study *Phys. Med. Biol.* **50** 3739–54

Lazaro D, Michel C, Bendriem B and Buvat I 2005 Monte Carlo simulation of the Hi-Rez using GATE *J. Nucl. Med.* **46** 487P

Michel C, Eriksson L, Rothfuss H and Bendriem B 2006 Influence of crystal material on the performance of the HiRez 3D PET scanner: a Monte-Carlo study *NSS/MIC Conf. Record* **4** 2528–31

Panin V, Kehren F, Michel C and Casey M 2006 Fully 3-D PET reconstruction with system matrix derived from point source measurements *IEEE Trans. Med. Imaging* **25** 907–21

Qi J and Huesman R H 2005 Effect of errors in the system matrix on maximum *a posteriori* image reconstruction *Phys. Med. Biol.* **50** 3297–312

Qi J and Leahy R M 2006 Iterative reconstruction techniques in emission computed tomography *Phys. Med. Biol.* **51** R541–78

Rafecas M, Böning G, Pichler B J, Lorenz E, Schwaiger M and Ziegler S I 2004 Effect of noise in the probability matrix used for statistical reconstruction of PET data *IEEE Trans. Nucl. Sci.* **51** 149–56

Rehfeld N S 2007 Monte Carlo simulations in positron emission tomography reconstruction: full matrix, dual matrix, and system matrix compression *PhD Thesis* University of Tübingen, Germany

Rehfeld N S, Stute S, Apostolakis J, Soret M and Buvat I 2009 Introducing improved voxel navigation and fictitious interaction tracking in GATE for enhanced efficiency *Phys. Med. Biol.* **54** 2163–78

Song X, Segars W P, Du Y, Tsui B M W and Frey E C 2005 Fast modelling of the collimator-detector response in Monte Carlo simulation of SPECT imaging using the angular response function *Phys. Med. Biol.* **50** 1791–804

Takoudis G, Xanthos S, Clouvas A, Antonopoulos-Domis M, Potiriadis C and Silva J 2008 Spatial and spectral gamma-ray response of plastic scintillators used in portal radiation detectors: comparison of measurements and simulations *Nucl. Instrum. Methods Phys. Res.* A **599** 74–81

Vauclin S, Hapdey S, Michel C, Rebani H, Buvat I, Edet-Sanson A, Doyeux K, Gardin I and Vera P 2008 Monte Carlo simulations of respiratory gated ^{18}F-FDG PET for the assessment of volume measurement methods *NSS/MIC Conference Record* pp 4050–3