Region-oriented CT image representation for reducing computing time of Monte Carlo simulations

David Sarrut[1,2] and Laurent Guigues[2]

1 : Léon Bérard cancer center, 28 rue Laënnec, 69373 Lyon cedex 08, France
2 : CREATIS laboratory (UMR CNRS 5220, Inserm U 630),
INSA, Bât. Blaise Pascal, av. Jean Capelle, 69621 Villeurbanne cedex, France

Abstract

Purpose. We propose a new method for efficient particle transportation in voxelized geometry for Monte-Carlo simulations. We describe its use for calculating dose distribution in CT images for radiation therapy.

Material and methods. The proposed approach, based on an implicit volume representation named *segmented volume*, coupled with an adapted segmentation procedure and a distance map, allows to minimize the number of boundary crossings which slows down simulation. The method was implemented with the Geant4 toolkit and compared to four other methods: one box per voxel, parameterized volumes, octree-based volumes and nested parameterized volumes. For each representation, we compared dose distribution, time and memory consumption.

Results. The proposed method allows to decrease computational time by up to a factor of 15, while keeping memory consumption low and without any modification of the transportation engine. Speed up is related to the geometry complexity and to the number of different materials used. We obtained an optimal number of steps with removal of all unnecessary steps between adjacent voxels sharing a similar material. However, the cost of each step is increased. When the number of steps cannot be decreased enough, due for example to the large number of material boundaries, such method is not considered suitable.

Conclusion. This feasibility study shows that optimizing the representation of an image in memory potentially increases computing efficiency. We used the Geant4 toolkit but could potentially use other Monte-Carlo simulation codes. The method introduces a tradeoff between speed and geometry accuracy allowing computational time gain. However, simulations with Geant4 remain slow and further work is needed to speed up the procedure while preserving the desired accuracy.

1 Introduction

In radiation therapy, simulations are used to compute the three-dimensional (3D) dose distribution within a patient's body from a given set of irradiation parameters. Monte Carlo (MC) methods allow accurate simulation of the physical interactions of irradiation particles within patient tissues with photon/electron beams used in conventional radiotherapy [25], but also with proton [11] or carbon [15] beams used in hadrontherapy. Amongst other applications, MC simulations are a promising basis for designing fast treatment planning systems based on analytical simulations.

The present paper will focus on condensed simulation model which simulates the global effects of collisions in the course of a given step, but uses approximations. Condensed MC is known to track particles on a step-by-step basis [17]. There are two types of *s*teps: *physical steps* transport particles and simulate physical processes (e.g. photoelectric effect, Compton scattering, etc) in homogeneous media, whereas *geometrical steps* take into account the spatial environment of the particle and occur when it passes from a medium to another during transport.

Generally, the anatomical representation of the patient's organs is obtained from a Computerized Tomography (CT) image. Such a spatial description requires a large amount of data. In [7], the authors have used about 38 million voxels. Besides the need for computer memory, the lengthy processing time limits the use of MC. Besides using more powerful computers, several other approaches are currently under study to reduce the computational time. For example, several authors have used variance reduction [1], optimized energy cut or production range parameters [13, 14, 9]. Hybrid approaches combining analytical models and MC simulation have also been proposed [4]. All these methods either decrease the number of physical steps or accelerate the stepping process.

At least two possibilities exist to reduce the computational burden due to complex geometry: either reduce the complexity of the scene or speed up the geometrical stepping algorithm. We propose here a new approach to fasten the simulation when representing a 3D image using a MC code. It is based on allowing particles to cross voxel boundaries between two similar materials. The principle is not new since several authors have proposed comparable approaches, for example for DOSXYZnrc (EGSnrc) code [27, 23]. However, the way such boundaries are removed is different since, in our case, the transportation algorithm is not modified and thus any algorithm can be used. Instead, a new volume representation was defined, and we used a distance map to efficiently return to the transportation engine the distance to the nearest real boundary. Moreover, the method was implemented in the Geant4 toolkit. At our knowledge, similar approaches has not been published before. This method can be coupled with other acceleration techniques (e.g. variance reduction, use of parallel machines, etc) to further decrease simulation times.

This paper is organized as follows. Section 2 describes the state-of-the-art procedure to prepare a CT image in order to use it in a MC simulation (2.1) and the existing representations allowing to handle it in Geant4 (2.2). Section 3 describes

the proposed approach, which consists in an optimized image representation (3.1) for CT data which have been preprocessed by an adequate image segmentation procedure (3.2). Section 4 describes and discusses three series of experiments that we have carried out in order to evaluate the newly proposed method in comparison to existing ones. Section 5 finally concludes.

2 Existing image representations for particle tracking

2.1 From CT image to tissue composition

In order to represent a patient's anatomy given by a CT image in a MC simulation, it is necessary to associate each voxel of the image with a given tissue composition. A material m is defined by a mass density ρ ($g.cm^{-3}$) and an element weight vector denoted by $\omega_i \in [0;1]$, with $\sum_i \omega_i = 1$. The MC simulator uses such a description to load cross sectional datafiles and compute physics tables. The image acquisition process of a CT is the measure of the attenuation coefficients (μ, expressed in Hounsfield Units, H) of traversed tissues. H are obtained with eq. 1 ([20]), where $\mu(E)$ is the linear attenuation coefficient of a tissue $m = (\rho, \{\omega_i\})$ at energy E (see eq. 2), with N_A (mol^{-1}) the Avogadro constant, i the element index and σ_i the total cross section of the physical processes involved during the image acquisition.

$$H = \left(\frac{\mu}{\mu_{water}} - 1\right) \times 1000 \qquad \mu = \frac{H - H_{air}}{H_{water} - H_{air}} \tag{1}$$

$$\mu(E) = \rho N_A \sum_{i=1}^{n} \left(\frac{\omega_i}{A_i}\sigma_i(E)\right) \tag{2}$$

A stoichiometric calibration method to establish a relationship between H and tissue compositions has been proposed by Schneider et al. [21]. Images of phantoms made up of materials with known compositions and densities are used to calibrate the CT scanner. The authors have assigned the known materials to the measured H. Materials corresponding to intermediate H values have been interpolated both in terms of atomic composition and density according to eq. 3 and eq. 4.

$$\rho = \frac{\rho_1 H_2 - \rho_2 H_1 + (\rho_2 - \rho_1)H}{H_2 - H_1} \tag{3}$$

$$\omega_i = \frac{\rho_1(H_2 - H)}{(\rho_1 H_2 - \rho_2 H_1) + (\rho_2 - \rho_1)H}(\omega_{1,i} - \omega_{2,i}) + \omega_{2,i} \tag{4}$$

As stated by the authors, such a calibration procedure provides an approximated description of the composition of body tissues. Indeed, at conventional scanner energies (around 120 KeV) the imaging process mostly involves the photoelectric effect, coherent scattering and Compton scattering. Tissues with different compositions may thus result in similar H (this is particularly true for soft tissues). Moreover,

the partial volume effect may assign an artificial H value to a voxel containing a mixture of several tissues. Schneider et al [21] have estimated the accuracy of their calibration method to be better than $0.04 g.cm^{-3}$, which corresponds to about 20 to 40 H units, depending on the density (slightly lower slope for denser tissues). Kanematsu et al [8] have estimated the precision to be 1% (about 20 H). Schaffner et al. [19] have estimated the accuracy to be around 1.1% for soft tissues and 1.8% for bone tissues (from 20 to 40 H).

2.2 Existing image representations in Geant4

Once the CT image has been converted to an image of materials, different possibilities exist to insert the data into a MC simulation. In a MC simulation code, the physical world is described by means of elementary volumes of homogeneous composition. This work was elaborated using the Geant4 toolkit [1] but could be adapted to other simulation codes (provided that they force interactions at voxel boundaries, unlike DPM or VMC [5]). In Geant4, an image of materials can be introduced using several techniques :

1. *Box Volume (BV).* This first method is straightforward. For each voxel of the image, it creates a parallelepiped box (`G4Box`) filled with the associated material. Memory consumption is high (about 56 bytes per voxel [5]) and navigation is very slow, thus preventing the use of such a representation for large images.

2. *Parameterized Volume (PV).* This method, advocated for example by Jiang et al. [7], allows storing a single voxel representation in memory and dynamically changing its location and composition at run-time during the navigation. The main advantage of this method is high efficiency in memory space.

3. *Isothetic Volume (IV).* By merging adjacent voxels sharing similar material (with a given tolerance) into a larger voxel, it is possible to reduce the number of parallelepipeds. Hubert-Tremblay et al. [5] have proposed the use of octree compression to merge adjacent voxels. All resulting parallelepipeds are inserted into Geant4 by means of the BV (or possibly PV) approach. *"Isothetic"* means that the image is composed of parallelepiped cells arranged with their faces parallel to the three principal axes.

4. *Nested Parameterized Volume (NPV).* While reusing the same mechanism as PV, this representation also splits the 3D volume along the three principal directions, allowing logarithmic finding of neighboring voxels.

All these representations have in common that images are splitted into subvolumes of homogeneous composition which are parallelepipeds, either of the voxel size or larger. In BV, PV and NPV all parallelepipeds have the same dimensions, while in IV parallelepipeds have different sizes (the side length can be a multiple

of the initial voxel side length). The main advantage is that geometrical operations needed for particle tracking (for example computing the intersection between a vector and a volume) are fast to compute for a parallelepiped. However, the main drawback is that all the particles are forced to stop at the boundaries of all parallelepipeds, generating a supplementary step and additional time cost, even if the two neighboring parallelepipeds share the same content. Such artificial steps occur very often as human organs are far from being parallelepipedic.

Another potential source of slowness is the navigation algorithm used to find parallelepiped neighbors when a particle moves outside a volume. Geant4 uses a technique called *SmartVoxel* or *voxelisation* which consists in spanning the space with virtual geometrical slices. When seeking for the next volume, research is performed hierarchically, dimension by dimension. *Smartless*, a parameter corresponding to the average number of slices used per contained volume, defines the granularity of the voxelisation. Setting a high value when using an image composed of millions of parallelepipeds, leads to too many created slices, which increases memory consumption and prevents launching the simulation. The default value in Geant4 is 2, which is not adapted to very large geometry. As proposed in [6], a value of 0.02 was used.

3 Proposed approach : regions of arbitrary shape

The CT calibration accuracy estimations reported in section 2.1 above suggest that it should not be necessary to use the whole range of H and that voxels with neighboring H can be associated in a same homogeneous region. This is typically an *image segmentation problem* aiming at reducing the complexity of the scene while preserving overall accuracy. Our approach is based on two steps : (1) segmenting the CT image into homogeneous regions, whatever the shape of the resulting regions (parallelepiped or not), and (2) inserting the segmented image into the simulation using a new representation which allows to handle voxelised regions of arbitrary shape. The next section first describes the new representation proposed which allows to handle arbitrary shaped voxelised regions in Geant4 simulations. This method is very general and can be used with any segmented image as input. We then describe a method used to segment a CT image into regions.

3.1 Handling segmented volumes in Geant4 simulations

We assume here that an initial CT image has been segmented into regions of homogeneous composition, resulting into a *label image* in which each voxel is labeled with the index of a material. Together with this label image comes a file which associates each label to a given material composition and density. In the label image, a region is defined as a set of voxels sharing the same label, and is not necessarily a parallelepiped. Instead of providing an analytical description (such as NURBS or superquadric modeling [12]), we propose to describe the regions using the initial underlying discrete uniform voxel grid. We propose new geometric operators to

manage such a geometry. This approach is named *SegmentedVolume (SV)*.

In Geant4, the geometrical properties of a volume are handled by a `G4VSolid` class which must be able to answer several geometrical queries during the navigation process. The main operations are:

- `Inside(p)`, which computes whether the point `p` is inside, outside or on the surface of the volume.
- `DistanceToIn(p,v)`, which computes the distance covered by a particle at point p to enter the volume when travelling in direction v. If the line defined by (p, v) does not intersect the volume then an infinite distance is returned. Another version of `DistanceToIn` computes the shortest distance between the point p and the volume, independently of the direction. This is used as a *safety distance*, avoiding further computation when, for example, a shorter distance to another volume has been found previously.
- `DistanceToOut(p,v)` is similar to `DistanceToIn` and computes the distance needed by a particle at point p to move out of the volume according to the direction `v`. A direction-independent version is also available.

Such functions are very fast to compute if the volume is as simple as a parallelepiped (independently of its size). To represent a SV, we propose a new `G4VSolid` class, called `RegionSolid`, and we provide an efficient way to compute the above mentioned geometric queries. Given a label image, a `RegionSolid` is created for each region in the image ; it has access to the label image and to the label of the region it represents. Each `RegionSolid` then performs the computation of the different geometrical queries as follows. For `Inside(p)`, the calculation is straightforward. By rounding the coordinate of the point `p` according to the underlying voxel grid, one directly obtains the region to which `p` belongs by reading the voxel label. For `DistanceToIn` and `DistanceToOut`, we adapted a voxel-based discrete ray-tracing algorithm allowing to compute the distance on a step-by-step basis, each step being determined according to the voxel boundaries (see figure 1). For the direction-independent versions, we used a distance map which stores, for each voxel, the shortest distance to the nearest boundary of the region to which the voxel belongs (see figure 1). More precisely, for a voxel v belonging to region $R(v)$, the distance $dmin(v)$ is the shortest distance between any location in v and any location in $\overline{R(v)}$, the complementary set of $R(v)$:

$$dmin(v) = min\{d(p, q)|p \in v, q \in \overline{R(v)}\} \tag{5}$$

Hence for a given particle located at non integer coordinates within a voxel, the distance is an underestimate of the real nearest distance to the boundary of the region. Such an image of distances is usually called a *Distance Map* and efficient algorithms to compute them have been proposed in the image processing community. We used the `Insight Toolkit`[1] implementation of Danielsson's algorithm [3].

[1] http://www.itk.org

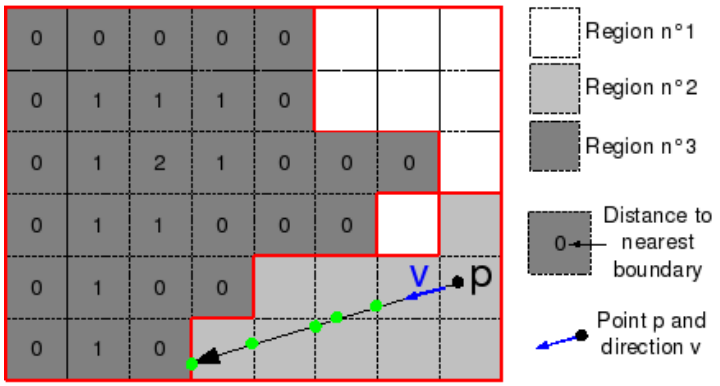


Figure 1: 2D illustration of the *SegmentedVolume* approach. To compute the `DistanceToOut` operation from the point `p` and along the direction `v`, a discrete ray-tracing is performed (green dots : geometrical steps). The numbers indicated inside the voxels in the dark region on the left represent the distance map values used for *safety distance* computation.

Of course, the computation of these geometrical operations is longer for a `RegionSolid` than for a parallelepiped (`G4Box`). However, in case of homogenous regions spanning multiple voxels, the technique optimizes the number of geometrical steps, which now only occur at real boundaries (between two different adjacent materials) and not at artificial boundaries (between two parallelepipeds sharing the same material).

In practice, such an approach does not involve modifying the Geant4 code. It is sufficient to create the new `RegionSolid` class derived from the abstract class `G4VSolid`. The distance map is computed and stored before the simulation. The two voxel matrices (the initial one composed of the matrix of labels and the distance map) are loaded and stored in memory. The `SegmentedVolume` in itself does not occupy much more space in memory than the NPV approach (see section 4.1).

3.2 Image segmentation procedure

In the image processing field, the term segmentation refers to the process of partitioning an image into multiple regions (sets of pixels) in order to decrease scene complexity by removing noise or unnecessary image details. For the sake of simplicity, we will only focus on photon beams for which the majority of physical processes depend on the electronic density of the material. Hence, the merging of two materials will be performed with a certain tolerance regarding the difference of electronic density between the voxels. For other types of beams (proton, carbon), such a step should be performed using a different density distribution, such as the stopping power density [8]. Electronic density was computed with eq. 6, where ρN_g is the number of electrons per unit volume of the mixture, Z_i the atomic number and A_i the atomic weight of element i.

$$\rho_e = \rho N_g / \rho^{water} N_g^{water} \qquad N_g = N_A \sum_i \frac{\omega_i Z_i}{A_i} \tag{6}$$

Based on such a density image, several image segmentation methods can be applied to merge homogeneous neighboring regions. Amongst other approaches, region-based segmentation methods are generally composed of a homogeneous criterion and a regularization one. For example, Hubert-Tremblay et al. [5] have proposed to define a region as homogeneous if it has no density gradient value greater than a user-specified threshold (named DGT, Density Gradient Threshold). Based on an octree method, the regions were constrained to be parallelepipeds. For the sake of comparison, we used this procedure for image segmentation. The resulting octree structure was used with the IV method and the resulting segmented image was also used with the proposed SV method. After segmentation, resulting images have a large number of regions (the leaves of the octree) with an averaged density. Like others authors [7, 5], we then attributed labels according to a quantization of the range of density, leading to a discrete number of different densities and corresponding materials. Quantization was performed according to the mean density difference obtained between initial and segmented density images. Such a procedure allows to obtain both an octree structure for the IV method and a label image for the SV one. The difference is that in our SV model, all parallelepipeds sharing the same label are viewed as an unique volume.

3.3 Dose scoring : voxel and dosel grids

We decided to separate the 3D matrix of voxels describing the patient (the geometrical grid) from the 3D matrix of scoring voxels (the scoring grid). By analogy with the term "voxel" describing a volume element used to record a material description, we propose to name "dosel" any volume element used to record a deposited dose. As emphasized in [24], this method allows, when high resolution is required for image description (for example when inhomogeneous materials are considered), to gain computational time by scoring doses in dosels that are larger than voxels and thus to accelerate convergence to reduced statistical uncertainty. Note that when dealing with segmented images, we used the average density obtained after segmentation to compute the dose in order to be consistent with the way energy deposit was simulated.

Relative statistical uncertainty $\varepsilon(\boldsymbol{x})$ at a given dosel $\boldsymbol{x}$ was estimated by eq.7 (see for example [10]). n is the number of primary events (or history), and $d^i(\boldsymbol{x})$ is the deposited energy in dosel $\boldsymbol{x}$ at (primary) event i. Average relative uncertainty s (eq. 8) was computed with the method proposed by [16] for all m dosels $\boldsymbol{x}$ such that $d(\boldsymbol{x}) > 0.5dmax$, with $dmax$ the maximum deposited dose in a dosel. More details on statistical uncertainty can be found in [2]. We used mean relative difference $MRD(d_1, d_2) = \frac{1}{m}\sum_x^m \frac{|d_1(\boldsymbol{x}) - d_2(\boldsymbol{x})|}{d_1(\boldsymbol{x})}$ to compare the dose distribution d_2 relatively to d_1. For comparison purpose, we will also present the results using dose difference

relatively to $dmax$, because this form is sometimes used in the literature ([5]) : $MRD_{dmax}(d_1, d_2) = \frac{1}{m}\sum_x^m \frac{|d_1(\boldsymbol{x})-d_2(\boldsymbol{x})|}{dmax}$.

$$\varepsilon(\boldsymbol{x}) = \sqrt{\frac{n\sum_i^n d^i(\boldsymbol{x})^2 - (\sum_i^n d^i(\boldsymbol{x}))^2}{(n-1)(\sum_i d^i(\boldsymbol{x}))^2}} \tag{7}$$

$$s = \sqrt{\frac{1}{m_{d>0.5dmax}}\sum_x^m (\varepsilon(\boldsymbol{x}))^2} \tag{8}$$

Contrary to [24], in the proposed Geant4 implementation, adding a dosel matrix to the voxel matrix does not significantly increase the computational time (less than 10% of the total time). Each dose deposition event leads to a floating point coordinate (in the world coordinate system) which is rounded to find the correct dosel index. The memory requirement is four floating point numbers per dosel: one for the deposited energy, one for the squared energy (for computing statistical uncertainty), one temporary value and one value storing the last hit event number as proposed in the efficient update method of [26, 22] that we implemented in Geant4. Finally, in Geant4, each step is defined by a line segment determined with a pre (starting) and a post (ending) position. In order to avoid biases and as advocated in the documentation, we computed a random location on the segment and added the current energy deposition to the dosel containing this location.

4 Experiments

We tested the proposed *SegmentedVolume* in three experiments. The first experiment was the irradiation of a simple water box. The goal was to validate the approach and estimate the computational burden due to the introduction of voxels. The second experiment used a phantom. It aimed at illustrating the time decrease as a function of the number of voxels. The third experiment was performed on patient data. It illustrates the proposed method in a complex geometry with numerous heterogeneities.

4.1 Water box experiment

4.1.1 Description

This first experiment aimed at illustrating the difference between non-voxelized and voxelized geometry in terms of computation time, dose distribution and memory consumption. It also aimed at validating the proposed method. The geometry was composed of a box of water. The beam source was a 6 MV photon conic beam with an energy spectrum obtained by MCNPX simulation of an Elekta Precise^{TM} device. The physics list was based on the low energy electromagnetic package. Distance to axis was set to 100 cm and we used a 20 mm radius beam size (at isocenter). Sixty million primary events were simulated. The box ($160 \times 160 \times 300$ mm^3) was

Param.	Param.	Rel. diff.		Diff rel to d_{max}	
		mean	stdev.	mean	stdev.
SB	BV	1.0%	0.7	0.7%	0.5
SB	PV	1.0%	0.8	0.7%	0.5
SB	NPV	1.0%	0.7	0.7%	0.5
SB	SV	0.7%	0.6	0.5%	0.4
BV	PV	0.0%	0.0	0.0%	0.0
BV	NPV	0.1%	0.0	0.0%	0.2
BV	SV	1.0%	0.8	0.7%	0.5
NPV	PV	0.5%	0.4	0.4%	0.3
NPV	SV	1.0%	0.8	0.7%	0.5

Table 1: *Water box experiment.* Mean (standard deviation) relative difference between various image parameterization; differences relative to *dmax* are also given. Parameterizations were *Single Box* (SB), *Box Volumes* (BV), *Parameterized Volumes* (PV), *Nested Parameterized Volumes* (NPV) and *Segmented Volumes* (SV).

described successively with a single homogeneous box (denoted by SB) or a voxelized volume with cubic voxels. Three sizes were used : 2^3, 1.5^3 and 1^3 mm^3, leading to 1, 2.3 and 7.7 million voxels, respectively. Four methods were compared : multiple G4Box (BV), parameterized volume (PV), nested parameterized volume (NPV) and segmented volume (SV). Of course, the IV model would give the same result as SB because the octree segmentation would lead to only one box. For all configurations, the deposited energy distribution was stored in a $80 \times 80 \times 150$ dosel grid with $2 \times 2 \times 2$ mm^3 dosel size. Production cuts were set to 0.5 mm for photons and 0.3 mm for electrons/positrons in the water box, and 2 mm for all particles outside the box. Such cuts prevent the creation of secondary particles when the energy range is below the chosen values ; instead, the energy is deposited locally.

4.1.2 Results

Table 1 displays the mean relative difference between the dose distribution obtained with the various parameterizations (BV,PV,NPV,SV) and without voxelisation (SB). All simulations led to average relative uncertainties below 0.9%. We also computed the total dose deposited inside the whole box with the different methods and found the following relative differences : 0.2 % between SB and PV (or BV), 0.18 %, between SB and NPV , 0.0023 % between SB and SV. Computational time, number of geometrical and physical steps (relatively to the SB experiment with 0.96 million voxels) are given in table 2 for SB, NPV and SV configurations, and for 0.96, 2.3 and 7.7 million voxels. Memory consumption (on a 64-bits AMD Athlon PC) according to the different parameterizations is displayed in table 3.

4.1.3 Discussion

In table 1, BV, PV and NPV should produce exactly the same results because Geant4 has been forced to use the same random number sequence. Indeed, the history of all

Time	0.96 m (2mm)	2.3 m (1.5 mm)	7.7 m (1 mm)
SB	1.0	1.0	1.0
NPV	4.6	5.6	7.6
SV	1.1	1.2	1.5
Geom Steps	**0.96 m (2mm)**	**2.3 m (1.5 mm)**	**7.7 m (1 mm)**
SB	1.0	1.0	1.0
NPV	55.1	73.3	109.5
SV	1.0	1.0	1.0
Physic Steps	**0.96 m (2mm)**	**2.3 m (1.5 mm)**	**7.7 m (1 mm)**
SB	1.0	1.0	1.0
NPV	3.0	3.2	3.7
SV	1.0	1.0	1.0

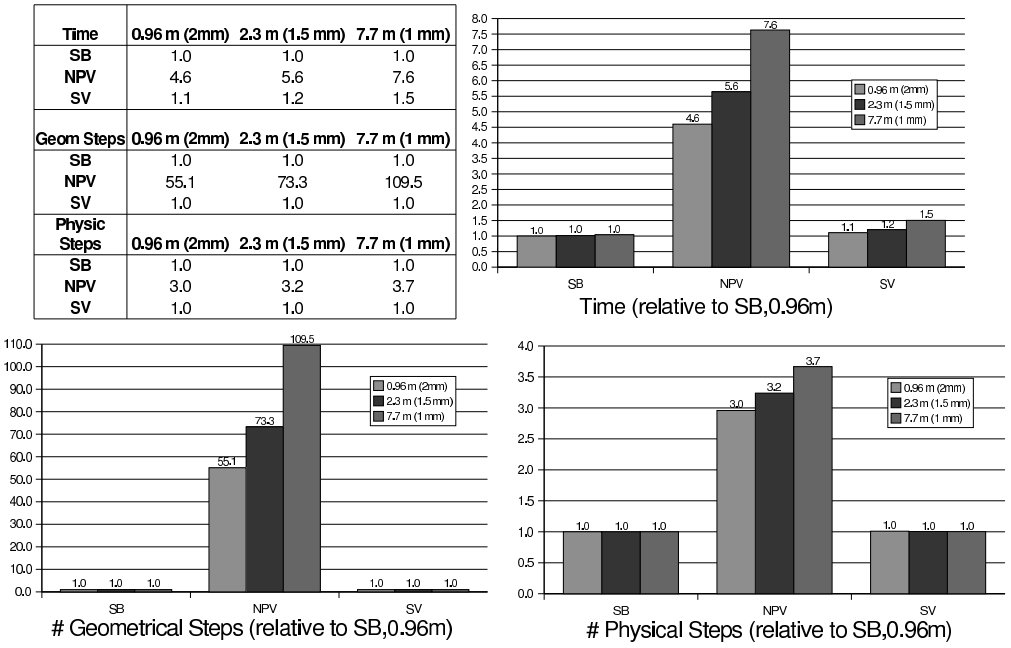


Table 2: *Water box experiment.* Time, number of geometrical steps and number of physical steps (expressed relatively to the SB experiment with 0.96 million voxels), for the three different configurations (SB,NPV,SV) and three voxel numbers (0.96, 2.3 and 7.7 million voxels).

Parametrisation	RAM
SB (no voxels)	*62 Mb*
BV	1416 Mb
PV (smartless = 2)	850 Mb
PV (smartless = 0.2)	303 Mb
PV (smartless = 0.02)	122 Mb
NPV	66 Mb
SV	70 Mb

Table 3: *Water box experiment.* Memory (in Mega-bytes) used for processing 1 million voxels according to the different methods. Three versions of the PV parameterization with different *smartless* values are indicated.

particles should be identical because all are stopped at exactly the same positions, whatever the parameterization used. However, even if it came true for BV compared to PV, NPV induced a slightly different behaviour. Inadequate management of some particular situations (a particle touching the edge of a volume but not entering it, for instance) was observed in Geant4 when using NPV. We have proposed the use of a patch to partially correct the problem (submitted to the Geant4 community). This patch reduces relative differences between BV/PV and NPV to less than 0.07%. We thus consider that the three parameterizations BV, PV, NPV are equivalent in terms of deposited dose.

First, we observed that dose differences between voxelized (BV, PV, NPV, SV) and non-voxelized (SB) volumes remained equivalent or below the statistical uncertainty (1%). The number of histories (60 million) was chosen to keep statistical uncertainty below 1%. SV should also be statistically identical to SB because it removes all voxel boundaries inside the box. However, there are some approximations in the computation of safety distance based on the distance map and we observed slight differences. We recall that all the mentioned differences are made (on average) dosel by dosel. Global differences computed on the whole box were very low ($< 0.003\%$ between BV and SV for example).

Such experiments also bring to light the computational cost of introducing voxels. For NPV, time was increased by a factor between 4 and 8 when using 0.96, 2.3 and 7.7 million voxels, respectively. We observed that the number of geometrical steps dramatically increased (by a factor of 50 to 100). The number of physical steps was also increased by a factor of 3 to 4. SV remained slower than SB (1.5 times slower for 7.7 million voxels) due to the burden of geometrical steps which increases when the image is complex. The simulation is however about 5 times faster with SV than with NPV for approximately the same memory requirement as SB (table 3).

4.2 Phantom experiment

4.2.1 Description

In these experiments, we used a CT image of a manufactured phantom composed of wood ($0.47\,g.cm^{-3}$), polyethylene ($0.9\,g.cm^{-3}$) and PMMA ($1.2\,g.cm^{-3}$), used for portal image calibration in our radiotherapy department. The phantom image (see figure 2) was segmented into five materials (by adding air and graphite). The image was $512 \times 512 \times 47$ (about 12.3 million voxels) with a resolution of $0.6 \times 0.6 \times 5$ mm^3. We compared NPV, IV and SV methods. For IV, as only five materials were used, we used an exact octree segmentation [5] with a DGT value of 0. Parameters were the same as for the previous experiment, except for the beam radius which was 50 mm. One hundred million primary events were simulated. Dosel size was set to $2 \times 2 \times 2$ mm. Production cuts were set to 0.01 mm for gamma, electrons and positrons.

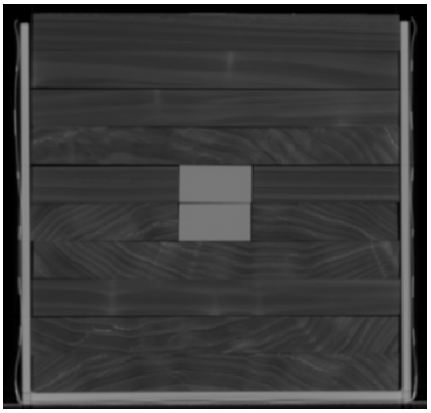


Figure 2: Slice of the phantom used in the experiment. The phantom is composed of wood, polyethylene and PMMA.

Param.	Param.	Rel. diff.		Diff rel to dmax	
		mean	stdev.	mean	stdev.
NPV	IV	0.92%	0.67	0.68%	0.53
NPV	SV	0.93%	0.72	0.68%	0.53
IV	SV	0.90%	0.71	0.66%	0.53

	NPV	IV	SV	NPV/IV	NPV/SV	IV/SV
Time (mn)	216 mn	43 nm	14 mn	**5.1**	**15.1**	**3.0**
SG	562 million	72 million	16 million	7.8	33.9	4.4
SP	48 million	28 million	24 million	1.7	2.0	1.2

Table 4: *Phantom experiment*. *Top*: Mean (standard deviation) relative difference between NPV, IV and SV for the phantom experiment. Differences relative to *dmax* are also given. *Bottom*: Time, number of geometrical steps and number of physical steps with NPV, IV and SV for 1 million events.

4.2.2 Results

Octree compression (IV) led to about 800 thousand parallelepipeds (compression ratio of about 93%, comparable to what is found in [5] for another phantom image). Simulations performed with 100 million primary particles led to average relative uncertainties below 0.8%. Table 4 displays the relative dose difference, the time and the number of geometrical and physical steps. The simulation was performed on a remote cluster of workstations (composed of AMD Opteron dual-core processors under Linux, 2Ghz). The time measurements were done for 1 million events. Time measurements were subject to variation due to the load of the cluster, however the normalization of values according to the percentage of CPU used allowed to compare simulation times.

4.2.3 Discussion

Computational time was decreased by a factor greater than 15 when using SV compared to NPV. Such a performance can be explained by the large reduction in the number of steps (almost 34 times less). We note that the reduction here was superior to the one obtained in the waterbox experiment. There are more voxels (12.3 million) and the number of physical interactions is higher. Average dose relative differences between the three methods remain lower than 1% which in the order of the statistical uncertainty. Figure 3 illustrates the dose distribution in the phantom and figure 4 shows a depth dose distribution plot with the three methods (NPV, IV and SV) and the corresponding differences between NPV and SV. We observed that there were no dosel with relative error greater than 3%. The effect of the boundaries between different media can be observed with the vertical lines.

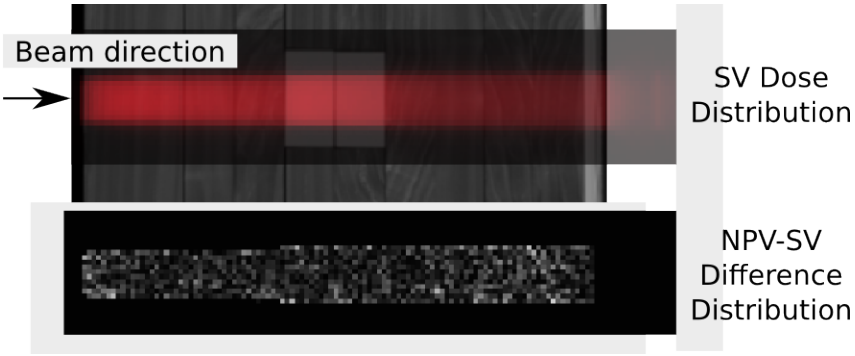


Figure 3: Example of the dose distribution superimposed on one phantom CT slice. The corresponding dose difference distribution between NPV and SV is shown at bottom.

4.3 CT image experiment

4.3.1 Description

For the third series of simulations, we used a thorax CT image of a patient with NSCLC (Non-Small Cell Lung Cancer). Initial image size was $512 \times 512 \times 77$ (more than 20 million voxels) with a resolution of $1.2 \times 1.2 \times 5$ mm^3. We reduced the number of voxels by segmenting the air region surrounding the patient (according to the method described in [18]) and kept the bounding box corresponding to $349 \times 228 \times 77$ voxels (about 6 million). After CT calibration [21], six segmentations were performed with the octree method according to six DGT parameters (see table 5 and figure 5). In this figure, we do not show all segmentations because they are hardly visually distinguishable from the original image. Three methods were compared for dose and timing: NPV, IV and SV. One hundred million primary events were simulated. Dosel size was set to $2.5 \times 2.5 \times 5\, mm^3$ (about 1.3 million dosels). Other parameters were the same as for previous experiment.

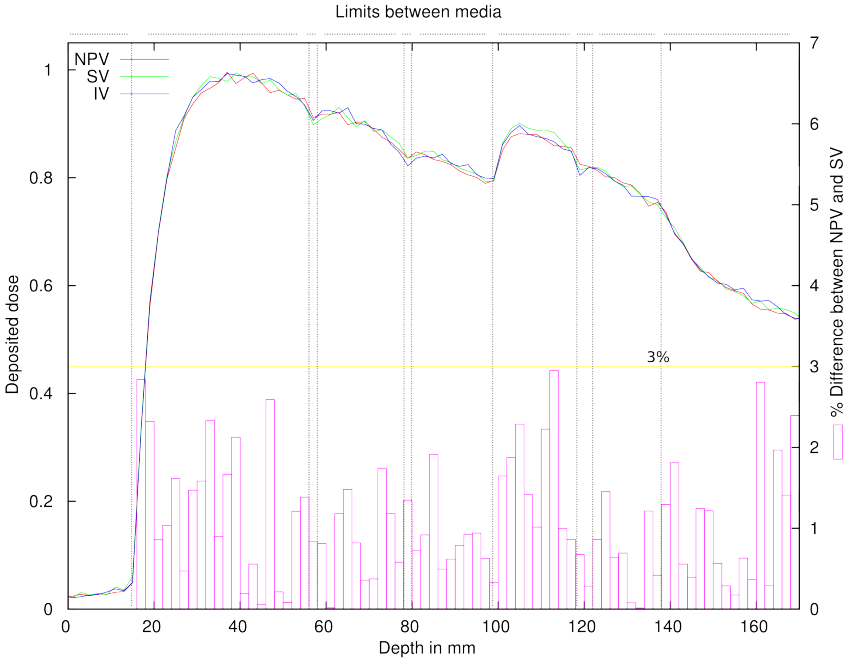
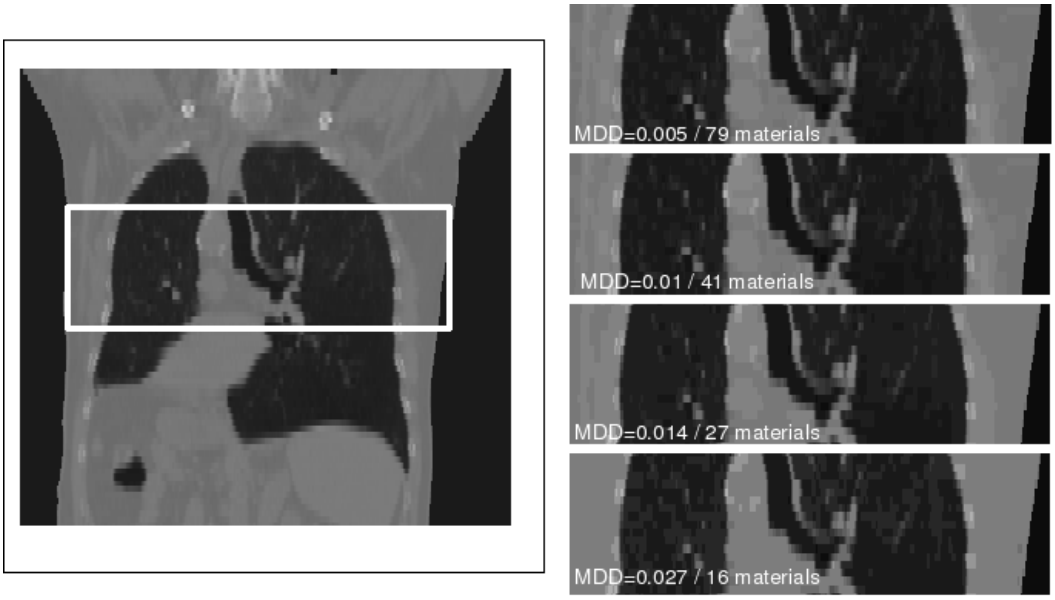
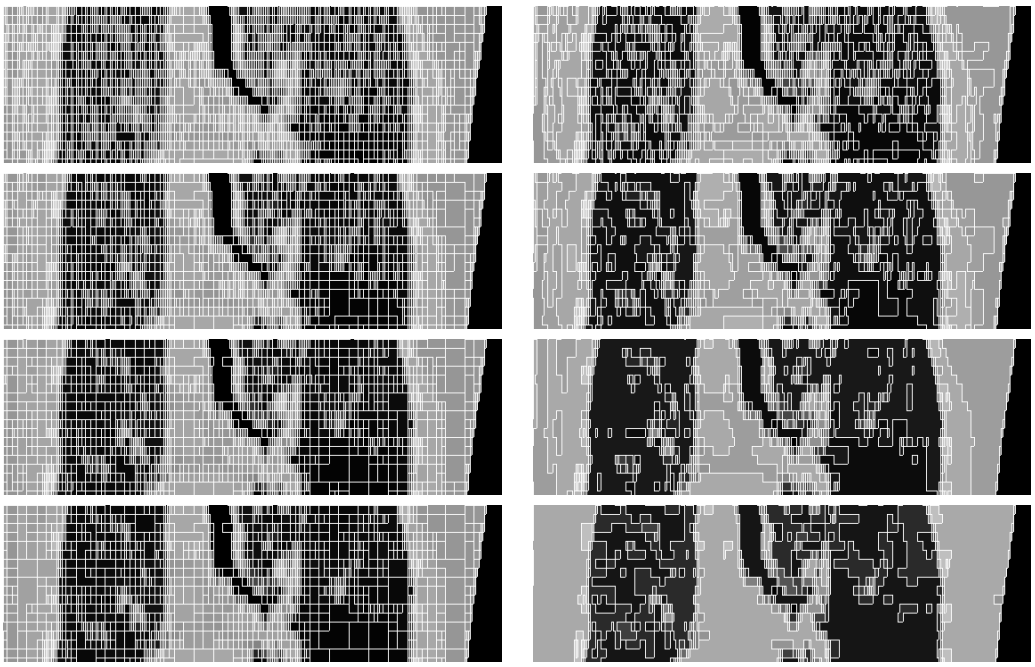


Figure 4: Dose profile in the phantom computed with the methods NPV, SV and IV (dose is expressed relatively to the maximum dose observed with NPV, on the left axe). Relative differences between NPV and SV are shown with vertical boxes (values are on the right axe). Vertical lines show the limits between different media.



Original CT slice | Segmented CT slice, mean HU



Octree segmented CT slice | Final segmented CT slice

Figure 5: Top-left: original CT slice. Top-right: The four images show the segmentation results obtained with different DGT values. Mean Density Difference (MDD) and corresponding number of materials is indicated on the images (see table 5). Bottom-left: parallelepipeds leaves resulting from the octree procedure. Bottom-right: same images after quantization procedure. Simplification is especially visible on the last images.

	DGT	Mat.	Mean	Max	Stdev	Nb of boxes
Test1	0.010	242	0.002	0.013	0.002	2,733,826
Test2	0.020	117	0.004	0.028	0.004	2,069,821
Test3	0.025	79	0.005	0.041	0.006	1,855,670
Test4	0.050	41	0.010	0.079	0.012	1,233,620
Test5	0.075	27	0.014	0.120	0.017	934,133
Test6	0.100	16	0.027	0.181	0.030	750,708

Table 5: Segmentation DGT ($g.cm^{-3}$), quantization (number of materials), density differences (mean, max, standard deviation) and number of resulting boxes (octree leafs).

4.3.2 Results

All 18 simulations (3 methods, 6 cases) led to average relative uncertainties below 0.9%. Table 6 displays the dose relative differences and table 7 the computational time according to the different segmentations, for NPV, IV and SV methods. Time is expressed relatively to NPV and test1 experiment.

4.3.3 Discussion

The mean accuracy in mass density during the segmentation stage ranged from 0.002 to 0.027 $g.cm^{-3}$. The octree stage yielded between 750,000 and near 2 million boxes. Table 7 illustrates that using SV with a reduced number of materials allowed to reduce the computational time by up to a factor greater than 4. Using the octree structure also decreased the time by up to a factor of approximately 1.5. The gain is attributable to the large decrease in the number of geometrical steps, the number of physical steps only slightly decreases. We also observed that for excessive numbers of materials, the methods are not efficient and can even lead to increased computing time (test1). We observed that the dose distribution differences between NPV, IV and SV methods remained below 1.2% (left part of table 6). In this case, this is the same material approximation and, as in the phantom experiment, differences are in the order of the uncertainty (which is below 0.9%). In the right part of the table 6, dose differences due to material approximation can be observed and range from 1.1% to 1.5%. Figure 6 illustrates the experiment with profiles : on the upper part, electronic density profiles are displayed for reference image, test n°5 and test n°6. The approximation due to the segmentation part can be observed (note that the segmentation was performed in 3D, which explain that in the right part of the figure, the segmentation n°6 seems crude). On the bottom part of the figure, deposited dose distribution profiles are shown for test n°1 compared to test n°5 and 6, with the corresponding differences between NPV and RV. In this example, some differences are greater than 3% with test n°6, while this is not the case with test n°5. We computed that 90% of the dosels in test n°6 have

	Param.	Param.	Rel. diff.		Diff rel to dmax	
			mean	stdev.	mean	stdev.
Test 1 242 mat.	NPV	IV	1.0%	0.8	0.7%	0.6
	NPV	SV	1.1%	0.9	0.8%	0.6
	IV	SV	1.1%	0.8	0.8%	0.6
Test 2 117 mat.	NPV	IV	1.1%	0.8	0.7%	0.6
	NPV	SV	1.1%	0.9	0.8%	0.6
	IV	SV	1.1%	0.8	0.8%	0.6
Test 3 79 mat.	NPV	IV	1.0%	0.8	0.7%	0.6
	NPV	SV	1.1%	0.9	0.8%	0.6
	IV	SV	1.1%	0.9	0.8%	0.6
Test 4 41 mat.	NPV	IV	1.0%	0.8	0.7%	0.6
	NPV	SV	1.2%	0.9	0.8%	0.6
	IV	SV	1.1%	0.9	0.8%	0.6
Test 5 27 mat.	NPV	IV	1.1%	0.8	0.7%	0.6
	NPV	SV	1.2%	0.9	0.8%	0.6
	IV	SV	1.2%	0.9	0.8%	0.6
Test 6 16 mat.	NPV	IV	1.1%	0.8	0.7%	0.6
	NPV	SV	1.2%	0.9	0.8%	0.6
	IV	SV	1.2%	0.9	0.8%	0.6

	SV (ref=test1/NPV)			
	Rel. diff.		Diff rel to dmax	
	mean	stdev.	mean	stdev.
Test1	1.1%	0.9	0.8%	0.6
Test2	1.1%	0.9	0.8%	0.6
Test3	1.2%	0.9	0.8%	0.6
Test4	1.3%	1.1	0.9%	0.7
Test5	1.5%	1.4	1.0%	0.9
Test6	1.5%	1.4	1.0%	0.9

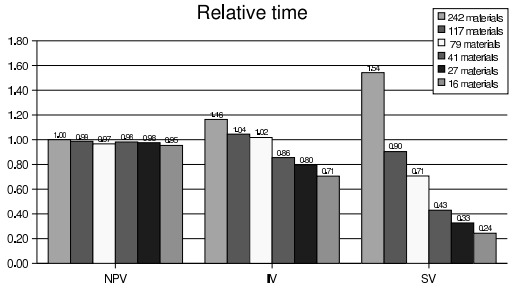
	IV (ref=test1/NPV)			
	Rel. diff.		Diff rel to dmax	
	mean	stdev.	mean	stdev.
Test1	1.0%	0.8	0.7%	0.6
Test2	1.1%	0.8	0.7%	0.6
Test3	1.1%	0.9	0.8%	0.6
Test4	1.2%	1.1	0.8%	0.7
Test5	1.4%	1.4	1.0%	0.9
Test6	1.4%	1.3	1.0%	0.8

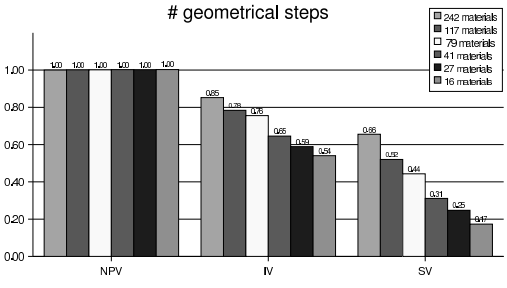
Table 6: *CT experiment.* Dose relative difference between NPV, IV and SV, according to different segmentations. Left: differences between methods using the same segmentation. Right : differences relatively to the reference NPV with test1 case.

Time	242 mat.	117 mat.	79 mat.	41 mat.	27 mat.	16 mat.
NPV	1.0	1.0	1.0	1.0	1.0	1.0
IV	0.9	1.0	1.0	1.2	1.3	1.4
SV	0.6	1.1	1.4	2.3	3.1	4.1

Geom S	242 mat.	117 mat.	79 mat.	41 mat.	27 mat.	16 mat.
NPV	1	1	1	1	1	1
IV	0.85	0.78	0.76	0.65	0.59	0.54
SV	0.66	0.52	0.44	0.31	0.25	0.17

Phys S	242 mat.	117 mat.	79 mat.	41 mat.	27 mat.	16 mat.
NPV	1	1	1.01	1.02	1.03	1.04
IV	0.97	0.94	0.93	0.9	0.9	0.9
SV	0.94	0.9	0.88	0.84	0.85	0.79





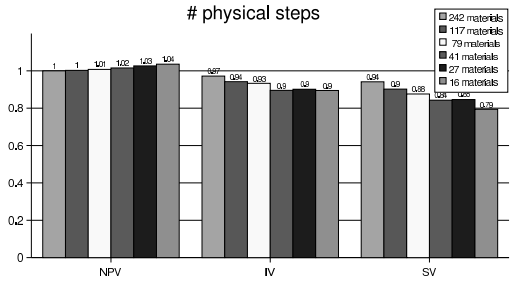


Table 7: Time, number of geometrical steps and number of physical steps (expressed relatively to the NPV case with 242 materials), for the three different configurations (NPV,IV,SV) and the six segmentation cases.

relative dose difference lower than 3%. Maximum differences were greater than 7% for about 50 dosels and can reach 10% in four dosels. It is however difficult to separate what is due to statistical uncertainties (around 0.9%) from what is due to NPV vs. SV implementation (maybe about 0.5%) and what is due to material approximation. Note that [5] also observed that the mean dose difference expressed in % of the maximum is around 1%, which is in the same order as our observation. More validation are still needed to compare the different approaches according to experimentally measured data in such very inhomogeneous media.

Finally, the gain of computational time with the proposed method increases almost linearly according to the parameter DGT. This parameter is linked to the number of regions of the segmentation. For high DGT values, a small number of regions is obtained and the proposed method is faster. However, dose differences increase as the image is simplified. The image segmentation parameters (DGT here) can thus be used to control the tradeoff between speed and desired accuracy.

5 Conclusion

We have proposed a method to decrease the computational time required for Geant4 simulations involving a 3D image-based description of the scene, such as a CT image of a phantom or a patient. Our method consists in representing the matrix of voxels with regions having specific boundary-finding functions based on discrete ray-tracing. It is different from previously published methods because the transportation algorithm is not modified. We have also described experiments permitting to validate the proposed method in different configurations. According to the image segmentation procedure, the proposed method allowed to eliminate the MC steps required at artificial interfaces (between two voxels sharing the same material) until reaching an optimal number of geometrical steps. We are able to decrease the computational time by up to a factor of 15 (in the phantom case involving few materials).

The gain still remains a tradeoff between time and accuracy, with the reduction in the number of different materials leading to larger speed up. We want to emphasize that the main goal of this paper was not the in depth study of the influence of this segmentation stage on final simulation results. The loss of accuracy due to the material simplification still needs to be evaluated for each application : it is the responsibility of the user to decide if dose differences that occur between NPV and SV are acceptable of not for a given application. For example, scene simplification will probably not have the same importance for dosimetric or for imaging applications. As in the method DOSSCORE proposed by Smedt at al. [23], the speed gain is much more significant for weakly inhomogeneous medias (factor of 15). However, it also remains interesting (reduction in time of 3 to 4) for patient data.

On patient data, the segmentation of the CT image could be difficult to perform automatically a priori because of the noise. Manual segmentation can also be employed. Other criteria than the DGT of homogeneous criteria could be used (such as

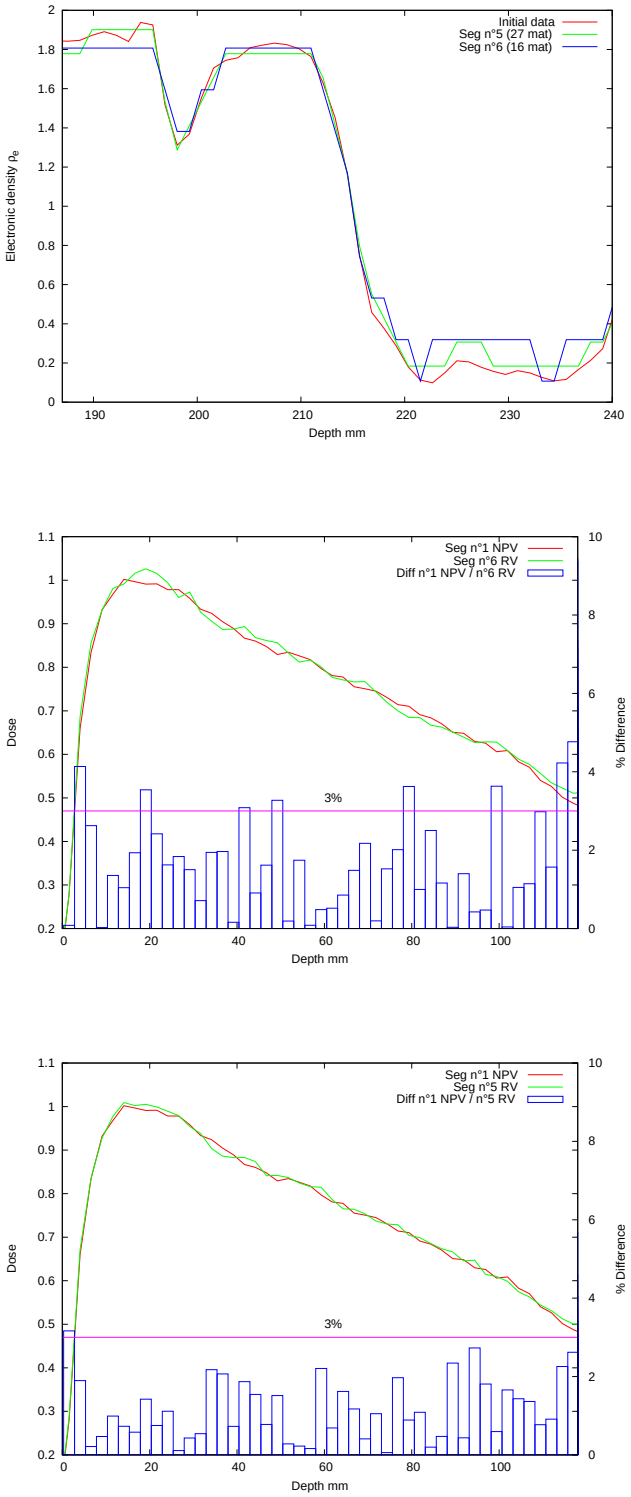


Figure 6: (Top) electronic density profiles for initial image data and tests n°5 and 6. (Bottom) deposited dose profiles for test n°1 with NPV and tests n°5 and 6 with RV. Corresponding differences are shown with boxes.

region density variance). Generally, segmentation combines a homogeneity criterion (internal energy) with a shape regularization (external energy) allowing to introduce a certain a priori in the region shape. Finding the best way to segment an image could be the subject of another study.

The proposed approach has been applied to the Geant4 code but should also be applicable to other simulation codes. It does not require to modify the transportation algorithm. It is not dedicated to accelerating radiotherapy simulations only and is also applicable to other simulations, for example PET imaging with the GATE software [6]. Finally, even with the proposed technique, complete simulation still remains relatively slow. Other acceleration techniques (variance reduction, tracking cuts) must be used to further speed up the whole process. The particle navigation algorithm [7] should still be optimized with the proposed SV approach. It would accelerate the time reduction. A discrete ray-tracing algorithm faster than the simple one we used should also contribute to further decrease computational time. The method proposed here is available within an open source licence on the following website http://www.creatis.insa-lyon.fr/rio/ThIS . The proposed method has been proposed to the OpenGate collaboration for further study and will be introduced in a future release of the GATE [6] platform.

References

[1] J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce Dubois, M. Asai, G. Barrand, R. Capra, S. Chauvie, R. Chytracek, G.A.P. Cirrone, G. Cooperman, G. Cosmo, G. Cuttone, G.G. Daquino, M. Donszelmann, M. Dressel, G. Folger, F. Foppiano, J. Generowicz, V. Grichine, S. Guatelli, P. Gumplinger, A. Heikkinen, I. Hrivnacova, A. Howard, S. Incerti, V. Ivanchenko, T. Johnson, F. Jones, T. Koi, R. Kokoulin, M. Kossov, H. Kurashige, V. Lara, S. Larsson, F. Lei, O. Link, F. Longo, M. Maire, A. Mantero, B. Mascialino, I. McLaren, P. Mendez Lorenzo, K. Minamimoto, K. Murakami, P. Nieminen, L. Pandola, S. Parlati, L. Peralta, J. Perl, A. Pfeiffer, M.G. Pia, A. Ribon, P. Rodrigues, G. Russo, S. Sadilov, G. Santin, T. Sasaki, D. Smith, N. Starkov, S. Tanaka, E. Tcherniaev, B. Tome, A. Trindade, P. Truscott, L. Urban, M. Verderi, A. Walkden, J.P. Wellisch, D.C. Williams, D. Wright, and H. Yoshida. Geant4 developments and applications. *Nuclear Science, IEEE Transactions on*, 53(1):270–278, Feb. 2006.

[2] I.J. Chetty, M. Rosu, M.L. Kessler, B.A. Fraass, R.K. Ten Haken, F.M.S K., and D.L. McShan. Reporting and analyzing statistical uncertainties in monte carlo-based treatment planning. *Int J Radiat Oncol Biol Phys*, 65(4):1249–1259, Jul 2006.

[3] P.E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.

[4] N. Freud, J.M Létang, and D. Babot. A hybrid approach to simulate X-ray imaging techniques, combining Monte Carlo and deterministic algorithms. *IEEE Transactions on Nuclear Science*, 52(5):1329–1334, oct 2005.

[5] V. Hubert-Tremblay, L. Archambault, D. Tubic, R. Roy, and L. Beaulieu. Octree indexing of dicom images for voxel number reduction and improvement of monte carlo simulation computing efficiency. *Med Phys*, 33(8):2819–31, July 2006.

[6] S. Jan, G. Santin, D. Strul, S. Staelens, K. Assié, D. Autret, S. Avner, R. Barbier, M. Bardiès, P. M. Bloomfield, D. Brasse, V. Breton, P. Bruyndonckx, I. Buvat, A. F. Chatziioannou, Y. Choi, Y. H. Chung, C. Comtat, D. Donnarieix, L. Ferrer, S. J. Glick, C. J. Groiselle, D. Guez, P. F. Honore, S. Kerhoas-Cavata, A. S. Kirov, V. Kohli, M. Koole, M. Krieguer, D. J. van der Laan, F. Lamare, G. Largeron, C. Lartizien, D. Lazaro, M. C. Maas, L. Maigne, F. Mayet, F. Melot, C. Merheb, E. Pennacchio, J. Perez, U. Pietrzyk, F. R. Rannou, M. Rey, D. R. Schaart, C. R. Schmidtlein, L. Simon, T. Y. Song, J. M. Vieira, D. Visvikis, R. Van de Walle, E. Wieërs, and C. Morel. Gate: a simulation toolkit for pet and spect. *Phys Med Biol*, 49(19):4543–4561, Oct 2004.

[7] H. Jiang and H. Paganetti. Adaptation of GEANT4 to monte carlo dose calculations based on CT data. *Med Phys*, 31(10):2811–2818, September 2004.

[8] N. Kanematsu, N. Matsufuji, R. Kohno, S. Minohara, and T. Kanai. A CT calibration method based on the polybinary tissue model for radiotherapy treatment planning. *Phys Med Biol*, 48(8):1053–1064, April 2003.

[9] M. Ljungberg, S.E. Strand, and M.A. King. *Monte-Carlo Calculation in nuclear medicine*. Medical Science Series. Institute of Physics Publishing, Bristol and Philadelphia, 1998.

[10] C. M. Ma, J. S. Li, T. Pawlicki, S. B. Jiang, J. Deng, M. C. Lee, T. Koumrian, M. Luxton, and S. Brain. A monte carlo dose calculation tool for radiotherapy treatment planning. *Phys Med Biol*, 47(10):1671–1689, May 2002.

[11] H. Paganetti, H. Jiang, JA. Adams, GT. Chen, and E. Rietzel. Monte carlo simulations with time-dependent geometries to investigate effects of organ motion with high temporal resolution. *Int J Radiat Oncol Biol Phys*, 60(3):942–950, November 2004.

[12] J. Peter, M.P. Tornai, and R.J. Jaszczak. Analytical versus voxelized phantom representation for monte carlo simulation in radiological imaging. *Medical Imaging, IEEE Transactions on*, 19(5):556–564, May 2000.

[13] E. Poon, J. Seuntjens, and F. Verhaegen. Consistency test of the electron transport algorithm in the GEANT4 Monte Carlo code. *Phys Med Biol*, 50(4):681–694, February 2005.

[14] E. Poon and F. Verhaegen. Accuracy of the photon and electron physics in Geant4 for radiotherapy applications. *Med Phys*, 32(6):1696–1711, June 2005.

[15] I. Pshenichnov, I. Mishustin, and W. Greiner. Neutrons from fragmentation of light nuclei in tissue-like media: a study with the geant4 toolkit. *Phys Med Biol*, 50(23):5493–507, December 2005.

[16] D.W.O Rogers and R. Mohan. Questions for comparison of clinical monte carlo codes. In W. Schlegel and T. Bortfeld, editors, *13th Int. Conf. on the Use of Computer in Radiation Therapy (ICCR)*, Springer, pages 120–122, Heidelberg, 2000.

[17] S. Agostinelli et al. Geant4 - a simulation toolkit. *Nuclear Instruments and Methods*, A506(3):250–303, July 2003.

[18] D. Sarrut, V. Boldea, S. Miguet, and C. Ginestet. Simulation of 4D CT images from deformable registration between inhale and exhale breath-hold CT scans. *Med Phys*, 33(3):605–617, March 2006.

[19] B. Schaffner and E. Pedroni. The precision of proton range calculations in proton radiotherapy treatment planning: experimental verification of the relation between ct-hu and proton stopping power. *Phys Med Biol*, 43(6):1579–1592, Jun 1998.

[20] U. Schneider, E. Pedroni, and A. Lomax. The calibration of CT Hounsfiled units for radiotherapy treatment planning. *Phys Med Biol*, 41:111–124, 1996.

[21] W. Schneider, T. Bortfeld, and W. Schlegel. Correlation between CT numbers and tissue parameters needed for Monte Carlo simulations of clinical dose distributions. *Phys Med Biol*, 45(2):459–478, February 2000.

[22] J. Sempau, A. Sánchez-Reyes, F. Salvat, H. O. ben Tahar, S. B. Jiang, and J. M. Fernández-Varea. Monte carlo simulation of electron beams from an accelerator head using penelope. *Phys Med Biol*, 46(4):1163–1186, Apr 2001.

[23] B. De Smedt, N. Reynaert, W. De Neve, and H. Thierens. Dosscore: an accelerated dosxyznrc code with an efficient stepping algorithm and scoring grid. *Phys Med Biol*, 49(19):4623–4635, Oct 2004.

[24] B. De Smedt, B. Vanderstraeten, N. Reynaert, W. De Neve, and H. Thierens. Investigation of geometrical and scoring grid resolution for monte carlo dose calculations for imrt. *Phys Med Biol*, 50(17):4005–4019, Sep 2005.

[25] F. Verhaegen and S. Devic. Sensitivity study for CT image use in Monte Carlo treatment planning. *Phys Med Biol*, 50(5):937–946, March 2005. Special Issue: International Workshop on current topics in Monte Carlo treatment planning.

[26] B. R B Walters, I. Kawrakow, and D. W O Rogers. History by history statistical estimators in the beam code system. *Med Phys*, 29(12):2745–2752, Dec 2002.

[27] L. Wang, C. S. Chui, and M. Lovelock. A patient-specific monte carlo dose-calculation method for photon beams. *Med Phys*, 25(6):867–878, Jun 1998.