Enhancing the speed of radiotherapy Monte Carlo dose calculation with applications in dose verification

by

Reid William Townson
B.Sc., University of Victoria, 2010

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Physics and Astronomy

Enhancing the speed of radiotherapy Monte Carlo dose calculation with applications in dose verification

by

Reid William Townson
B.Sc., University of Victoria, 2010

Supervisory Committee

Dr. S. Zavgorodni, Co-supervisor
(Department of Physics and Astronomy)

Dr. A. Jirasek, Co-supervisor
(Department of Physics and Astronomy)

Dr. W. Ansbacher, Departmental Member
(Department of Physics and Astronomy)

Dr. D. Karlen, Departmental Member
(Department of Physics and Astronomy)

Dr. V. King, Outside Member
(Department of Computer Science)

Supervisory Committee

Dr. S. Zavgorodni, Co-supervisor
(Department of Physics and Astronomy)

Dr. A. Jirasek, Co-supervisor
(Department of Physics and Astronomy)

Dr. W. Ansbacher, Departmental Member
(Department of Physics and Astronomy)

Dr. D. Karlen, Departmental Member
(Department of Physics and Astronomy)

Dr. V. King, Outside Member
(Department of Computer Science)

ABSTRACT

Monte Carlo (MC) methods for radiotherapy dose calculation are widely accepted as capable of achieving high accuracy. In particular, MC calculations have been demonstrated to successfully reproduce measured dose distributions in complex situations where alternative dose calculation algorithms failed (for example, regions of charged particle disequilibrium). For this reason, MC methods are likely to play a central role in radiotherapy dose calculations and dose verification in the future. However, clinical implementations of MC calculations have typically been limited due to the high computational demands. In order to improve the feasibility of using MC simulations clinically, the simulation techniques must be made more efficient.

This dissertation presents a number of approaches to improve the efficiency of MC dose calculations. One of the most time consuming parts of source modeling is the simulation of the secondary collimators, which absorb particles to define the rectangular boundaries of radiation fields. The approximation of assuming negligible transmission through and scatter from the secondary collimators was evaluated for accuracy and efficiency using both graphics processing unit (GPU)-based and central processing unit (CPU)-based MC approaches. The new dose calculation engine, gDPM, that utilizes GPUs to perform MC simulations was developed to a state where accuracy comparable to conventional MC algorithms was attained. However, in GPU-based dose calculation, source modeling was found to be an efficiency bottleneck. To address this, a sorted phase-space source model was implemented (the phase-space-let, or PSL model), as well as a hybrid source model where a phase-space source was used only for extra-focal radiation and a point source modeled focal source photons. All of these methods produced results comparable with standard CPU-based MC simulations in minutes, rather than hours, of calculation time. While maintaining reasonable accuracy, the hybrid source model increased source generation time by a factor of $\sim$2-5 when compared with the PSL source model. A variance reduction technique known as photon splitting was also implemented into gDPM, to evaluate its effectiveness at reducing simulation times in GPU calculations.

Finally, an alternative CPU-based MC dose calculation technique was presented for specific applications in pre-treatment dose verification. The method avoids the requirement of plan-specific MC simulations. Using measurements from an electronic portal imaging device (EPID), pre-calculated MC beamlets in a spherical water phantom were modulated to obtain a dose reconstruction.

Contents

List of Tables

List of Figures

ACRONYMS

AAA anisotropic analytical algorithm

APR azimuthal particle redistribution

BCCA British Columbia Cancer Agency

CCC collapsed cone convolution

CSDA continuous slowing down approximation

CSI Centre for the Southern Interior

CT computed tomography

CTV clinical treatment volume

DNA deoxyribonucleic acid

DPM Dose Planning Method

DTA distance-to-agreement

ECUT electron energy cut-off

EGS Electron Gamma Shower

EPI electronic portal image

EPID electronic portal imaging device

FFF flattening filter free

GPU graphics processing unit

gDPM GPU Dose Planning Method

HDD hard-disk drive

IAEA International Atomic Energy Agency

ICRU International Commission on Radiation Units

IMRT intensity modulated radiation therapy

LBTE linear Boltzmann transport equation

linac linear accelerator

MC Monte Carlo

MFP mean-free-path

MLC multi-leaf collimator

MU monitor unit

OAR organ at risk

PCUT photon energy cut-off

PDF probability density function

PHSA Provincial Health Services Authority

PhspA patient-independent phase-space

PhspB patient-dependent phase-space

PhspC phase-space collimation

PSL phase-space-let

PSM phase-space modulation

PTV planning target volume

QA quality assurance

RMSD root mean square deviation

ROF relative output factor

RT radiation therapy

SAD source-to-axis distance

SDM spherical doselet modulation

SID source-to-image distance

SIMD single-instruction-multi-data

SSD source-to-surface distance

STOPS simultaneous transport of particle sets

TMR tissue maximum ratio

TPS treatment planning system

VCC Vancouver Cancer Centre

VIC Vancouer Island Centre

VIMC Vancouver Island Monte Carlo

VMAT volumetric modulated arc therapy

VRT variance reduction technique

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor Dr. Sergei Zavgorodni for his unwavering support throughout the majority of my educational development. It was his guidance and encouragement of independence that inspired my interest in medical physics since early undergrad. Ever since, his confidence in my abilities has continued to motivate me to strive for excellence. Sergei always had my best interests in mind, and has instilled a dedication to ethical science that will forever remain a foundation of my practice.

I would like to thank Dr. Wayne Beckham for his role in supporting my research at the BC Cancer Agency, and for modeling what excellent leadership should look like. His profound logic and understanding, coupled with genuine charisma is an inspiration to those he works with. I have the utmost respect for Wayne, and am thankful to have experienced the positive workplace community he has created.

Throughout my career in medical physics, Dr. Karl Bush been an invaluable supporter of my research and motivated ambitious goals. Particularly in the early years, Karl was an unmatched source of meticulous guidance and advice. As both a mentor and a friend, Karl encouraged me to take on challenges I might not have otherwise attempted, always with positive results. Our research collaborations were some of the most enjoyable projects of my career, and I hope we are able to partake in additional mutual endeavors in the future.

I am indebted to Dr. Andrew Jirasek for his advice over the years, assisting with my progress throughout the medical physics program as well as my research. I would also like to give heartfelt thanks to Dr. Steve Jiang and Dr. Xun Jia for welcoming me into their research group for collaboration. Over the short time that I spent at the University of California: San Diego, I felt supported and valued in a meaningful way.

All of this would not be possible without the incredible efforts of my parents, Ron and Barb Townson, who have been an endless source of support throughout my life. As humble as they may be, I cannot express enough my appreciation of their parenting prowess. I would also like to thank my extended family Tia, Bob, Carson, Devon and Mackenzie Farmer, for making me feel like such a welcome part of the family during my stay in Victoria. Finally, I would like to thank my grandma, Isabel Farmer, for her enduring encouragement, support and exceptional humor over all these years.

I give my sincere thanks to all of the other graduate students at the BC Cancer

Agency, who have been an immeasurable source of assistance, reassurance and escape throughout the highs and lows of my studies.

Chapter 1

Introduction

1.1 External beam radiation therapy

Approximately 187,600 new cases of cancer and 75,500 deaths from cancer occurred in Canada in 2013 (Canadian Cancer Society's Advisory Committee on Cancer Statistics, 2013 [61]). It is expected that 2 in 5 Canadians will develop cancer in their lifetime. In roughly half of new cancer cases, external beam radiation therapy (RT) is prescribed at some point in the treatment. External beam RT refers to radiotherapy performed using a radiation source external to the patient. Tumour control is typically achieved through the application of RT combined with chemotherapy, immunotherapy, hormone treatments, transplantation techniques and/or surgical removal of cancerous materials. Radiation therapy is an effective curative treatment for many types of cancer, and can also provide valuable non-curative palliation (pain relief) for patients. Modern efforts to improve RT outcomes include:

1. raising the standards of radiation dose prescription and delivery;
2. improving dose distributions by conformal techniques;
3. integrating image-guidance into treatments;
4. exploiting radiobiological dose personalization initiatives.

Many aspects of these efforts would benefit from more accurate and efficient radiotherapy computer simulation schemes. In this work, I will focus on improvements of RT delivery standards through investigation of efficient dose calculation and verification techniques.

The most common form of RT involves high energy photons (produced by 4-25MV[1] linear accelerators). These machines produce photons with a spectrum of energies, and most of these beams have the capability to ionize (eject electrons from) atoms. The primary biologically damaging effects from ionizing radiation result from this ability. In particular, ionized molecules are reactive and rapidly undergo chemical changes and break bonds. Ejected electrons may cause further ionizations and excitations as they interact and lose energy.

As an electron loses energy, the number of interactions increases in frequency, resulting in clusters of ionizations just before it is absorbed. The scale of these ionizations is such that several can occur within a few base pairs of deoxyribonucleic acid (DNA). Energy deposition in DNA can result in a variety of damage and initiate cell repair pathways. DNA is quite large, important to cellular functions, and present in only two copies (in contrast to most other molecules in the cell that have many copies). As a result, damage to DNA is often lethal to the cell.

The damaging effects of radiation can be characterized using the quantity absorbed dose, D. This quantity is defined as the mean energy imparted by a radiation, $d\epsilon$, to a unit mass of medium dm:

$$D = \frac{d\epsilon}{dm}. \tag{1.1}$$

Absorbed dose is measured in units of Gray (Gy), where 1 Gy = 1 J/kg. When discussing dose deposition in this work, I will be referring to the absorbed dose. Note that this is different from the energy *transferred* to a given region from indirectly ionizing radiation (photons, neutrons), since some of the transferred energy can escape from the region in the form of bremsstrahlung or high energy electrons. The quantity characterizing this is the kinetic energy released per unit mass, or kerma, K. The kerma is the sum of the initial kinetic energies of all the ionizing charged particles liberated by uncharged ionizing particles.

$$K = \frac{dE_{tr}}{dm}, \tag{1.2}$$

where dE_{tr} is the energy transferred to some element of mass dm. At low energies (e.g. <4.5 MeV linear-accelerator radiations in water), absorbed dose and kerma are

[1]When discussing polyenergetic photon beams, it is convention to refer to a beam by the maximum photon energy E_{max} it contains. When this is done, the beam is cited as an MV beam instead of MeV, since units of energy would signify a monoenergetic beam. The photons in a E_{max} MV beam has a spectrum of energies from 0-E_{max}.

nearly equal on the falling part of a depth dose curve, but at higher energies they diverge as the incidence of bremsstrahlung and energy of ejected electrons increases.

External beam radiation therapy aims to maximize the death of tumour cells while minimizing normal tissue cell death. Modern RT techniques attempt to achieve this by geometrically localizing ionizing radiation as much as possible to the tumour tissue. However, as is evident in its name, external beam radiation therapy requires the radiation to enter the patient externally, and this causes some dose to be delivered to healthy tissues. As radiation dose increases, there will be both increased tumour and normal tissue response. In order to achieve a beneficial effect, endpoints for evaluation must be chosen and considered statistically using large sample populations. For example, consider the endpoint of the tumour to be *local control*, that is, absence of tumour regrowth over the normal lifespan of the patient. For normal tissues, a useful measure of response depends on the tissue in question and could range from mild discomfort to life-threatening complications. The curves statistically relating tissue response to dose for both tumour and normal cells to dose are sigmoid shaped. For any fixed level of normal-tissue damage, there will be an associated local tumour control. This provides a simple way to compare the toxicity and benefit of various treatment techniques.

1.1.1 The linear accelerator

Linear accelerators use a waveguide to accelerate electrons to MeV energies (figure 1.1). To produce photon beams, electrons are directed to impact a high atomic number target. X-ray production occurs through the Bremsstrahlung[2] process with a relatively low efficiency - most of the energy is lost as heat. The resulting photons are collimated into a conical diverging beam by the primary collimator (usually made of lead or tungsten) and filtered by passing through a flattening filter (figure 1.2). The main purpose of the flattening filter is to improve uniformity of the radiation dose in a treated volume. However, the increased ability for modern linacs to precisely shape the treatment beam and increased use of small field sizes has led to flattening filter free (FFF) treatments becoming more common. The primary benefit of FFF treatments is a 2-4 fold reduction in treatment times, due to the higher output of the beam. Downstream of the flattening filter is the monitor chamber, a specialized ionization

[2]Bremsstrahlung radiation is the result of energy conservation when electrons are deflected and decelerate near atomic nuclei.

chamber that is used to measure the instantaneous and integral dose rates of the beam, as well as monitor beam symmetry. The monitor chamber is comprised of a number of sectors along the beam direction to allow for beam symmetry measurements, which are then employed to make alignment corrections using the electron beam steering magnets. The beam-on time is controlled by the monitor chamber, which signals when the requested number of monitor units (MUs) have been measured. A MU is a measure of machine output, defined to produce a particular absorbed dose under calibration conditions.

In this text, the configuration of the target, primary collimator, flattening filter and monitor chamber components will be referred to as the *plan-independent*[3] part of the linac geometry, or the upper portion of the linac head.

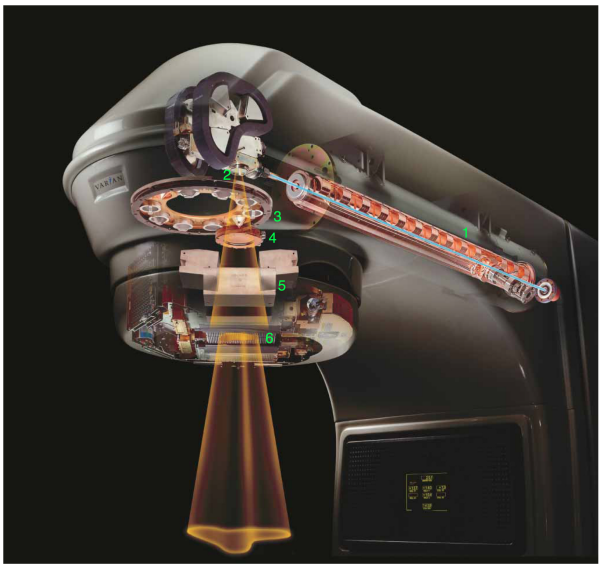

Figure 1.1: The internals of a Varian 21EX Clinac (Varian Medical Systems, Palo Alto, CA, USA). The numbered components are the (1) waveguide where electrons are accelerated, (2) target and primary collimator, (3) flattening filter and foil carousel, (4) monitor chamber, (5) secondary collimators and (6) multi-leaf collimators.

The secondary collimators (or jaws) and multi-leaf collimators (MLCs) produce the *plan-dependent* beam shaping apertures (figures 1.3 and 1.4). The purpose of these collimators is to absorb radiation that strikes them, shaping the treatment field

[3]This is a slight misnomer since the target and flattening filter can be swapped out for different photon beam energies, FFF treatments, and for electron beams. Thus, their configuration is not actually independent of a given treatment plan. However, this terminology is convenient in the context of Monte Carlo simulations in this dissertation, since these configurations can effectively be treated as entirely different treatment machines.

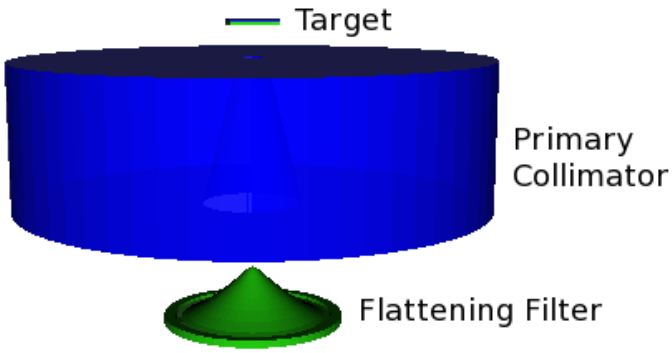

Figure 1.2: An illustration of the target, primary collimator and flattening filter for a typical 6MV photon linac. While it is difficult to see in this image, the primary collimator has a conical hollow centre to collimate photons into a diverging beam.

to conform to the tumour shape and avoid healthy tissues. The secondary collimators define a rectangular field, that determines the *field size*. Historically the field size was held constant during a particular radiotherapy treatment, but modern treatments may include motion of the secondary collimators throughout treatment (i.e. jaw tracking). The secondary collimators are comprised of two pairs of high-Z slabs (e.g. tungsten), each pair collimating along an axis perpendicular to the beam axis. They are able to create a field size up to a maximum of about 40×40 cm^2 at the isocentre. The MLC is a relatively recent addition to the modern medical linear accelerator, and contains two parallel opposing banks of mobile high-Z slabs (or *leaves*), each attached to a computer-controlled motor. The MLC enables precise conformal dose distributions, often using dynamic leaf motion during a single treatment beam. Thinner leaves are normally used near the centre of the MLC device in order to increase collimation resolution, despite slightly increasing the radiation transmission through the small gaps between leaves (inter-leaf leakage).

Medical linear accelerators are typically mounted on a gantry that allows for rotations about the patient. The point along this axis of gantry rotation that intersects the beam axis is called the *isocentre*. The distance from the top of the target to the isocentre is called the source-to-axis distance (SAD). Treating the patient from multiple angles enables dose escalation in the tumour due to the entrance dose being spread over a larger volume in the patient. Other degrees of freedom include rotations of the couch that the patient is positioned on (referred to as non-coplanar treatments), and rotations of the secondary collimators. All of these rotations are about an axis through the isocentre. Finally, is it also possible to translate the treat-

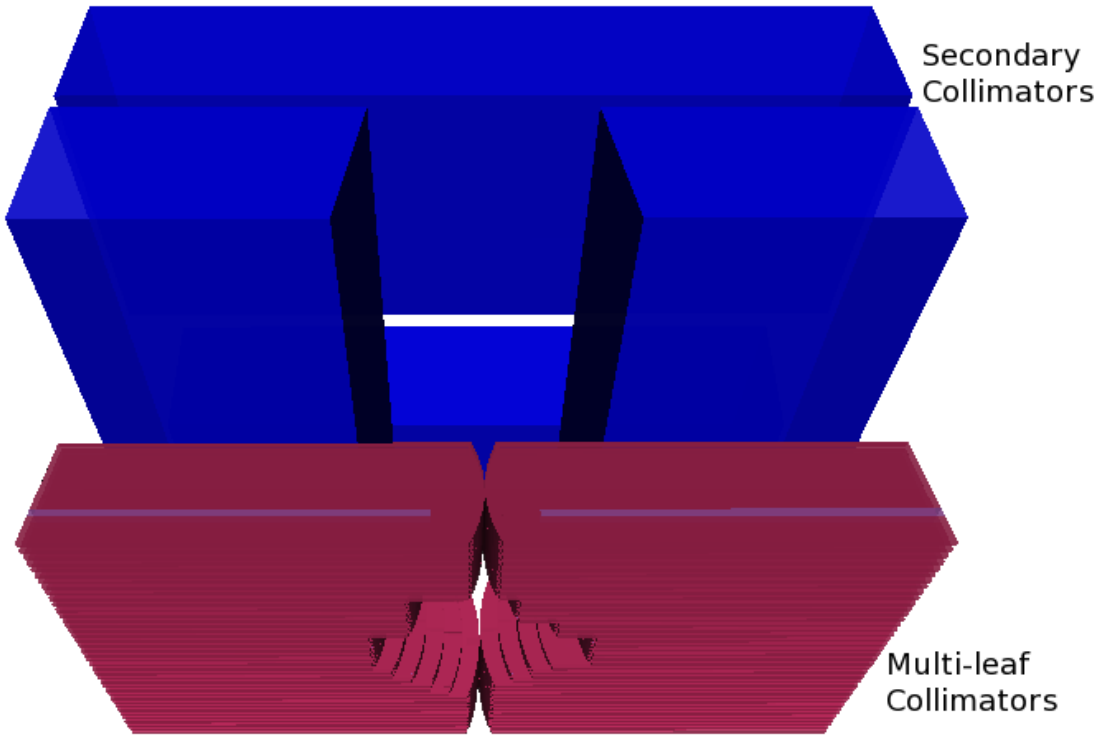

Figure 1.3: The *plan-dependent* beam shaping components of a medical linear accelerator. Pictured are the secondary collimators and multi-leaf collimator for a Varian linac.

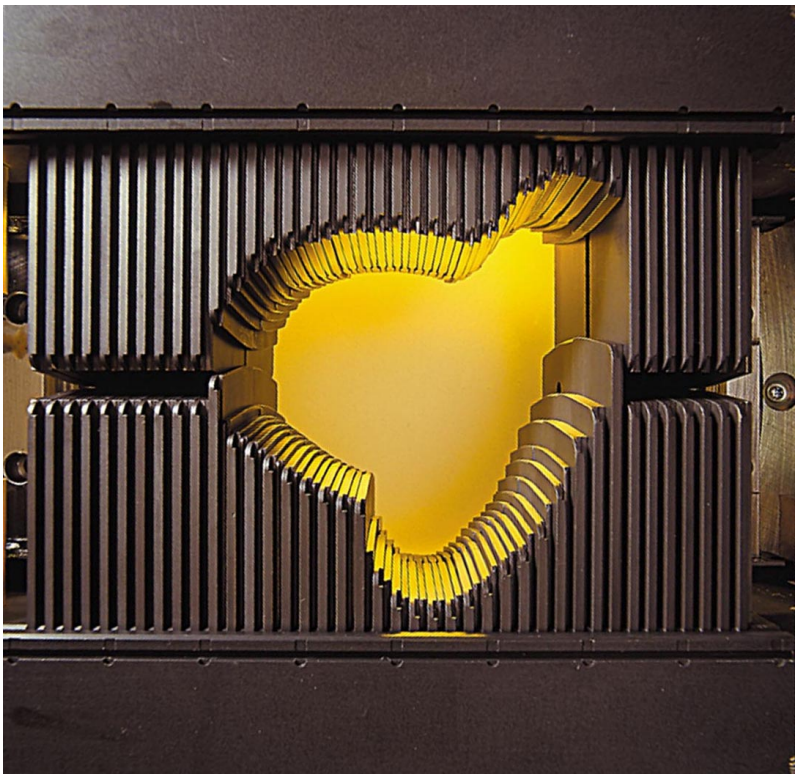

Figure 1.4: An image of Varian multi-leaf collimators (Varian Medical Systems, Palo Alto, CA).

ment couch. In recent years, these degrees have freedom are being used increasingly, adding complexity in order to improve conformity of radiotherapy treatments. In particular, improved mechanical, planning and verification capabilities are gradually allowing for some of these parameters to be varied continuously during treatment.

1.1.2 Modern radiation therapy techniques

The development of the MLC along with software able to calculate the associated dose delivery lead to treatment techniques known as intensity modulated radiation therapy (IMRT). With a stationary gantry for each beam, motion of the MLC leaves creates a finely shaped radiation field, modulating the photon field intensity. The leaf movement can occur either when the beam is on (dynamic MLC), or off in a series of static apertures. The improved conformity of radiation fields to tumour shape leads to reduced dose to healthy tissues, but the substantial collimation leads to decreased efficiency of dose per MU and increased leakage dose to the patient body.

One adaptation of IMRT is helical tomotherapy, where radiation is delivered through a narrow slit beam (approximate 2 cm $\times$ 20 cm) while specially designed gantry rotates continuously about the patient many times (Mackie *et al.*, 1993 [55]). In modern machines, due to continuous couch motion, the beam trajectory through the patient body makes a helical path, improving the dose uniformity and treatment delivery time. This arguably provides greater flexibility than linac-based IMRT in conforming the dose in 3-dimensions (Bortfeld and Webb, 2008 [9]).

An alternative to tomotherapy is volumetric modulated arc therapy (VMAT), where gantry rotation is utilized on a standard IMRT radiotherapy machine, with dynamic MLC motion (Otto, 2008 [62]). A typical treatment can usually be completed using only 1 or 2 arcs. VMAT generally produces more open apertures than IMRT, resulting in reduced MUs, shorter treatment times, reduced scatter absorbed in the patient and less patient motion during treatment. VMAT is widely used and able to achieve high quality treatment plans (Gagne *et al.*, 2008 [26], Nicolini *et al.*, 2008 [60], Zimmerman *et al.*, 2009 [102]).

1.1.3 The treatment planning and delivery process

Radiotherapy treatment planning generally begins with patient imaging. Computed tomography (CT) scans of the patient provide geometrical and density information that is necessary for dose calculations. The patient's oncologist outlines relevant struc-

tures in the patient image, recommends the treatment technique, and prescribes dose limits. Radiation therapists and medical physicists use this information to develop a treatment plan using commercial treatment planning software. Modern radiotherapy techniques use 'inverse' planning methods to obtain optimal dose distributions. In this process, objectives are defined based on the prescribed dose to the planning target volume (PTV) and dose constraints to nearby organs at risk (OARs). The inverse planning algorithm searches for the optimal MLC motions to best achieve these goals. For IMRT, the number of fields, gantry, couch and collimator angles are selected by therapists. However, sufficiently advanced inverse planning algorithms could potentially optimize over these additional variables as well as conformal jaw motion (jaw tracking), beam energy, beam filters (e.g. FFF), beam type (photons, electrons, etc.), number of beams and variation of the beam direction (gantry, collimator and couch angles). These optimizations are an area of ongoing research. After optimization, the final planning dose calculation should be performed using the most accurate dose calculation algorithm clinically available that is able to complete calculations on a reasonable time scale (section 2.1).

International Commission on Radiation Units (ICRU) Report 62 provides recommendations on the volumes and absorbed doses that are important in photon beam therapy. The ICRU Report 29 recommends that radiation dose is delivered to within 5% of the prescribed dose. To achieve this, the uncertainty in the treatment planning dose calculations (and each step of the treatment process) must be significantly less than 5%. Since errors in treatment planning dose calculations manifest systematically, there is the possibility for significant adverse effect on dose delivery. Therefore, it is typically mandated that a medical physicist must perform a dose validation check for each patient using an independent technique as a part of quality assurance (QA). This check provides confirmation that the dose calculation algorithm used in the treatment planning system produced the expected result for the given beam configuration. Depending on the conventions of the institution and complexity of the plan, this check may be performed by hand-calculation, using dose calculation software independent from the treatment planning system (TPS), or measurement.

MC algorithms achieve remarkable dose calculation accuracy and are widely accepted as the most accurate method of dose calculation (Boyer and Mok, 1985 [10], Mackie *et al.*, 1985 [56], Knoos *et al.*, 1995 [48], Wieslander and Knoos, 2000 [96], Cranmer-Sargison *et al.*, 2004 [18], Vanderstraeten *et al.*, 2006 [91], Pratx and Xing, 2011 [65]). For this reason, MC techniques are expected to play a substantial role

in radiotherapy treatment planning, optimization and verification schemes for the foreseeable future. Since MC dose calculations are computationally intense and require complex implementation schemes, the clinical usage in TPSs has been limited to date. However, the high accuracy of MC software in complex cases, when compared to alternative dose calculation algorithms, makes it a sought after technology for a variety of applications. As the techniques for MC simulations are made more efficient, the subsequent integration into the radiotherapy process will result in the improved clinical outcomes (Xing *et al.*, 2009 [97], Pratx and Xing, 2011 [65]).

1.2 Dissertation scope

Accuracy of dose delivery is of primordial importance in radiation therapy. A key component of accurate delivery is the ability to calculate the dose produced by a given treatment in a virtual representation of the patient. This calculation is performed by a dose calculation algorithm, of which there are many options with varying accuracy and complexity. The most accurate method and gold standard in the field, Monte Carlo techniques for the simulation of radiation interactions in the linear accelerator treatment head and patient geometry, is therefore an important tool. The Monte Carlo approach to dose calculation applies stochastic modeling of individual particle interactions occurring from transport of radiation through matter. As a result, the statistical uncertainty in the calculated dose is reduced when a greater number of particles are transported. The efficiency of these simulations is therefore based both on the achieved variance and the calculation time (section 2.2.6) for a given scenario. The objective of this dissertation is to improve the efficiency of radiotherapy MC simulations while maintaining a clinically acceptable level of accuracy, at least within a certain range of applications. A variety of methods were employed: physical approximations, leveraging graphics processing unit (GPU) hardware, variance reduction techniques and alternative algorithms. These are briefly outlined in the following paragraphs.

In radiotherapy, MC simulations for a given patient usually begin by using a source of particles located just above the secondary collimators (this is described in more detail in section 2.2). The source is usually either a 'phase-space' file containing a list of particles obtained from previous simulations of the static components of the linear accelerator, or an analytical model used to dynamically generate particles on-the-fly. When phase-space files are produced with sufficiently high particle density for the ap-

plication, they are an exact characterization of the source (unlike analytical models). One of the difficulties with source modeling, in terms of efficiency, involves the reality that many of the source particles will be subsequently absorbed by the secondary collimators and do not contribute dose to the volume of interest (the virtual phantom of the patient). This results in relatively low efficiency simulations, particularly for small field sizes. In chapter 3, a hypothesis is presented that efficiency enhancement can be achieved by assuming 100% absorption of all particles striking the secondary collimators without substantial loss of accuracy.

Chapter 4 introduces GPUs as a hardware-based approach to MC simulation efficiency enhancement. MC simulations of radiotherapy can be parallelized, a quality which translates into high efficiency on GPUs. By off-loading the parallelized computations to GPUs, the number of physical computers within a computing cluster can be reduced, resulting in lower maintenance, space and power requirements (often valuable benefits for hospitals that require hardware to be hosted locally). However, translation of CPU-based algorithms to GPUs is not straight-forward and can require significant innovation to re-design algorithms with efficiency in mind. In this dissertation, the source model was found to be of particular importance to the overall dose calculation time, so methods to improve source modeling efficiency were evaluated. In order to benchmark the accuracy of the GPU-based code against its CPU-based competitors, phase-space source modeling was implemented. Following this, further GPU-based efficiency enhancements were investigated through pre-processing of the phase-space (chapter 4), hybridization of the phase-space with an analytical point source model (chapter 5), and integration of the photon-splitting variance reduction technique (chapter 6). The objectives of these chapters were to (1) demonstrate that GPU-based MC dose calculations are competitive with CPU-based methods in terms of speed and accuracy, and (2) develop and test methods to further improve the efficiency of GPU-based dose calculations.

One of the important applications for high-speed dose calculations is verification of dose delivery procedures (comparing dose distributions reconstructed using measurements or parameters from delivery with what was planned). In chapter 7, an algorithm utilizing pre-calculated Monte Carlo sub-beam dose distributions (doselets) was developed for specific applications in pre-treatment dose verification. The signal from an electronic portal imaging device (EPID) was used to modulate the contributions of doselets according to pre-treatment delivery measurements, after correcting for scatter in the imaging device. The reconstructed dose distribution was then com-

pared with the predicted dose to provide verification of the delivery. The novelty of this method arises from the use of a spherical virtual water phantom, allowing for rotational symmetries to be leveraged for efficiency enhancement.

Finally, chapter 8 concludes the dissertation. Appendix A describes the Vancouver Island Monte Carlo (VIMC) web-based framework, WebMC, that was used for some of the MC simulations in the dissertation. In particular, the software developed in chapters 3 and 7 were integrated into WebMC. A description of how GPU-based dose calculations could be integrated into the system for future work is also discussed.

Chapter 2

Background

2.1 Dose calculation in the treatment planning system

There are various commercial TPSs available, each with different dose calculation algorithms. While MC codes are becoming available in some TPSs, this is a recent development. The (non-Monte Carlo) approaches follow either a correction-based or model-based scheme for dose calculation. Correction-based techniques are based on measured dose distributions in a water phantom, and use independent corrections for beam modifiers, surface contours and tissue heterogeneities. In contrast, model-based methods compute the dose directly in a 3D voxellized patient representation using computations based on beam intensity (e.g. energy fluence) rather than dose in a water phantom. Model-based algorithms also require the beam to be modeled explicitly[1]. In the following sections, some of the more widely used model-based algorithms will be outlined to provide a basis for comparison with Monte Carlo (MC) dose calculation.

To begin, let's consider the typical scheme for model-based dose calculations. Before the energy absorption process itself can be simulated, one first has to model the output of the treatment machine. This is done using a model for the primary energy fluence of the photons emerging from the linear accelerator. Since such a model must

[1]Model-based algorithms can become quite complex in order to account for physical effects. For example, models should account for: finite source size, angular distribution of photons, primary transmission, extra-focal radiation, differential hardening of the beam by the flattening filter, curved multi-leaf collimator ends, leaf positions, leaf tongue and groove effects, leaf transmission, electron contamination and tissue heterogeneities.

be calibrated to match measured dose in water for simple treatment fields, it is not independent of the energy absorption physics used in the dose calculation algorithm. The primary photon energy fluence is used as input for the dose calculation, which is typically divided into two components: (1) absorption of primary photons and (2) transport of secondary electrons and photons. The former is considered using the total energy released per unit mass, or TERMA. The TERMA describes the loss of radiant energy from uncharged primaries as they interact in a material (deposited locally or at a distance). The interactions of a photon of an energy E contribute to the total attenuation coefficient $\mu(E)$ as a sum of the attenuation coefficients of the various interactions in a given material

$$\mu(E) = \sigma_{Rayleigh}(E) + \sigma_{Compton}(E) + \tau(E) + \kappa(E) + \nu(E), \tag{2.1}$$

where the attenuation coefficients of Rayleigh scattering, Compton scattering, photo-electric absorption, pair production, and photo-nuclear interactions are $\sigma_{Rayleigh}$, $\sigma_{Compton}$, τ, κ, and ν, respectively.

To characterize a photon beam, a useful quantity is the particle fluence Φ. This is the number of particles dN incident on a sphere of cross-sectional area dA, and has units of m^{-2}:

$$\Phi = \frac{dN}{dA}. \tag{2.2}$$

The particle fluence is independent of the direction in the incident particles, whereas the planar particle fluence is defined by the number of particles incident on a plane per unit area, and depends on the angle of incidence. The energy fluence Ψ is the radiant energy incident on a sphere of cross-sectional area dA, and can be calculated from the particle fluence for particles with energy E in units of $\mathrm{J/m}^2$:

$$\Psi = \frac{dE}{dA} = \frac{dN}{dA}E = \Phi E. \tag{2.3}$$

For polyenergetic beams, the particle fluence spectrum and energy fluence spectrum differential in energy E are described by $\Phi_E(E)$ and $\Psi_E(E)$:

$$\Phi_E(E) \equiv \frac{d\Phi}{dE}(E) \tag{2.4}$$

$$\Psi_E(E) \equiv \frac{d\Psi}{dE}(E) = \frac{d\Phi}{dE}(E)E \tag{2.5}$$

Consider the dose deposition of a monoenergetic, infinitely narrow photon beam of energy E and initial radially symmetric photon fluence Φ in water. The energy fluence Ψ of the primary photons at an interaction point $\vec{r}$ is (to a first approximation)

$$\Psi(\vec{r}) = \Phi(\vec{r_\perp}, 0) E e^{-\mu(E)}, \tag{2.6}$$

where $\vec{r_\perp}$ is the coordinate perpendicular to the beam-axis. The TERMA is then calculated using the rate of primary interactions in the medium of density ρ:

$$\text{TERMA}(\vec{r}) = \Psi(\vec{r}) \frac{\mu}{\rho}(\vec{r}). \tag{2.7}$$

The locally released energy described by the TERMA is subsequently available for transporting from the interaction point. In model or correction-based algorithms, the transport of secondaries is accounted for using dose kernels, or energy deposition density functions. The details of how each of the dose calculation components is performed varies greatly between algorithms, as does the resulting accuracy and efficiency (Boyer and Mok, 1985 [10], Mohan *et al.*, 1986 [58]). A simplistic approach uses convolution with a single scatter kernel, $A(\vec{r} - \vec{r'})$ to calculate the dose $D(\vec{r})$ at a position $\vec{r}$:

$$D(\vec{r}) = \int \Psi(\vec{r}) \frac{\mu}{\rho}(\vec{r}) A(\vec{r} - \vec{r'}) d^3\vec{r'} = \int \text{TERMA}(\vec{r}) A(\vec{r} - \vec{r'}) d^3\vec{r'}. \tag{2.8}$$

2.1.1 Anisotropic analytical algorithm

The anisotropic analytical algorithm (AAA) (Ulmer *et al.*, 2005 [86]) is available in the *Eclipse*^{TM} TPS (AAA, Varian, Palo Alto, CA, USA). In AAA, dose calculation requires a source model, customized by each user through a series of measurements in water on the local treatment machines. The source model builds a planar phase-space fluence distribution, and the contributions of spatial regions called beamlets are tuned according to measurements. That is, the phase-space is divided into a Cartesian grid of beamlets $\beta(x, y)$ which correspond to the source divergent along fan-lines. The beamlets are split into three source components: (1) focal photons (a circular or elliptical source at the target plane modeling the bremsstrahlung photons that do not interact in the treatment head), (2) extra-focal photons (a Gaussian planar source located at the bottom of the flattening filter modeling the photons that results from interactions not in the target), and (3) contaminant electrons. When hard wedges are

employed in treatment, an additional wedge photon scatter source is included using a dual Gaussian model.

Each beamlet is associated with a field intensity function[2] Φ. The field intensity is a normalized Cartesian map representing the transmission through beam shaping apertures (1 for an element of the map that is never blocked by a collimation device, and equal to a transmission factor $0 < T < 1$ for an obscured element).

The photon beam attenuation in media is modeled with energy deposition density function $I_\beta(z, \rho)$, where z is depth in a material of density ρ. To account for heterogeneities in density, the concept of radiological scaling is applied by using a radiological depth of z'. This is obtained by integration of the ratio of the density in a material with the density of water over the depth t,

$$z' = \int_0^z \frac{\rho(t)}{\rho_{water}} dt, \tag{2.9}$$

and setting $I_\beta(z, \rho) = I_\beta(z')$. A scatter kernel $K_\beta(x, y, z, \rho)$ defines lateral photon scatter. Both I_β and K_β and pre-calculated by MC simulations. The photon fluence is assumed to be uniform over the the lateral cross-section of a beamlet. This process is the same for both focal and extra-focal photons.

The dose distribution at coordinate (X, Y, Z) resulting from photons in beamlet β, $D_{ph,\beta}(X, Y, Z)$, is calculated by convolution of the scatter kernel with the energy deposition density and intensity functions. The convolution is a mathematical representation of the fact that the energy transfer from the incident beam to the medium is a two stage process: (1) the primary photon interacts in the medium, producing an electron, and (2) the electron transfers energy to the medium along its track.

$$D_{ph,\beta}(X, Y, Z) = \Phi_\beta * I_\beta(z, \rho) * \int\int_{Area(\beta)} K_\beta(u - x, v - y, z; \rho) du dv. \tag{2.10}$$

The final dose distribution D is a superposition of the individual dose components from the focal ($D_{ph\_focal,\beta}$), extra-focal ($D_{ph\_extra\_focal,\beta}$) and contaminating electrons ($D_{e-,\beta}$) for beamlet β:

$$D(X, Y, Z) = \sum_\beta (D_{ph\_focal,\beta}(X, Y, Z) + D_{ph\_extra\_focal,\beta}(X, Y, Z) + D_{e-,\beta}(X, Y, Z)) \tag{2.11}$$

[2]This is different from the ICRU definition of fluence as the number of photons per unit area.

While this method does construct the dose distribution in a 3D voxel grid, it is not a true 3D dose calculation technique, since scatter convolutions are performed two dimensionally. AAA has been shown to produce accurate dose distributions in homogeneous media (Fogliata *et al.*, 2006 [25], Gagne and Zavgorodni, 2007 [27]). However, this type of dose calculation algorithm tends to have inaccuracies in regions where charged particle equilibrium[3] does not exist. The regions of greatest charged particle disequilibrium are the build-up region, high dose gradients, field edges, shielding edges and inhomogeneity interfaces. In contrast, MC simulations are able to provide high accuracy in these regions.

2.1.2 Collapsed cone convolution

Another dose calculation algorithm is collapsed cone convolution (CCC). The dose is calculated in CCC by convolving the TERMA with an energy deposition kernel describing the energy deposited by secondary particles (charged particles and scattered photons).

The energy deposition kernel (e.g. point spread function) used in CCC is first calculated using Monte Carlo. The kernel is then divided into cones, each emanating from the source origin. The energy deposition for each kernel is collapsed to a line along the axis of the cone. All energy released into the coaxial cones, from volume elements on the cone axis, can be rectilinearly transported, attenuated, and deposited in elements on the axis. For M cones and N voxels, this changes the number of calculations required to construct the dose from N^3 for standard convolution to MN. Collapsing the cones removes the inverse square law, so that only exponential attenuation remains. For accurate dose calculation, the density of cones must be large enough to distribute energy to all or most voxels. At a larger distance from the source origin, a cone may contain several voxels, but the voxel intersecting the cone axis receives all of the deposited energy. This leads to reduced accuracy at extended distances.

A polyenergic spectrum can be accounted for by using a weighted sum of monoenergic kernels, the weights of which are derived by comparison with measured data. Advanced CCC implementations can include further improvements such as heterogeneity corrections through kernel scaling, modeling of lateral energy transport, beam

[3]Charged particle equilibrium is said to exist in a volume V in an irradiated medium if each charged particle of a given type and energy leaving V is replaced by an identical particle of the same energy entering V.

hardening and off-axis spectrum softening, and incorporating tilt of kernels. While CCC algorithms are able to achieve very good accuracy in homogeneous and many heterogeneous phantoms, heterogeneous cases with lateral charged particle disequilibrium in low density materials (e.g. lung treatments) are still not able to reconstruct dose with the same level of accuracy as MC simulations (Hasenbalg *et al.*, 2007 [36]).

2.1.3 Acuros® XB for the Eclipse TPS

Acuros® XB is an algorithm that solves the linear Boltzmann transport equation (LBTE) and accounts for the effects of heterogeneities in patient dose calculation (Varian Medical Systems [21]). Acuros uses the multiple-source model originally derived for AAA, and is integrated into the Eclipse TPS. The Boltzmann transport equation describes the behavior of radiation particles as they interact with matter. The LBTE is the linearized form of the BTE, and assumes that the radiation particles do not interact with each other or with external magnetic fields. The LBTE can be solved analytically only for very simplified problems, so modern algorithms must solve it non-analytically. Acuros uses numerical methods to explicitly solve the LBTE, and errors in the result are primarily systematic, resulting from discretization in energy, space and angle. Explicit solutions to the LBTE were developed to provide a high speed alternative to MC dose calculations. The primary efficiency to accuracy trade-off is in the choice of discretization granularity. Unlike Monte Carlo simulations, Acuros has the advantage of no statistical noise in the results.

In Acuros, dose calculation begins by ray tracing the external source models into the voxelized patient phantom, calculating uncollided photon and fluence distributions. Next, the scattered photon and electron fluences are calculated by iteratively solving the LBTE. Finally, the dose in each voxel is found using an energy dependent fluence-to-dose response function based on the local energy dependent electron fluence in the voxel.

The time-dependent three-dimensional system of coupled Boltzmann transport equations are solved (dependent variables not shown) (Varian Medical Systems [21]):

$$\hat{\Omega} \cdot \vec{\Delta}\Psi^{\gamma} + \sigma_t^{\gamma}\Psi^{\gamma} = q^{\gamma\gamma} + q^{\gamma}, \tag{2.12}$$

$$\hat{\Omega} \cdot \vec{\Delta}\Psi^{e} + \sigma_t^{e}\Psi^{e} - \frac{\partial}{\partial E}(S_R\Psi^{e}) = q^{ee} + q^{\gamma e} + q^{e}, \tag{2.13}$$

where

Ψ^{γ} = Angular photon fluence (or fluence if not time integrated), $\Psi^{\gamma}(\vec{r}, E, \hat{\Omega})$, as a function of position, $\vec{r} = (x, y, z)$, energy, E, and direction, $\hat{\Omega} = (\mu, \eta, \zeta)$

Ψ^{e} = Angular electron fluence, $\Psi^{e}(\vec{r}, E, \hat{\Omega})$

$q^{\gamma\gamma}$ = Photon-to-photon scattering source, $q^{\gamma\gamma}(\vec{r}, E, \hat{\Omega})$, which is the photon source resulting from photon interactions

q^{ee} = Electron-to-electron scattering source, $q^{ee}(\vec{r}, E, \hat{\Omega})$, which is the electron source resulting from electron interactions

$q^{\gamma e}$ = Photon-to-electron scattering source, $q^{\gamma e}(\vec{r}, E, \hat{\Omega})$, which is the electron source resulting from photon interactions

q^{γ} = Extraneous photon source, $q^{\gamma}(E, \hat{\Omega})$, for point source p, at position $\vec{r}_p$. This source represents all photons coming from the machine source model.

q^{e} = Extraneous electron source, $q^{e}(E, \hat{\Omega})$, for point source p, at position $\vec{r}_p$. This source represents all electrons coming from the machine source model.

σ_t^{γ} = Macroscopic photon total cross section, $\sigma_t^{\gamma}(\vec{r}, E)$, units of cm^{-1}

σ_t^{e} = Macroscopic electron total cross section, $\sigma_t^{e}(\vec{r}, E)$, units of cm^{-1}

σ_t = Macroscopic total cross section, $\sigma_t(\vec{r}, E)$, units of cm^{-1}

S_R = Restricted collisional plus radiative stopping power, $S_R(\vec{r}, E)$.

A macroscopic cross section is the probability that a particular interaction will occur per unit path length. The streaming operators $\hat{\Omega} \cdot \vec{\Delta}\Psi^{\gamma}$ and $\hat{\Omega} \cdot \vec{\Delta}\Psi^{e}$ describe motion of photons and electrons, respectively. Collisions, or the removal of particles from transport is accounted for in the collision operators $\sigma_t^{\gamma}\Psi^{\gamma}$ and $\sigma_t^{e}\Psi^{e}$. The third term on the left hand side of Equation 2.13 (the Boltzmann Fokker-Planck transport equation for electron transport), is the continuous slowing down operator, which includes Coulomb soft electron collisions. The right hand sides of both equations 2.12 and 2.13 contain the source operators for external sources, production and scattering. In the interest of brevity, complete definitions have not been provided.

Once the electron angular fluence for all energy groups has been calculated by Acuros, the dose in voxel i of the virtual phantom is calculated by

$$D_i = \int_0^{\infty} dE \int_{r\pi} d\hat{\Omega} \frac{\sigma_{ED}^{e}(\vec{r}, E)}{\rho(\vec{r})} \Psi^{e}(\vec{r}, E, \hat{\Omega}), \tag{2.14}$$

where

σ_{ED}^{e} = Macroscopic electron energy deposition cross sections in units of MeV / cm

ρ = Material density in g / cm^3.

2.2 Monte Carlo dose calculation

The Monte Carlo (MC) approach to dose calculation stochastically models individual particle interactions to simulate transport processes and energy deposition. MC methods indirectly solve the LBTE, and are convergent on an exact solution. In practice, MC solutions to the LBTE are not exact, but contain random errors that result from simulating a finite number of particles and uncertainties in particle interaction data. The track that a primary particle and all of its secondaries take though the geometry is referred to as a history. Usually, the simulation of millions of histories is necessary in order to achieve a high precision dose result. In the following sections, an introduction to the Monte Carlo method is presented, followed by its application to radiotherapy dose calculation. Sections 2.2.1 and 2.2.2 introduce MC sampling algorithms. The sampling algorithms used in realistic MC simulation of radiotherapy may be more complex in practice.

2.2.1 Transformation of random number by integral inversion

The MC method achieves the simulation of physical systems by distributing (pseudo) random numbers according to probability density functions (PDFs). There are many well-established algorithms for generating uniformly distributed numbers, so the typical approach is to use a method that transforms these into the required distribution. A simple example of this is the integral inversion transformation technique. Consider the outcome r from sampling a uniform random variable R. It is possible to transform r to an outcome x from random variable X modeling the PDF $f(x)$ by inverting the cumulative of the PDF, the cumulative distribution function (CDF) $F_X(x)$. To do this, first note that if X is a continuous random variable with CDF $F_X(x)$, then $R = F_X(X) \sim \text{Uniform}(0, 1)$. The inverse transformation function is defined as

$$F_X^{-1}(r) = inf\{x : F_X(x) = r\}, \tag{2.15}$$

where $inf\{\}$ is the infimum. If $R \sim \text{Uniform}(0,1)$, then for all $x \in \mathbf{R}$,

$$\begin{aligned} P(F_X^{-1}(R) \leq x) &= P(inf\{t : F_X(t) = R\} \leq x) \\ &= P(R \leq F_X(x)) \\ &= F_R(F_X(x)) \\ &= F_X(x). \end{aligned}$$

Therefore, $F_X^{-1}(R)$ has the same distribution as X. To generate random samples from the PDF $f(x)$, first generate a Uniform$(0,1)$ variable r, then deliver the inverse $F_X^{-1}(r) = x$. The inversion technique can only be used when the cumulative of the PDF can be inverted analytically.

This method can be applied directly to MC simulation of photon transport. For example, it can be used to sample the distance x to the next interaction of a photon incident on an infinitely thick slab of homogeneous material. The PDF $f(x|\mu)$ is exponential and depends on the sum of linear attenuation coefficients μ (cm^{-1}) for all interaction types in the material at a given energy. A uniform random number r can be transformed into the random number x distributed according to $f(x|\mu)$ as follows. For the function

$$f(x|\mu) = \mu e^{-\mu x}, 0 \leq x \leq \infty, \tag{2.16}$$

the CDF is calculated as

$$r = \int_0^x \mu e^{-\mu x} dx = 1 - e^{-\mu x}. \tag{2.17}$$

Inverting the CDF,

$$x = -\frac{1}{\mu} \ln(1 - r). \tag{2.18}$$

2.2.2 Transformation of random numbers by acceptance-rejection

In contrast to the integral inversion transformation technique, the acceptance-rejection method of transforming random numbers does not require the inverse of the PDF $f(x)$ to be known. Instead, a sampling envelope is defined that encloses the area under

the PDF. Consider a pair of uniform random numbers r_1 and r_2, generated such that

$$f_{min}(x) \leq r_1 \leq f_{max}(x)$$
$$x_{min} \leq r_2 \leq x_{max},$$

form the coordinates of the point (r_1, r_2) within the sampling envelope. Generate N such pairs, and for each incidence where

$$r_2 \leq f(r_1), \tag{2.19}$$

the random value r_1 is accepted as a transformed value. Otherwise, the pair is rejected and discarded.

While effective at generating random numbers distributed as $f(x)$, there are situations where acceptance-rejection is inefficient. For some functions or choices of sampling envelopes, a large number of random numbers may be rejected to obtain a small number of accepted values. One technique to improve efficiency is by defining a non-rectangular sampling envelope.

In MC simulation of photon transport, the acceptance-rejection method can be applied to sampling of the Compton photon scattering angle θ. The PDF of scattering angles is described by the Klein-Nishina cross-section

$$\sigma(\theta) = \frac{\pi \sin(\theta) r_e^2 (1 + \cos^2(\theta) + \alpha^2 \frac{(1-\cos(\theta))^2}{1+\alpha(1-\cos(\theta))})}{\sigma_c (1 + \alpha(1 - \cos(\theta)))^2}, \tag{2.20}$$

where σ_c is the total cross section for Compton interaction, r_e is the classical electron radius, θ is the photon scattering angle and α is the fractional incident photon energy to electron rest mass. To sample from this PDF, generate two random numbers r_1, r_2 such that

$$0 \leq r_1 \leq \pi$$
$$0 \leq r_2 \leq \sigma_{max}(\theta),$$

and accept values of r_1 under the criteria that $r_2 \leq \sigma(r_1)$.

In modern MC codes, the sampling of Compton scattering angles follows more complex but efficient approaches.

2.2.3 Monte Carlo modeling of radiation transport

In Monte Carlo modeling of radiation transport, the simulated photon interaction types are incoherent (Compton) scattering with atomic electrons, pair production, photoelectric absorption and coherent (Rayleigh) scattering. Each process transfers energy from the photon to electrons in the medium, except for coherent scattering. The cross sections for each interaction type depend both on the medium through which the photon is moving, and the photon energy.

Given a model for the generation of source particles, photon transport begins by determining the path length to the next interaction (accounting for heterogeneities). Then the interaction type in sampled, making use of the interaction cross sections for the current photon energy and material type at the location of interaction. The interaction is then simulated, potentially resulting in a change of direction and energy of the photon, photon absorption and generation of secondaries.

Compared to electrons, photons undergo relatively few interactions. Electrons scatter elastically a large number of times through collisions with atomic nuclei, particularly at low energies. Electrons lose energy through inelastic collisions with atomic electrons and radiations (bremsstrahlung and annihilation events). Explicitly simulating the frequent direction changes is very computationally intense, leading most Monte Carlo software for macroscopic dosimetry to combine multiple scattering interactions into larger steps (the condensed history technique) to reduce simulation times. The challenge of this technique is in accounting for heterogeneity boundaries. As a solution, single scattering events may be modeled near boundaries (Kawrakow and Bielajew, 1998 [42]). For further speed-ups in dose calculation, energy cut-offs for both photons (PCUT) and electrons (ECUT) allow for the immediate absorption of particles with low energies, likely insufficient to result in transport to a new voxel (this is called range rejection). Larger energy cut-offs increase simulation speeds, but reduce accuracy. Simulation efficiency is defined in section 2.2.6.

2.2.4 Simulation of the accelerator head

A critical part of MC simulation of radiation therapy is the linac treatment head model. The modeled components usually include everything downstream of the electron beam just before it strikes the target or scattering foils. While accelerator engineers use sophisticated MC software to model the wave guide, focusing magnets, bending magnet, etc., for the purposes of radiotherapy simulations usually only the

final electron beam is modeled. This requires a number of assumptions to be made regarding its composition, such as the spot size and beam divergence. After the electron beam has been parameterized, subsequent downstream modeling requires a detailed physical model of the geometry. Specifications are sometimes provided by the accelerator manufacturer, but may not always be available. The most commonly used MC code for linac head simulation is BEAMnrc (Rogers *et al.*, 1995 [69]).

Simulation of the accelerator head can be split into two parts: (1) the upper portion of the model containing plan-independent components such as the target, primary collimator and flattening filter; (2) the lower potion with plan-dependent components such as the secondary collimators and MLCs. The upper portion of the model usually begins with a circular electron beam incident on the target (i.e. electron acceleration is not modeled). Since the upper components of the linac head do not change, redundant simulations can be avoided by storing the resulting particle distribution in a file, called a phase-space file. The plan-independent phase-space representing this stage of the simulation will henceforth be referred to as *PhspA*. To generate the PhspA, a simulation of the upper head is performed and particle data are recorded as they pass through the phase-space surface (usually a plane perpendicular to the beam axis just above the secondary collimators). For each particle the position, direction cosines, energy, particle type (photon, electron or positron) and weight are recorded. The weight of a particle corresponds to its relative importance in the particle distribution, and is propagated to all child particles. Subsequent energy depositions are multiplied by the particle's weight. To accurately model the particle distribution, it is necessary to generate a phase-space with relatively high particle density[4].

There are two different file formats that are used for phase-spaces in radiotherapy MC. The older, simpler format is Electron Gamma Shower (EGS), as defined by EGS software such as BEAMnrc. In an EGS phase-space, all particles reside on a plane, and each particle is specified by its position (x, y), direction cosines (u, v, w for the x- y- and z-directions, respectively), energy E, weight wt, and LATCH. A particles weight is used to signify the relative importance of particles, and all dose depositions from the particle and its secondaries are multiplied by wt. The LATCH is an inheritable record of a particles history encoded in a 32-bit variable. In contrast,

[4]In general, it is not trivial to specify how high a particle density is sufficient - different applications have vastly different requirements. Additionally, some variance reduction techniques (section 2.2.6) can reduce density requirements.

the newer International Atomic Energy Agency (IAEA) phase-space format uses a separate header file alongside the phase-space that specifies what data is provided for each particle in the phase-space. The IAEA format is less rigid than EGS, enabling the phase-space to contain particles in all three dimensions (instead of a plane) as well as additional (or fewer) parameters per particle.

To simulate the plan-dependent components of the linac, particles from the upper head model (or from the PhspA) are transported through the beam shaping apertures. The particles still in motion past the last component may again be stored in a phase-space, called *PhspB*. This is the final particle distribution that is output from the linac head.

The method of modeling the linac by geometrical definition and MC simulation described above results in an accurate characterization of the beam, but is also computationally expensive. Additionally, it requires detailed specifications of the linac components and their composition, which can only be provided by manufacturers and is not generally available. For this reason, numerous alternative source models have been presented in the literature, as discussed below. A good characterization of a linac head for photon radiotherapy should consider the appropriate energy spectra, angular and spatial distributions, and particle fluences resulting from a particular treatment head model. Particle sources have been previously reported to model fluences through linac treatment head components that have invariable geometries (Ma 1998 [54], Wittenau *et al.* 1999 [93], Deng *et al.* 2000 [19], Fix *et al.* 2004 [24]) or variable geometries (Fippel *et al.* 2003 [23]). Source characterization parameters are derived from simulation, measurement or combination of the two. Usually this characterization is broken into several sub-source components representing the photon and electron contributions from different geometrical structures in the linac (primary collimator, flattening filter, etc.). While these models tend to be less accurate than phase-space sources derived from full MC simulation, they may be more efficient, avoid phase-space latent variance (Sempau *et al.* 2000 [75]), and have tuneable parameters to match machine output. Phase-space derived source models that use histograms to characterize the sub-sources have also been developed by a number of groups (Ma 1998 [54], Wittenau *et al.* 1999 [93], Deng *et al.* 2000 [19], Fix *et al.* 2004 [24]). An analytical model was developed by Fippel *et al.* (2003 [23]), which combined two Gaussian photon sources with a uniform electron source. In this case it was necessary to account for fluence variations using numerous free parameters, some of which were fit with measurement in air.

2.2.5 Simulation of radiation transport in the patient

The software which performs MC simulations of radiotherapy dose deposition in a phantom is often called a *dose calculation engine*. These engines model a patient using 3D material density maps of patient geometry obtained from a computed tomography (CT) scan. Complex heterogeneous anatomy is thus approximated by polygonal (usually cubic) voxels. Each voxel is homogeneous and has a material and density assigned. This 3D matrix is referred to as a *phantom*, that models a patient or arbitrary geometry. Since CT images are very high resolution, they are usually down-sampled in the process of building a patient phantom to a coarser resolution. This improves the efficiency of MC simulations, though voxellized spatial discretization can lead to partial volume averaging artefacts near high density gradients (this can cause the dose calculation engine to select an unrealistic material for MC simulation). The efficiency of MC simulations also depends on the phantom resolution, since crossing boundaries between voxels tends to require additional computational steps to account for possible heterogeneities. This dependency varies greatly between different transport algorithms.

The particle distribution output from linac head simulation is used as input for the dose calculation engine. The incident distribution of particles can then be transported through air to the surface of the phantom. Since few photon interactions occur in air, some dose engines simply project particles directly to the phantom surface without simulating transport in air. Then, transport begins in the patient geometry. As the simulation proceeds, particles deposit energy in the voxels of the phantom. The energy depositions are cumulated throughout the simulation and can be later converted to dose in units of Gy per initial electron, and calibrated to match measured machine output. The final result of the computations is the dose distribution, a 3D voxelized dose map that can be overlaid on the phantom.

Absolute dose conversion

The simplest method to calibrate output from a dose engine with measurement is to simply determine the single calibration factor that matches the dose under calibration conditions (e.g. a point in a 10×10 cm^2 field and 90 cm source-to-surface distance (SSD) in water). The MC dose, D^{MC}, at 10 cm depth along the beam axis is used as the calibration point. Then the tissue maximum ratio (TMR) is applied to match measurement, $D^{measured}$, if it is at a different depth. If the dose calculation engine

does not already account for the MU of the beam, the MU is a further multiplicative factor.

$$D' = D_o \frac{TMR(10, 10 \times 10) D^{measured}(d^{max}, 10 \times 10)}{D^{MC}(10, 10 \times 10)} MU. \tag{2.21}$$

However, there is an experimental effect that this does not account for. Namely, the output of some linac models is affected by backscatter from the secondary collimators into the monitor chamber. For example, a smaller jaw setting results in a higher backscatter signal into the monitor chamber, causing the set number of monitor units (MUs) to be reached sooner. Verhaegen *et al.* (2000) [92] showed that this backscatter decreases approximately linearly with field size, and the relative output factor (ROF) of a 40×40 cm^2 field compared to a 10×10 cm^2 field is nearly 2%. When modeling the treatment head completely, it is possible to record the backscatter into the monitor chamber during the simulation and adjust the final dose accordingly (Popescu *et al.* 2005 [64]). This is done by recording the dose in the chamber separate for the forward (toward the phantom) or backward moving particles, $D_{ch}^{forward}$ and $D_{ch}^{back}(field)$ respectively. The term $D_{ch}^{forward}$ is constant, while $D_{ch}^{back}(field)$ depends on field size. The absolute dose, corrected for backscatter, is then calculated as

$$D = D' S_b, \tag{2.22}$$

where

$$S_b = \frac{D_{ch}^{forward} + D_{ch}^{back}(ref)}{D_{ch}^{forward} + D_{ch}^{back}(field)}, \tag{2.23}$$

and $D_{ch}^{back}(ref)$ is the dose from backwards moving particles for the reference field size.

When the linac geometry is not available, it is not possible to obtain $D_{ch}^{forward}$ and $D_{ch}^{back}(field)$ by simulation. In this case, measurements can be used to create a look-up table of the values of S_b for various field sizes, and to extract $D_{ch}^{forward}$ using equation 2.23 (Zavgorodni *et al.* 2014 [99]).

2.2.6 Variance reduction techniques

MC dose calculations are very computationally intensive, usually requiring powerful computing resources (or clusters of computers) to achieve reasonable simulation times and low statistical uncertainty. Consequently, significant work has been dedicated to improving the efficiency of simulations without substantially degrading the accuracy.

The statistical uncertainty is the limiting factor on the accuracy of a MC calculated mean $< f(N) >$ of a quantity f. It depends on the number of particles N that deposit energy to a given voxel in the phantom. The variance, the square of the standard deviation $\sigma(N)$, is a measure of the statistical fluctuations, and tends to zero as $N \to \infty$. However, since the true value of f is generally unknown, $\sigma(N)$ cannot be calculated. Instead, estimated standard deviation $s(N)$ in a voxel can be calculated during a MC simulation as

$$s(N) = \sqrt{\frac{< f^2(N) > - < f(N) >^2}{N-1}}. \tag{2.24}$$

This is calculated most accurately when it is performed history-by-history during MC simulation, averaged over all histories (Salvat *et al.*, 1996 [71], Sempau and Bielajew, 2000 [74], Kawrakow, 2001 [41], Salvat *et al.*, 2011 [70]). The estimated variance can also be calculated by analyzing the results of several statistically independent simulations of the same scenario. For the purposes of this dissertation, the efficiency is therefore defined as

$$\epsilon = \frac{1}{s(N)^2 T(N)}. \tag{2.25}$$

Techniques that do not degrade the accuracy are referred to as variance reduction techniques (VRTs)[5]. These techniques allow for the reduction of variance in the dose distribution for a given amount of calculation time in order to increase the simulation efficiency. Many VRT strategies use a combination of the methods that will be described in the following sections: particle splitting and Russian Roulette. Based on these ideas, complex VRT algorithms can be developed, such as uniform, selective, and directional bremsstrahlung splitting (Kawrakow *et al.*, 2004 [45], Rodriguez *et al.*, 2012 [68]). Other efficiency enhancement techniques include range rejection, Woodcock tracking, correlation sampling, initial calculation of the primary interaction density, quasi-random sequences, macro MC, history repetition, simultaneous transport of particle sets (STOPS), kerma approximation, and transport parameter optimization (Kawrakow and Fippel 2000 [43]).

[5]Following convention in the literature, I will not make the distinction between approximate VRTs and real VRTs. Even if the final result of a simulation is not affected in a significant way, approximate methods are not VRTs. However, approximate efficiency enhancement methods form the basis of the majority of MC simulation in RT and are commonly referred to as VRTs (for example, the condensed history technique and the continuous slowing down approximation). Pragmatically, the methods presented in this dissertation could be generally referred to as efficiency enhancement techniques.

Particle splitting/recycling

Particle splitting, which may also be called recycling, involves duplicating a particle N^{split} times, and reducing the weight of the sub-particles by $1/N^{split}$ to maintain the energy fluence. Since subsequent transport processes are random and independent for the split sub-particles, the number of interactions simulated in the region will increase. This technique is usually used with specific spatial regions of interest in mind. For example, particle splitting at the surface of the phantom enhances the interaction density within the phantom, decreasing the variance without requiring additional simulations through the linac head.

The benefit of particle splitting is limited by the number of particles in a phase-space source and inherent statistical uncertainty. This is referred to as latent variance (Sempau *et al.* 2000 [75], Ezzati and Sohrabpour, 2013 [20]). For example, consider the case where only a few independent particles exist in the initial phase-space and excessive splitting is employed. The splitting helps to increase the particle density, but their fluence, energy and angular distribution will not represent reality, introducing error to the dose calculation.

Russian roulette

Russian Roulette is commonly used in conjunction with particle splitting. This technique reduces the number of particles that need to be simulated, and is particularly useful in reducing the number of "unimportant" particles that are transported (for example, those directed away from the phantom). Russian Roulette terminates some particles by comparing the probability for survival $p_{survive}$ with a random number ξ: the particle is terminated if $\xi > p_{survive}$. Surviving particles have their weight increased by the factor $w = 1/p_{survive}$ to compensate for the decreased energy fluence. The calculation time dedicated to unimportant particles is therefore reduced without completely eliminating the contributions.

2.2.7 Hardware for Monte Carlo simulations

Typically, MC simulations in RT have been processed using CPU-based computing resources. For example, in order to calculate a typical clinical case to reasonable accuracy, simulations can require hours of processing time on a 64-core computer. For this reason, it is common to combine several multi-core computers to process

simulations in parallel, combining the results cumulatively after all calculations have completed. The initial cost and maintenance of these systems can be substantial, presenting a deterrent for institutions with limited resources.

One alternative is to use externally hosted commercial computing resources. To do this, one would need to remove patient-specific identifiers from all data, rendering it anonymous, prior to exporting to an external server. In this case, there will be some sacrifice to calculation speed due to the data transfer to and from a CPU resource would likely be small compared to the total simulation times. Additionally, externally hosted computing resources may be susceptible to unexpected and uncontrollable outages, and the costs per CPU-hour of calculation tend to be higher. The advantages of this strategy is that hardware maintenance is not the responsibility of the institution and there are lower initial investment costs.

Modern graphics processing units (GPUs) offer a compelling alternative to powerful CPU resources. A small number of GPUs provide a high performance computing platform suitable for some RT applications for a substantially lesser hardware cost than an equivalent CPU cluster. Due to the reduced size of GPU clusters, they also require less maintenance, space, cooling and power. This is particularly beneficial to clinical situations where the computing resources are preferred to be located on-site. As with external CPU resources, it is possible to use external GPU resources to perform simulations.

2.2.8 GPU-based Monte Carlo

Clinical implementation of MC algorithms in commercial systems to date has been limited due to long computation times and substantial hardware requirements. It is only recently that some commercial MC solutions have been offered, such as CMS Monaco (Grofsmid *et al.*, 2010 [35]) and Accuray Multiplan (Sharma *et al.*, 2010 [76]). One avenue with promise is the utilization of GPUs for dose calculation. Specialized software for GPU-based dose calculation has been developed by several groups and demonstrated to have significant speed benefits over classical (CPU-based) codes (Badal and Badano 2009 [3], Hissoiny *et al.* 2011 [37], Jia *et al.* 2011 [39], Pratx and Xing 2011 [65], Jahnke *et al.* 2012 [38]).

The efficiency enhancement arises from the specialized hardware design and programming model of GPUs. In particular, GPUs contain a large number of processing units called stream processors, which are physically grouped on the device into multi-

processors. The clock speeds of these processors are typically lower than consumer grade CPUs, but the higher processing bandwidth leads to faster overall calculation speeds, particularly for highly parallelizable algorithms. There is also a very specific memory structure on the GPU that directly affects the design of GPU-compatible software. Analogous to RAM on CPU systems, there is global memory on the GPU (up to several gigabytes) that can be accessed by all processors. This memory space is accessible from both the CPU host and GPU device, unlike the following memory types, but it also has the lowest bandwidth. Each multi-processor has access to an independent memory space, called shared memory, that allows for memory to be shared between processors in the multi-processor. Due to its physical location directly on the multi-processor, this memory can be accessed quickly but has limited size. Finally, each stream processor has memory spaces called registers - this is where the local variables in a kernel are stored. A kernel is a special type of function that is launched in multiple copies on the GPU, termed threads. When kernels require more memory than is available in the registers, the overflow is allocated in slower local memory for the multi-processor. The threads are grouped into "blocks", which are distributed to the multi-processors for execution (figure 2.1).

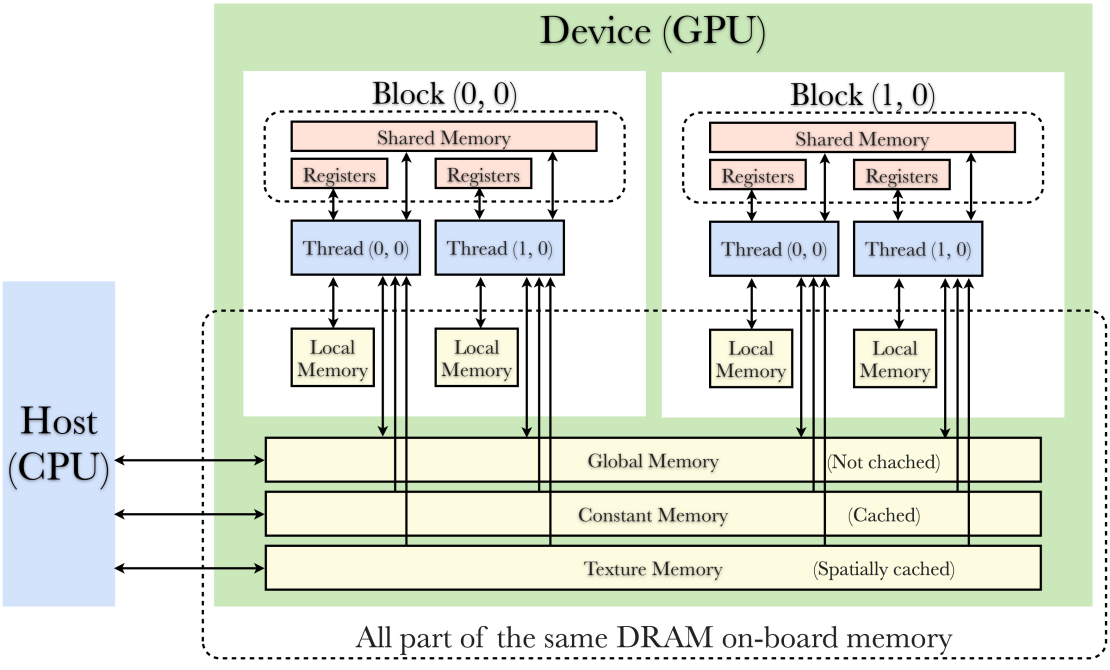

Figure 2.1: An illustration of memory access on a GPU device.

While modern CPUs are capable of hosting at most ~ 100 threads simultaneously, GPUs can host tens of thousands. Each multi-processor executes smaller groups of threads from a block in parallel, called warps (e.g. ~ 32 threads/warp). The execution model for these warps is essential to understand for efficient software design. While processing a warp, a single instruction is executed for all threads in the warp simultaneously, a strategy called single-instruction-multi-data (SIMD). This method is very efficient when the same instruction is issued on all threads, and the efficiency is reduced when a thread diverges into a different execution path due to conditional statements. When thread divergence occurs, instruction execution serializes, causing some of the processors to idle until convergence is reached.

The stochastic nature of MC simulations means that it is very difficult to avoid thread divergence completely, particularly when parallelization is distributed as one-particle-per-thread. However, it is possible to mitigate divergence through deliberate organization of data. For example, when particles in a warp are of a similar type and energy there is a higher probability that the same type of interactions will occur, leading to similar instruction paths and reduced thread divergence (Hissoiny *et al.* 2011 [37], Jia *et al.* 2011 [39]).

The GPU-based dose planning method (gDPM)

Much of the work presented in this dissertation will build on an existing dose calculation engine called GPU Dose Planning Method (gDPM), presented in Jia *et al.* 2010 [40], 2011 [39]. The physics of this code replicates that of a previously developed CPU-based code, Dose Planning Method (DPM), from Sempau *et al.* 2000 [75]. These codes use approximations valid for the a small range of energies (~ 100 keV to ~ 20 MeV) and materials practical for radiotherapy to simulate coupled photon-electron transport in a voxellized geometry. Due to clinical applications requiring only a few low-atomic-number materials, certain cross sections and distribution functions are determined by scaling them approximately to exactly computed data for water. The accuracies of both DPM and gDPM v2.0 have been validated to within 2% for both photon and electron clinical treatment beams.

Energy losses are treated using a class II 'mixed' scheme for energy losses (Berger 1963 [5]). That is, step-by-step simulation is used for inelastic scattering and bremsstrahlung emission above certain energy cut-offs, and the continuous slowing down approximation (CSDA) is used below a given energy threshold. A step-size independent multi-

ple scattering theory is used (Kawrakow and Bielajew 1998 [42]), based on the Lewis (1950 [50]) formulation of GoudsmitSaunderson theory (Goudsmit and Saunderson 1940 [30], [31]). These theories provide exact solutions for the angular distribution of an electron in motion over a given distance. The screened Rutherford cross section with Molière screening (Molière 1948 [59]) is used with the Bethe (1953 [7]) large angle correction. Photon transport uses the Woodcock tracking VRT, and electrons are transported using condensed histories.

A flowchart of the overall particle batching scheme of gDPM v2.0 is provided in figure 2.2. To summarize, source particles are obtained from the source model and placed into two stacks, one for photons and one for charged particles (electrons and positrons). A number of particles (up to the maximum that can be simulated simultaneously on the given GPU hardware) are then loaded from one of the stacks, and MC simulation proceeds. After simulation, the next set of particles (secondaries as well as new source particles) are loaded for simulation, and the process repeats. During simulation, secondary particles are added to the stacks. A number of statistically independent batches may be performed to allow for error estimates.

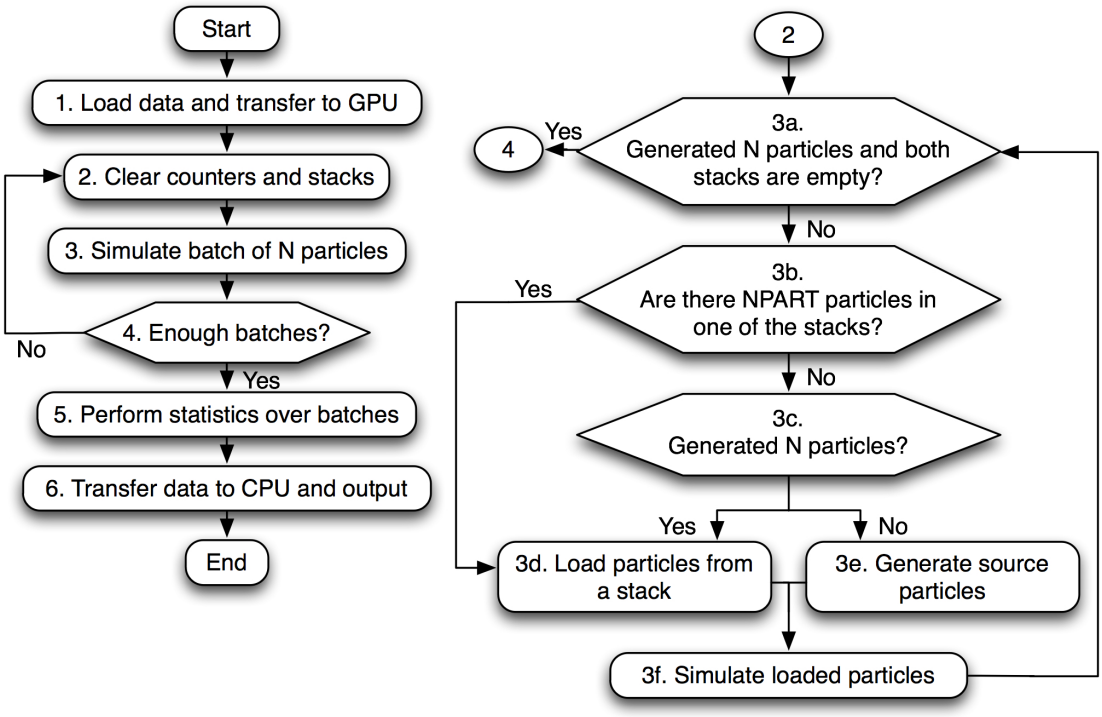

Figure 2.2: A flow chart illustrating how particles in gDPM v2.0 are stacked in GPU memory during source generation and transport.

Particle generation in gDPM v2.0 uses a point source with particle energies sampled from a spectrum. Each energy bin is simulated separately to avoid thread divergence. The software generates either pure photon or pure electron sources (but the transport itself may generate both types of particles). The directions of source particles are determined by sampling which fluence map element I a source particle will strike in a fluence map at the MLC plane. The fluence map, containing N_f elements in a 2D grid, represents MLC modulation and is derived directly from MLC motions using an algorithm that approximately accounts for rounded leaf ends and inter-leaf leakage. The procedure follows the Metropolis Monte Carlo algorithm (Jia *et al.*, 2011 [39]), sampling using the fluence map element values f_I.

For the Metropolis algorithm, begin by initializing I_{prev} with an arbitrary beamlet index in $\{1, 2, ..., N_f\}$.
Do the following steps each time a particle is generated:

1. Generate a trial beamlet $J \in \{1, 2, ..., N_f\}$ with equal probability;
2. Generate a random number r uniformly distributed in $[0, 1]$;
3. If $r < f_J/f_{I_{prev}}$, set $I = J$, otherwise set $I = I_{prev}$;
4. Generate a particle from the beamlet I, position sampled uniformly;
5. Set $I_{prev} = I$.

As illustrated above, the MLC model in gDPM accounts for modulation by generating fewer particles directed towards lower intensity regions of the fluence map. A complete geometrical model of the MLCs would provide higher accuracy results, but the fluence map method is faster to simulate.

Chapter 3

Phase-space collimation on the CPU

In radiotherapy MC simulations, modeling of the secondary collimators can be the most time consuming portion of the calculation. This is especially true when an approximate MLC model such as vcuDMLCcode (Keall *et al.* 2001 [47], Siebers *et al.* 2002 [77]) is used in combination with a fast dose calculation engine (e.g. VMC++), both of which are options in VIMC (sections 4.2 and A.1.1). To reduce the bottleneck of secondary collimator simulation, an algorithm based on assuming perfect absorption in the secondary collimators was developed, called the phase-space collimation (PhspC) method.

3.1 The PhspC algorithm

Phase-space collimation is an alternative to performing MC simulation of photon & electron transport through secondary collimators. Instead, the software uses the positions of the top surface (closest to the target) of the secondary collimators to remove photons & electrons from an input phase-space (this is a PhspA, a plan-independent phase-space). Using the original positions and directions of the particles in the phase-space, they are projected to the collimation plane (the top surface of each collimator) without accounting for scattering in air. The new particle coordinates are then compared with the secondary collimator positions on the plane, and only those particles that are within the collimator opening are written to the output phase-space. Neglecting transmission through and scatter from the secondary collimators

in this way is expected to result in underestimation of dose, particularly in out-of-field regions.

Uncertainty estimations are generally used to determine how many particles to simulate in the dose calculation engine, and this also determines how many particles are required in the phase-space output from the head model. For PhspC, the intermediary phase-space is generated to contain exactly the requested number, $N^{requested}$. The number of particles read from the input phase-space is N^{read}, and the number of particles contained in the input phase-space is N^{phsp}. The number of times to recycle each particle is set to a fixed number, $N^{recycle}$, which is chosen with the aim of being large enough to avoid needing to re-read the phase-space file multiple times for a typical simulation scenario, and small enough to avoid introducing latent variance artefacts in cases where entire phase-space is not used. azimuthal particle redistribution (APR) (Bush *et al.* 2007 [16], Brualla *et al.* 2010 [11]) is performed upon each recycling, and then the particles are ray-traced to the top of the secondary collimators to see if absorption occurs.

In APR, particles are rotated about the beam-axis each time they are recycled in order to reduce latent phase-space variance artefacts. A uniform random number is used to generate a new azimuthal coordinate in the interval $0 < \phi' < 2\pi$. The change in azimuthal coordinate $\alpha = \phi' - \phi$ must be accounted for in the x- and y-direction cosines of the particle, u and v respectively. The new direction cosines u' and v' are calculated as

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \tag{3.1}$$

The PhspC method for phase-space processing is presented in algorithm 1. Particles from the input phase-space are read, recycled, redistributed, collimated, and projected to the final phase-space plane before being written to the output phase-space. Both EGS and IAEA format input phase-spaces are supported, but all output phase-spaces are written in EGS format. Recall that w, E, and wt are a particles z-direction cosine, energy, and weight, respectively.

Algorithm 1 Phase-space collimation

```
1: procedure PhspCollimation
2:     Open EGS or IAEA format phase-space;
3:     Start from a random particle;
4:     N^processed = N^read = 0;
5:     while N^processed < N^requested do
6:         Read the next particle(x, y, z, u, v, w, E, wt);
7:         N^read = N^read + 1;
8:         if w < 0 or wt < 0 then
9:             Continue[1];
10:        end if
11:        for each integer j in N^recycle do
12:            Perform APR;
13:            Ray-trace particle to jaws;
14:            if particle is absorbed in jaws then
15:                Continue;
16:            end if
17:            Ray-trace particle to final output plane;
18:            Write particle to output phase-space;
19:            N^processed = N^processed + 1;
20:        end for
21:        if End of phase-space reached then
22:            Restart from top of phase-space;
23:        end if
24:    end while
25: end procedure
```

3.2 Methods used to compare 3D dose distributions

The three dimensional nature of dose distributions produced from MC simulations of radiotherapy present a challenge when it comes to validating new techniques and algorithms. It is possible for certain approximations, or even software bugs, to result in a subtle bias in the dose distribution. Such a bias may only be detectable using a specific type of analysis or under particular simulation conditions (beam energies, field sizes, phantom heterogeneities, etc.). For a thorough validation, it is therefore necessary to look at a range of cases and employ a variety of complementary agreement-tests.

Since relative uncertainty in MC simulations is higher in regions of low dose, and

areas of higher dose tend to be of greater interest, it is typical to perform analysis only using voxels where the dose in the reference is within some range. For example, where the dose D is greater than 20% of the maximum, D_{max}. The three techniques below can all be combined with this type of dose range selection technique.

3.2.1 Root mean square deviation

The simplest and most intuitive method for 3D validation of a dose distribution is by considering the percent differences between 1-to-1 voxels. In this dissertation, the dose grids under consideration will always be 1-to-1 and not require interpolations. The root mean square deviation (RMSD) is calculated as

$$RMSD = \sqrt{\frac{1}{N}\sum_{i}^{N}\delta_i^2}, \tag{3.2}$$

where N is the number of voxels, and δ_i is the percent difference between the analysis and reference cases in the voxel i.

3.2.2 Gamma-index test

The Gamma-index (γ-index) test (Low *et al.*, 1998 [53]) allows for the comparison of two dose distributions, and is generally used when both spatial and absolute dose discrepancies are expected. Two criteria are used to determine success or failure of the test for a given voxel: the dose difference criterion ΔD (the maximum % difference), and the distance-to-agreement (DTA) criterion Δr (the maximum distance within which ΔD must be achieved). Consider the reference and analysis dose distributions, D_r and D_a respectively. The γ-index at a point $(r_r, D_r(r_r))$ is

$$\gamma(r_r) = min_{r_a}\{\sqrt{\frac{|r_a - r_r|^2}{\Delta r^2} + \frac{|D_a(r_a) - D_r(r_r)|^2}{\Delta D^2}}\}. \tag{3.3}$$

If $\gamma(r_r) \leq 1$, then the dose at the point r_r passes the test. In the γ-index test algorithm used for this dissertation, all voxels within a distance of 5 mm from the edge of the current voxel along each dimension are included in the analysis. This ensures the test is sensitive to changes in Δr even when Δr is smaller than the sidelength of a voxel.

When comparing two MC dose distributions, it is important to recognize that statistical fluctuations can result in biased γ-index test results. Namely, the passing rate

is overestimated due to the statistical noise in the analysis dose and underestimated due to statistical noise in the reference dose (Graves *et al.* 2013 [32]). Additionally, when two dose distributions are perfectly aligned spatially (as may be the case when comparing two MC simulations), the DTA criterion may result in overestimation of agreement.

3.2.3 Chi-index test

The Chi-index (χ-index) test (Bakai *et al.*, 2003 [4]) is essentially a gradient weighted dose difference technique. For the dose difference criterion ΔD and DTA Δr, the χ value at $(r_r, D_r(r_r))$ is

$$\chi(r_r) = \frac{D_a(r_a) - D_r(r_r)}{\Delta D\sqrt{(\Delta r/\Delta D)^2(\nabla D_r(r_r) \cdot \nabla D_r(r_r)) + 1}}. \tag{3.4}$$

The χ-index test is passed when $-1 \geq \chi(r_r) \leq 1$. The χ-index test is a suitable alternative to the γ-index test when the two dose distributions are perfectly aligned (which eliminates the need for the a distance-to-agreement allowance). Instead, the χ-test includes an adjustment in the agreement based on the dose gradient.

3.3 CPU-based hardware

CPU-based computations is this dissertation utilized one or more nodes in the *cavivake* cluster, which was purchased using the British Columbia Cancer Foundation Innovation Support Fund. This cluster is a shared British Columbia Cancer Agency (BCCA) resource between Vancouer Island Centre (VIC), Vancouver Cancer Centre (VCC) and Centre for the Southern Interior (CSI), and was named by combining the first two letters of the locations Canada, Victoria, Vancouver and Kelowna. I designed and configured the system, under the supervision of Dr. Zavgorodni, and it is integrated into the Vancouver Island Monte Carlo (VIMC) framework. The cavivake cluster consists of three compute nodes, each with 4 AMD Opteron 2.1 GHz 16 core processors, 192 GB DDR3 RAM and 7200 RPM SATA hard drives. One of the nodes itself acts as the front-end submission host, and distributes jobs to the other nodes using the Condor batching system.

For any test cases where execution times were being monitored, it was ensured that the resource was not being shared with other users.

3.4 Benchmarking PhspC against BEAMnrc

Open fields in water were calculated to compare the accuracy of the PhspC method with full MC simulation of the secondary collimators using BEAMnrc. A virtual water phantom was used, positioned at a 90 cm SSD and comprised of $82 \times 82 \times 82$ voxels with 5 mm voxel resolution. The open field sizes 4×4, 10×10 and 30×30 cm^2 were simulated using the VMC++ code to transport 5.2 million, 32 million and 292 million particles in the phantom, respectively. The Varian TrueBeam 6MV model was used, with a phase-space provided by the linac manufacturer. The percentage differences shown are relative to the BEAMnrc benchmark. The percentage difference for a profile in voxel x was calculated as

$$\%Diff(x) = \frac{D_{PhspC}(x) - D_{BEAMnrc}(x)}{D^{max}_{BEAMnrc}}, \tag{3.5}$$

where D_{PhspC} is the dose from the PhspC method, $D_{BEAMnrc}$ is the dose from the BEAMnrc method, and $D^{max}_{BEAMnrc}$ is the maximum dose from the BEAMnrc method in the profile.

BEAMnrc was used with components modeling the monitor chamber, the MCTWIST[2] module for APR (Bush *et al.* 2007 [16]), and secondary collimators. Automatic recycling was used in BEAMnrc, which means that the number of recyclings was calculated as $N^{requested}/N^{phsp}$, rounded up. When using the PhspC method, the number of recyclings was set to 20.

In the VIMC framework, two dose engines are available for selection - DOSXYZnrc and VMC++. Using the simulation parameters presented in appendix A, VMC++ provides more efficient dose calculation using phase-space sources. For this reason, the VMC++ code was chosen for use with the PhspC method. For most clinical purposes, where the PhspC method is expected to be used, VMC++ has been shown to provide very good accuracy (Terribilini *et al.* 2010, [81]).

Cross-beam profiles and depth dose curves with SSD=90 cm are shown in figures 3.1 and 3.2, respectively, in units of Gy / e^- (the default output units for VMC++). Very good agreement is observed for the 4×4 and 10×10 cm^2 curves, but the largest field size has discrepancy near the beam-axis. Despite this, the overall agreement was good (table 3.1). The same open fields were repeated with SSD=80 cm, and similar

[2]The MCTWIST component module for BEAMnrc performs APR on each particle, resulting in variance reduction when combined with recycling.

artefacts near the centre of the field were observed for the 10×10 and 30×30 cm^2 field sizes (figures 3.3 and 3.4). For SSD=100 cm, the artefacts were not observed (figures 3.5 and 3.6).

To investigate latent variance effects as a possible cause of the discrepancy near the beam-axis, open field profiles were recalculated with recycling turned off. In the cases tested, the N^{read} was less than N^{phsp} even when recycling was turned off, so this was effective[3]. The profile curves for both $N^{recycle} = 0$ and $N^{recycle} = 20$ are shown in figure 3.7. The two recycling modes had little effect on the accuracy of the results, indicating that latent variance was not the problem. Rather, the effect must originate from differences in secondary collimator model. One hypothesis, is that the lack of simulation of scattering of photons and electrons in air during secondary collimator simulation lead to increased fluence directed along the beam-axis. Testing this hypothesis is left for future work.

Out-of-field regions are highlighted in figure 3.8. As expected, the dose out-of-field for PhspC is systematically underestimated compared to BEAMnrc, due to the lack of transmission and scatter modeling in PhspC. Therefore, simulations where the out-of-field dose is very important should not be calculated using PhspC.

Simulation speed of the secondary collimators increased in PhspC compared to BEAMnrc by a factor of 39, 53 and 100 for 4×4, 10×10 and 30×30 cm^2, respectively. Such considerable speed-ups motivate the use of PhspC, despite the accuracy trade-offs. Absolute calculation times for secondary collimator modeling and dose calculation are shown in figure 3.10.

Both the BEAMnrc benchmark and PhspC used the same calibration for absolute dose conversion (equation 2.21), but different methods for determining S_b values (equation 2.22). For BEAMnrc, the backscatter into the monitor chamber was determined by simulation and used in equation 2.23. For PhspC, backscatter is not modeled, so the S_b values were determined by look-up in a table (determined from BEAMnrc simulations for a variety of field sizes, as described in Zavgorodni *et al.* 2014 [99]).

Three realistic patient treatment plans were tested for the TrueBeam 6MV accelerator: an IMRT brain treatment using a homogeneous water cylinder, an IMRT oesphagus case using the patient phantom, and a VMAT lung case using the patient

[3]In cases where N^{read} is much larger than N^{phsp}, the phase-space must be "restarted" again from the first particle in order to achieve the requested number of output particles, effectively re-enabling recycling even if it was turned off.

phantom. Patient phantoms were adapted from CT scans into a format suitable for MC simulation, with the resolution downsampled. All of the virtual phantoms were created with $5 \times 5 \times 5$ mm^3 voxel size, and simulations achieved less than 1% statistical uncertainty. In χ- and γ-index tests, agreement was found to be good ($> 94\%$) for both 2% / 2 mm and 1% / 1 mm criteria above the 10% isodose in the BEAMnrc reference. Dose profiles were also produced for the IMRT oesphagus case, plotted separately for each beam in figure 3.9. The profiles were taken laterally at 10 cm depth in the patient phantom. The simulation times for all three cases are also shown in figure 3.10. As with the open field cases, there was a significant reduction in secondary collimator simulation times when using the PhspC method.

Plan	**Gamma (%)**		**Chi (%)**		**RMSD (%)**
	2% 2 mm	**1% 1 mm**	**2% 2 mm**	**1% 1 mm**	
4×4 SSD=80 cm	99.9	94.9	100.0	100.0	0.6
4×4 SSD=90 cm	100.0	100.0	99.9	98.8	0.5
4×4 SSD=100 cm	99.9	94.2	100.0	98.9	0.6
10×10 SSD=80 cm	100.0	99.4	100.0	99.7	0.4
10×10 SSD=90 cm	100.0	100.0	98.9	98.5	0.4
10×10 SSD=100 cm	100.0	98.6	100.0	98.9	0.5
30×30 SSD=80 cm	99.9	98.5	100.0	98.9	0.4
30×30 SSD=90 cm	100.0	100.0	98.2	98.2	0.5
30×30 SSD=100 cm	100.0	97.8	100.0	97.7	0.5
IMRT Brain (cylinder)	99.6	95.9	99.4	96.7	0.6
IMRT Oesphagus	99.9	96.4	99.5	96.0	0.6
VMAT Lung	100.0	94.9	99.8	98.9	0.7

Table 3.1: Results comparing the PhspC and BEAMnrc head modeling methods. Comparisons were performed only in voxels containing $> 10\%$ of the dose in the BEAMnrc reference distribution. The RMSDs are also shown. The IMRT brain case was performed in a homogeneous water cylinder, while the other two cases used heterogeneous patient phantoms, all with $5 \times 5 \times 5$ mm^3 voxel size. Statistical uncertainty was $< 1\%$.

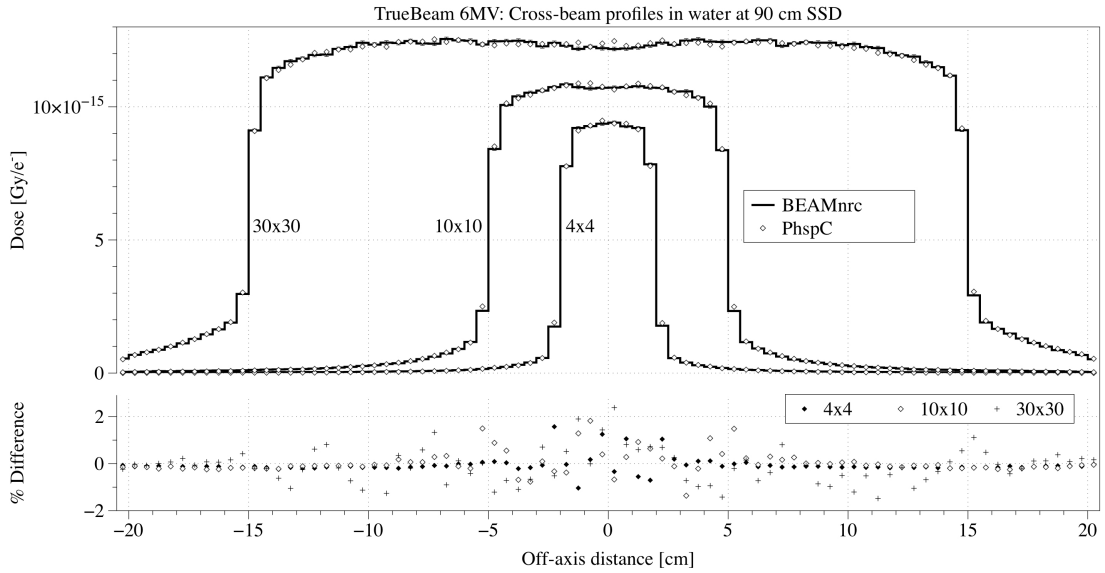

Figure 3.1: Cross-beam profiles for the TrueBeam 6MV accelerator at 10 cm depth and SSD=90 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

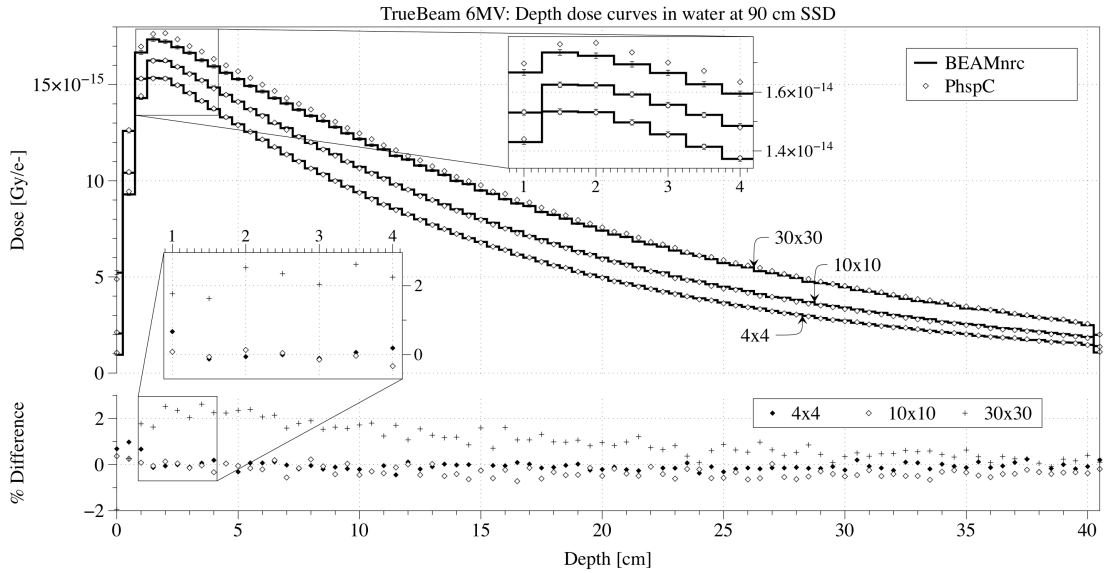

Figure 3.2: Depth dose curves for the TrueBeam 6MV accelerator at SSD=90 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

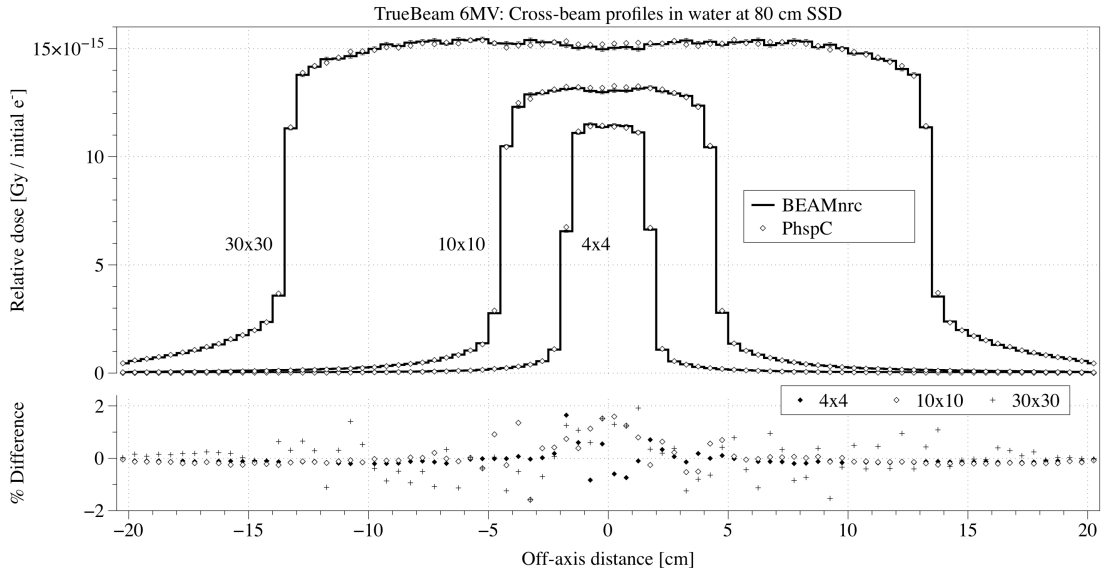

Figure 3.3: Cross-beam profiles for the TrueBeam 6MV accelerator at 10 cm depth and SSD=80 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

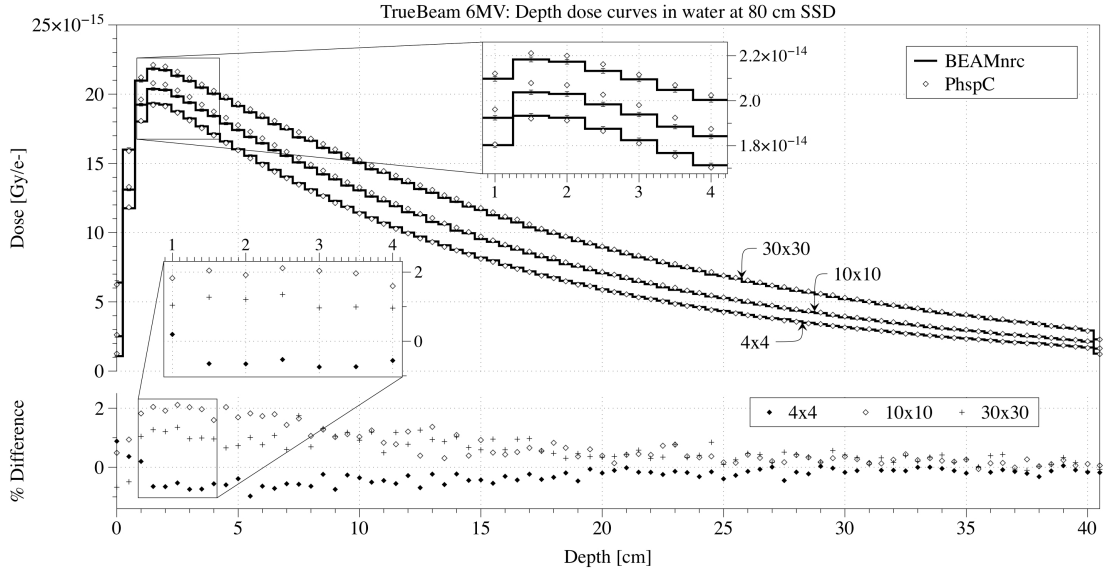

Figure 3.4: Depth dose curves for the TrueBeam 6MV accelerator at SSD=80 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

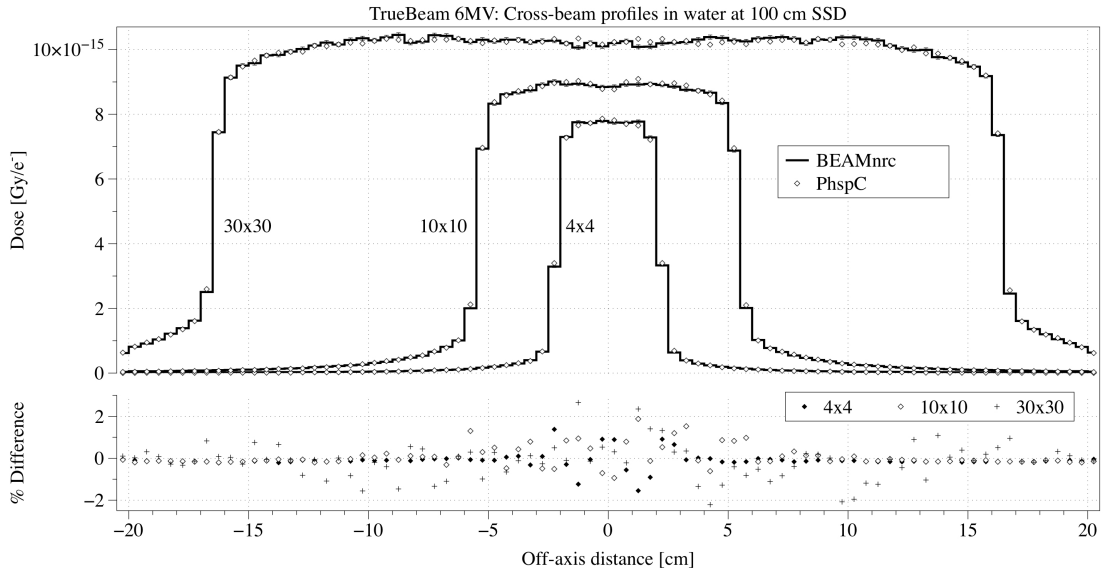

Figure 3.5: Cross-beam profiles for the TrueBeam 6MV accelerator at 10 cm depth and SSD=100 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

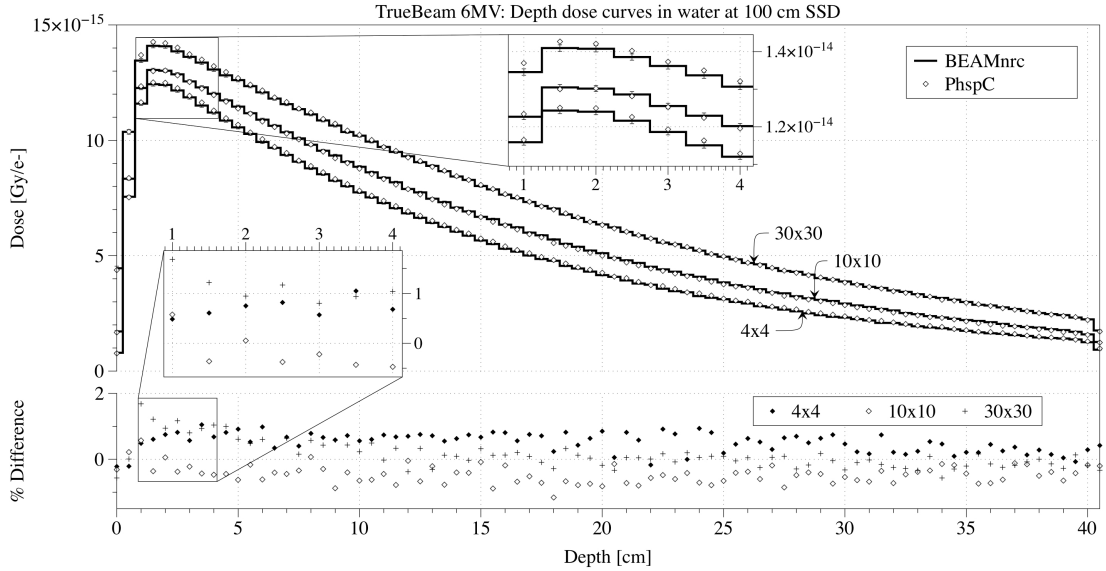

Figure 3.6: Depth dose curves for the TrueBeam 6MV accelerator at SSD=100 cm are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

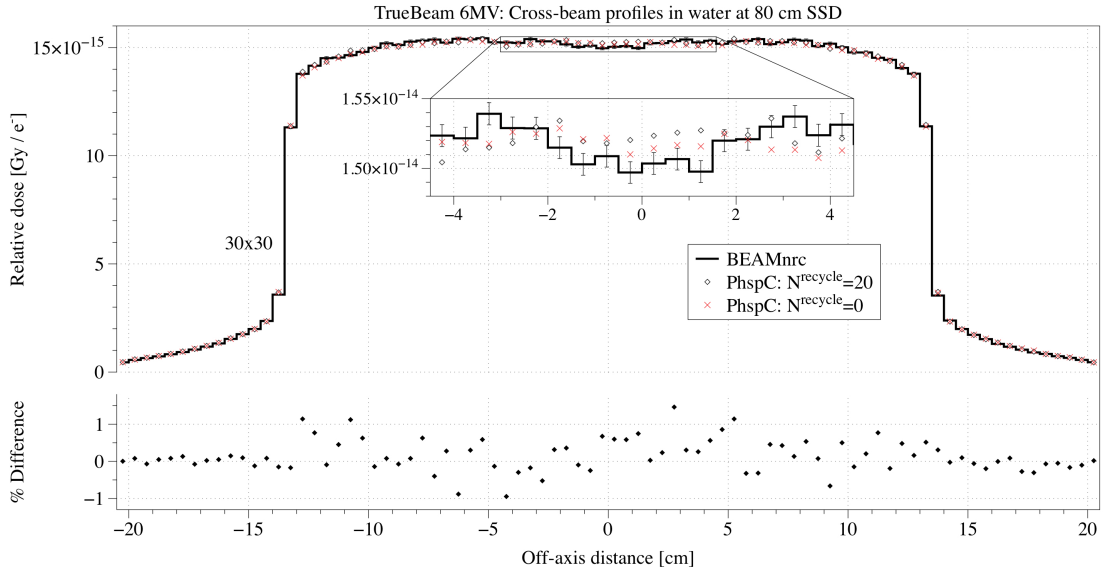

Figure 3.7: Cross-beam profiles for the TrueBeam 6MV accelerator at 10 cm depth and SSD=80 cm are shown in Gy / e^- for the PhspC method with $N^{recycle} = 0$ (red crosses), $N^{recycle} = 20$ (dots) and the benchmark, BEAMnrc (lines) all derived from the same initial phase-space. Uncertainties are shown only for the BEAMnrc curves. The percentage differences for the $N^{recycle} = 0$ case are also shown, relative to the maximum benchmark dose in the curve.

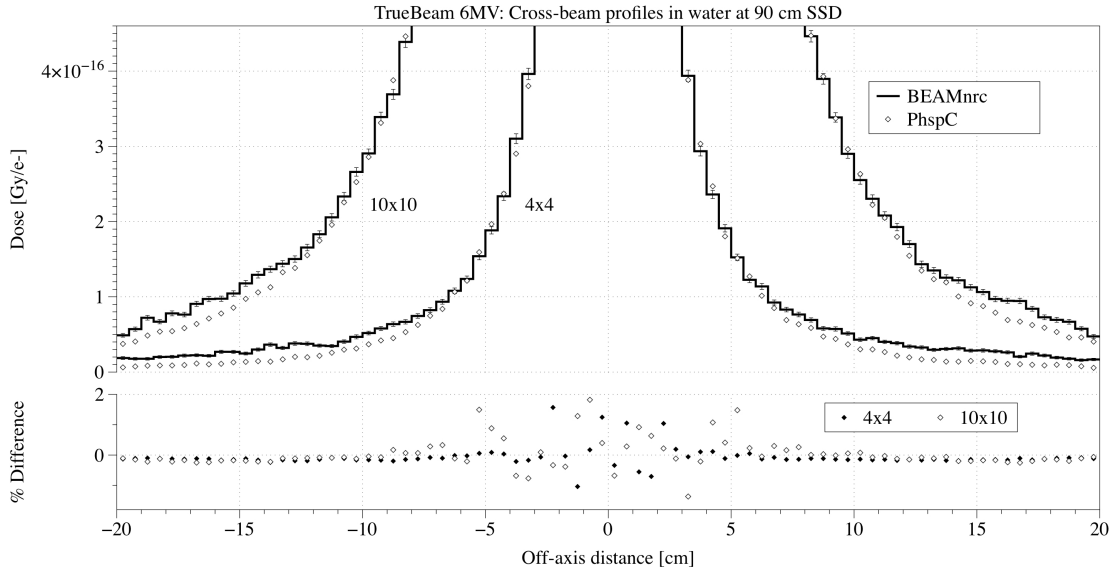

Figure 3.8: The same as figure 3.1, but zoomed in on the 4×4 cm^2 and 10×10 cm^2 field sizes to highlight out-of-field discrepancies.

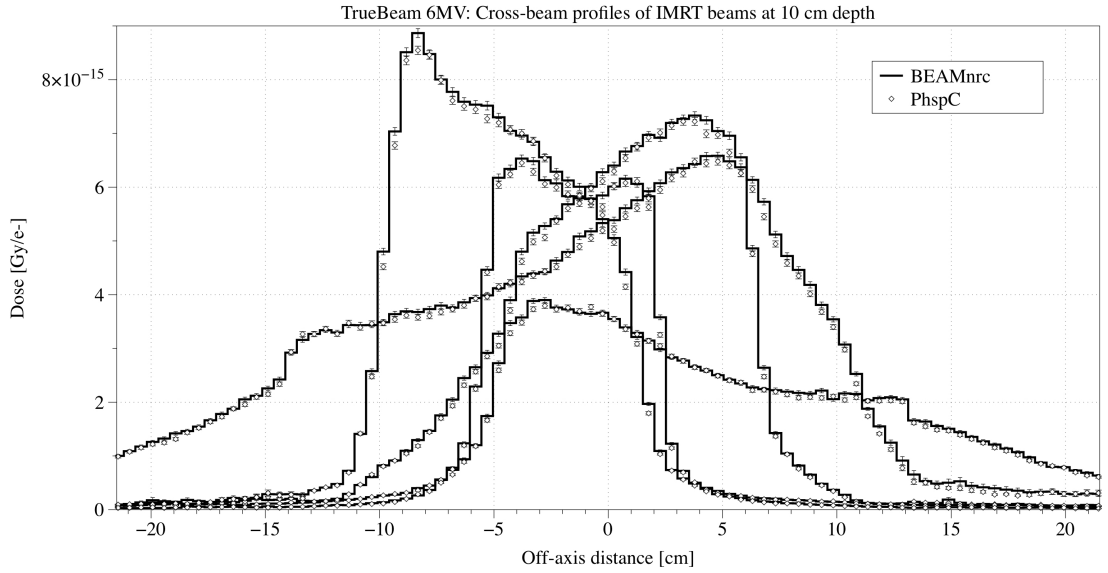

Figure 3.9: For the TrueBeam 6MV accelerator: profiles at 10 cm depth in a realistic patient phantom are shown in Gy / e^- for the PhspC method (dots) and the benchmark, BEAMnrc (lines) all derived from the same initial phase-space. Profiles for each of the individual beams in the plan are shown. This case is labelled as IMRT oesphagus in table 3.1.

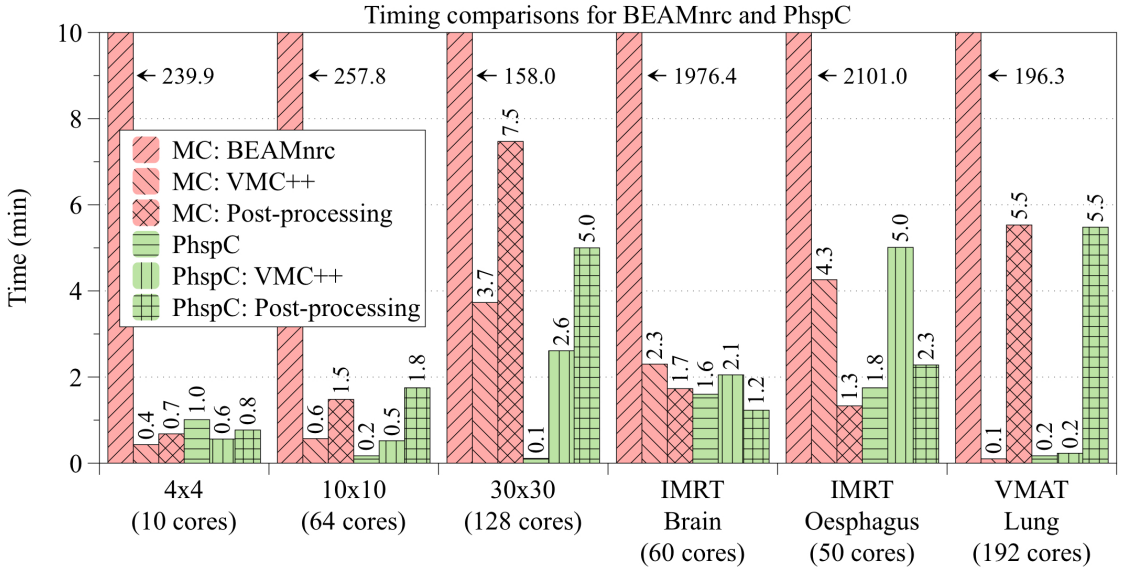

Figure 3.10: A breakdown of the simulation times for the PhspC method, compared to BEAMnrc. There are three components: modeling of the secondary collimator, dose calculation, and post processing (primarily dose summation). The first two components were determined by averaging the calculation time over all of the CPU-cores used for parallelization. The dose summation component occurs only on the last core to finish.

3.5 Discussion and conclusions

Schmidhalter *et al.* (2010) [73] tested absorbing secondary collimators similar to that used in PhspC. Two 100% absorbing secondary collimator models were used: (1) where the geometry was completely modeled (called the "Absorbing" model), and (2) where each collimator position was projected to a plane perpendicular to the beam-axis, located at the middle of the jaw (called the "Flat-Absorbing" model). Schmidhalter found the Absorbing and Flat-Absorbing simulation times to be faster than EGSnrc by 146× and 274×, respectively, for a 10×10 field using a 6MV Varian Clinac 2300C/D. While both methods found similar agreement as PhspC for 10×10 fields, these models were tested for IMRT cases using a simplified MLC model (also "Absorbing" or "Flat-Absorbing") and achieved poor agreement (a mean over 10 cases of 67.7% and 64.5% success, respectively, for 1% / 1 mm γ-index test).

In PhspC, the secondary collimator model is similar to the Flat-Absorbing, but the projected collimation plane was located at the top surface of the corresponding jaw instead of the middle. Additionally, the secondary collimator model is combined with the vcuDMLCcode for MLC simulation. In IMRT cases, the accuracy of MLC modeling is more important than the secondary collimator model, since the MLC usually blocks the edges of the field where the greatest dose differences from a Flat-Absorbing model are expected. Therefore, the improved γ-index test agreement results in PhspC compared to Schmidhalter for IMRT cases are attributed to the more accurate MLC model.

In future work, it may be worth investigating the difference in accuracy and efficiency when both the top and bottom surface of each secondary collimator is used to absorb particles, as in the "Absorbing" model. However, this would lead to reduced dose to the penumbra regions of the distribution, which is already underestimated by the current model. A better approach may be to include an approximate model of attenuation, instead of assuming 100% absorption. Particularly if applied only to particles near the edges of the collimators, this could lead to increased accuracy with minimal additional computations.

As shown in figure 3.10, the post processing component of the PhspC method is a bottleneck. The parallelization of computations over many CPU-core leads to increased time needed to sum the independent dose distributions. In the current process, the last core to finish dose calculations also performs summation of the results from all parallel cores. This is particularly apparent in VMAT plans, where

the simulation is divided into a large number of independent parallel simulations to discretely model the dynamic gantry rotation. In the future, a more efficient approach would be to parallelize the summation, adding together dose distributions as soon as they are complete instead of first waiting for all processes to complete.

Dynamic collimator motion (jaw tracking), where the secondary collimators move throughout treatment, can also be easily modeled using the PhspC method. This was done by integrating the PhspC method directly into the MLC model, vcuDMLCcode. At each MU checkpoint, or control point, used for discretized MLC motion, secondary collimator absorption was modeled using PhspC. The results of dynamic PhspC have not been included in this work, but the method is considered successful.

In conclusion, the PhspC technique provides a substantial reduction in simulation times at the sacrifice of a less accurate secondary collimator model. So long as the user is aware of the accuracy limitations, PhspC is a valuable addition to the clinical toolset. For example, PhspC is suitable for clinical MC dose calculations, dose second-check applications, and dose verification (by replacing the planned MLC motions with the delivered records). All of these capabilities have already been integrated into the Vancouver Island Monte Carlo (VIMC) web-based framework WebMC, as described in section A.1.2.

Chapter 4

Phase-space source models in GPU-based dose calculation

In this chapter, GPU-based dose calculation will be investigated in order to determine its feasibility for high speed radiotherapy dose calculations. The gDPM software (section 2.2.8) was adapted to use phase-space source models in order to achieve accuracy comparable to standard MC codes. However, it was found that the standard procedure of using phase-space files was very inefficient for GPU calculations. A novel method of pre-processing and utilizing the particles from phase-space files was necessary in order to attain reasonable simulation times.

The primary results of this chapter were published in Townson *et al.* (2013) [84]. Development of phase-space source models for gDPM v3.0 was supported in part by the University of California Lab Fees Research Program.

4.1 GPU-based source modeling: phase-space sources

Phase-space sources have the potential to provide the most accurate source characterization, and are accepted as inputs by many of the popular CPU-based MC dose calculation tools such as DOSXYZnrc (Walters *et al.* 2003 [94]), MCNP (Siebers *et al.* 1999 [78]) and some versions of VMC++ (Gardner *et al.* 2007 [28], Bush *et al.* 2008 [14]). However, prior to the work of this dissertation, phase-space source functionality had yet to be implemented in the published GPU-based MC dose engines. The vast differences in hardware architecture and simulation schemes mean that substantial and creative work is required to adopt phase-space sources in GPU-based

MC dose calculation without considerable loss of efficiency. If efficiency was not of concern, the use of a phase-space file in a GPU-based dose calculation engine would be quite straightforward. Each particle could be sequentially loaded from the file, and then transported through subsequent patient-specific beam modifiers (secondary collimators and multi-leaf collimators, or MLCs), and the patient body. However, to achieve a high efficiency in a GPU-based MC dose calculation, where a large number of threads transport particles simultaneously, it is important to avoid the thread divergence problem (section 2.2.7). Since particles are usually ordered in a phase-space file randomly, their sequential use will maximize thread divergence, resulting in low computational efficiency. It has been observed previously that this problem can be greatly reduced by separating photon and electron transport and grouping particles of similar energy (Hissoiny *et al.* 2011 [37], Jia *et al.* 2011 [39]).

There are further efficiency concerns when using a plan-independent phase-space file, which stores particles at locations above all plan-dependent components of a linac. The phase-space captures the particle fluence formed by the primary collimator and covers an area exceeding the largest possible field size, but the field sizes in a treatment plan tend to be much smaller. In these cases, a majority of the particles are absorbed by the secondary collimators, and relatively few reach the patient phantom. In traditional MC simulations, the overhead of transporting those particles that will not eventually contribute to the dose is generally considered acceptable in order to maintain the most accurate possible calculation result. However, in GPU-based computations where particle transport times are comparably fast and high speed is a primary goal, the extra overhead of processing of these extraneous particles is generally considered unacceptable.

It is mainly because of the above two issues that integrating a phase-space source into GPU-based MC dose calculations is not straightforward. In this dissertation, three phase-space implementations compatible with GPU-based dose calculation engines are presented. These have been integrated into the latest version of a GPU-based MC code originally developed by Jia *et al.* (2011) [39] called gDPM (now version 3.0).

Previous versions of gDPM used a point photon or electron source model with energies sampled from a spectrum. Due to a lack of treatment head scatter and contaminants in this model, the accuracy was not sufficient for clinical use. In order to improve the dose accuracy, the capability to use phase-space sources was built into the system. The three methods presented below illustrate and address the challenges involved with efficient phase-space source implementations. The first method (section

4.1.2) is based on the standard implementation of most CPU-based dose calculation engines: transporting all particles directly from a plan-dependent phase-space file. However, this was initially so slow that an adaptation was included to sort particles into stacks based on energy and type. This was performed "on-the-fly" on the CPU host, and greatly reduced thread divergence. The second method (section 4.1.3) allows for the use of plan-independent phase-space files (again, particles sorted on-the-fly) by introducing simple secondary collimator and MLC models. The third and particularly novel method (section 4.1.4) involves binning a plan-independent phase-space by particle energy, type and position prior to calculation. The sorted data structure can be then be reused in all subsequent calculations, eliminating the need for on-the-fly particle sorting.

For all of the following work, support for two phase-space file formats has been included: the Electron Gamma Shower (EGS) format (Kawrakow and Rogers 2003 [44]) and the IAEA format (Capote 2007 [17]).

4.1.1 Sorting particles on-the-fly

The computational instructions performed during transport depend strongly on particle energy and type (e.g. photon or electron/positron). To resolve the thread divergence caused by this, a scheme was developed that sorts particles in a phase-space file on-the-fly into different bins of a data array labelled by particle type and energy. Electrons and positrons were grouped together, since the transport mechanics are similar. This technique was used in the first two of the following phase-space implementations, in sections 4.1.2 and 4.1.3.

For a simulation of $N^{simulated}$ particles using a phase-space source containing N^{phsp} particles, consider two arrays P^{γ} and $P^{e\pm}$ for photons and charged particles, respectively. Each array has $N^{energies}$ bins, each of which can store up to $N^{GPU-batch}$ particles. The parameter $N^{GPU-batch}$ is chosen based on the memory available on the GPU hardware, and is the maximum number of particles of a given type that are copied to GPU memory prior to transport simulation. For all simulations in this work, $N^{GPU-batch} = 8388608$. Once $N^{GPU-batch}$ particles have been stacked into one of the bins, or $N^{simulated}$ particles have already been stacked in total, the particle data is copied to the photon and/or charged particle stack in global GPU device memory (P^{γ}_{GPU} and/or $P^{e\pm}_{GPU}$) and the transport kernels are launched. The photons are processed first, including all scattered photons. Secondary charged particles generated

in transport are added to the end of the charged particle stack $P^{e\pm}_{GPU}$.

4.1.2 Plan-dependent phase-space source (PhspB method)

The first and simplest implementation of phase-space sources that will be presented has two goals: (1) enable the use of plan-dependent phase-space sources in a standard format, and (2) avoid the thread divergence issue that occurs when particles from an unsorted phase-space are transported sequentially. In this method, called the PhspB method, it is assumed that the phase-space file was generated with the effects of the secondary collimators included. The inclusion of intensity modulation from MLCs is optional (see section 4.1.7). It is necessary to launch a separate instance of gDPM v3.0 for each phase-space file (typically each field in a treatment plan is associated with one plan-dependent phase-space file). The multiple resulting dose distributions can be simply combined cumulatively.

As mentioned previously, separating photon and electron simulation and grouping particles of similar energy greatly reduces thread divergence. Hence, a bin-sorting of particles from the phase-space source is conducted on-the-fly before launching them for dose calculations. Specifically, a number of bins are allocated that divide the particles by type and energy. Due to the similar transport mechanisms of electrons and positrons, it is not necessary to separate them, so there are only two divisions for particle type (photons, and electrons/positrons). Other particle types are not supported. When the CPU is sequentially loading particles from the file, each particle is placed into the energy bin corresponding to the correct particle type. Note that in gDPM, an optimal buffer size of $N^{GPU-batch}$ particles must be chosen that maximizes hardware utilization while staying within memory limitations. Thus the maximum number of particles a single bin may contain is set to $N^{GPU-batch}$. Once there are $N^{GPU-batch}$ particles in one of the bins, the bin is "full" and the particle data is moved to GPU memory and the particles are transported. This process of sorting some particles from the phase-space file and then transporting them on the GPU is repeated many times until the desired number of particles from the phase-space has been processed, as illustrated in the flowchart 4.1. At the end of this, some bins in the data structure may remain partially filled so these particles are also transferred to the GPU for transport and dose deposition simulation. In the current implementation, particle sorting and other CPU operations are performed in series (not simultaneously) with particle transport on the GPU.

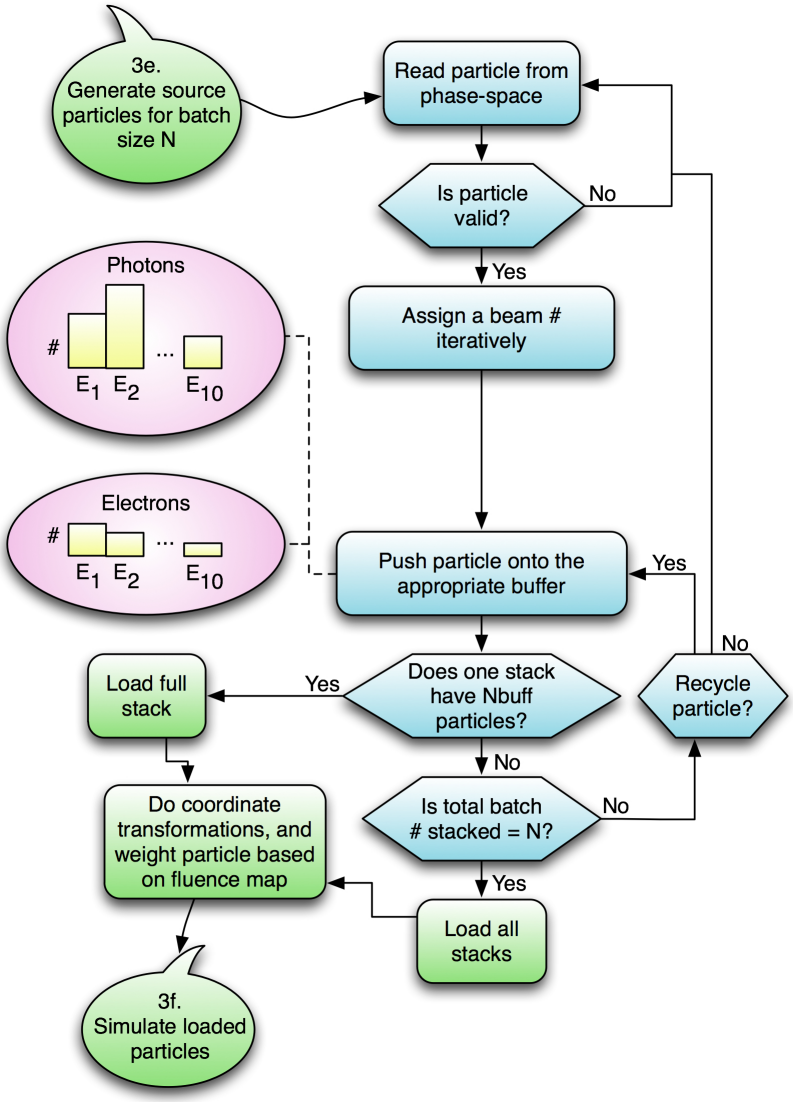

Figure 4.1: A flowchart of source generation in gDPM using a plan-dependent phase-space (PhspB), an addition to figure 2.2.

If plan-dependent phase-space files are available, this method is efficient. This is because the phase-space has already been collimated, so all particles can be loaded directly into GPU memory to be used. Additionally, since the phase-space will probably only be used once, is it more efficient to sort the phase-space on-the-fly than to sort and re-write it prior to simulation. However, the generation of plan-dependent phase-spaces using a code like BEAMnrc is time consuming and not clinically practical. In the following sections two plan-independent phase-space implementations are

presented.

Intensity modulation (i.e. MLC modeling) is achieved by supplying a grid of phase-space modulation factors (fluence map) for each treatment field (section 4.1.7).

4.1.3 Plan-independent phase-space source (PhspA method)

In order to enable the use of plan-independent phase-space sources, a secondary collimator model was integrated into gDPM. In this rather simple model, transmission through and scatter from the secondary collimators was ignored. Instead, simple geometrical tests were used in order to either "accept" particles that passed through the secondary collimator's aperture or "reject" and terminate the particles that intersected the top surface of one of the jaws (assuming these particles to be absorbed in the collimator material). In the flowchart in figure 4.1, this step is performed immediately after assigning a beam number to a particles. Phase-space particles that pass through the jaw openings are projected to the fluence map plane (generally defined at the top surface of the MLCs), to perform MLC modeling (section 4.1.7).

This approximation neglects some contributions to the final dose, but the effect has been shown to be small and mostly limited to distant off-axis regions. Schmidhalter *et al.* (2010) [73] showed that transmission and scattering in the secondary collimators can be ignored and still achieve 95% agreement of 1% / 1 mm gamma criteria with EGSnrc for 6 MV and 15 MV 10×10 cm^2 open fields.

It is worth noting that this approach of sorting particles on-the-fly is redundant when the same phase-space file is re-used for multiple plans. Repeated sorting operations can be avoided by sorting the particles prior to simulation, as presented in the next section.

4.1.4 Plan-independent phase-space-let source (PSL method)

Most clinical applications of GPU-based dose calculation engines involve re-using a single plan-independent phase-space per linac beam model. In these cases, the same phase-space data can be applied to all the beams of the same model and energy in any treatment plan. As mentioned previously, it is sub-optimal to sort particles on-the-fly in this situation. The sorting algorithm only needs to be performed once per phase-space and this can be done at the model commissioning[1] stage before any

[1]Commissioning a phase-space model refers to the process of validating and adjusting the model so that dose calculation results are satisfactory. For example, commissioning may involve repeatedly

dose calculations. Additionally, recall that a significant number of computational operations are spent processing particles that are subsequently removed from the simulation by the secondary collimator model. Since particle transport on the GPU is fast, reading data from the hard disk and transferring it to the GPU can comprise a large portion of the total simulation time. To help resolve these issues, a phase-space sorting utility was developed that reads the original phase-space source and divides it into separate files based on particle type, energy and position, called phase-space-let (PSL) files. A PSL file contains a group of particles that are within the same spatial, energy and particle type bins. These are distinct from beamlets as termed in the literature; beamlets typically refer to phase-space divisions only by position, used in the context of treatment plan optimization (Bush *et al.* 2008a [13]).

Phase-space-let generation

Phase-space-lets (PSLs) are divisions of a plan-independent phase-space based on particle type, energy and position. The divisions by type and energy serve to reduce thread divergence on the GPU. These parameters affect simulation time for each particle based on the number and complexity of interactions that take place. The position parameter is used to avoid spending time processing particles that are expected to be eliminated by the secondary collimator model (section 4.1.4).

To illustrate the PSL generation algorithm, first consider a phase-space divided into two files, one containing photons only, and the other both electrons and positrons. Next, divide each of these files into $N^{energies}$ more files, each containing particles within some energy range (for example, the first of these files will contain only particles with energy in $[0, E^{max}/N^{energies}]$, where E^{max} is the maximum energy in the phase-space file). Now divide each of these energy and type binned files into position bins, such that each position bin i contains only particles in $[x_i, x_i + dx]$ and $[y_i, y_i + dy]$, where x_i and y_i are the minimum x and y boundaries of the position bin with dimensions dx and dy. This process was roughly based on the phase-space sorting software by Bush *et al.* (2008a) [13]. Note that, while the generation of PSLs is similar to the beamlet generation Bush describes, performing dose calculations using PSLs is not as straightforward and the extra dimensions (particularly energy) necessitate substantial internal changes to the dose calculation engine, as described in the next section. In particular, the energy dimension is of critical importance in GPU-based

adjusting particle weights in the phase-space until open field profiles and depth dose curves agree with measurement. Generally this only needs to be performed once per phase-space source.

MC dose calculation, allowing for the simultaneous transport of particles with similar energy.

For the testing cases evaluated in this work, particles within 20×20 cm^2 in the phase-space (residing on a plane at $z = 27.3$ cm) were divided into 400 1×1 cm^2 bins according to the x and y position coordinates. Using $N^{energies} = 10$ energy bins and 2 particle type bins, the total number of PSL files is 8000, all derived from a single patient independent phase-space. In terms of efficiency, the optimal choice of bin sizes depends on the hardware configuration in use (for example, the speed of the hard-disk drive and processor). One may note that reading data from 8000 files is a slow process on most computer systems. However, since the field sizes in a treatment plan are usually require only a small geometrical section of the phase-space, it will only be necessary to access a small subset of PSLs near and inside the field openings (described in more detail below). For example, a 10×10 cm^2 field would use 128 PSL files under this scheme.

Dose calculations using PSLs

Dose calculations begin by reading source particles from the PSL files. Utilizing the division of PSLs by position, only those files that contain particles within a region of interest (ROI) defined by the field size are used. In a multi-field plan, each field may make use of different PSLs. The ROI is a rectangular region determined by back-projecting from the field-centre at the SAD through the jaw openings to the plane where the PSLs are defined. If the area of a given PSL overlaps with the ROI, it will be used (see figure 4.2).

When the user requests a simulation of $N_{total}^{simulated}$ particles, it will be necessary to determine how many particles to read from each of the PSL files. In order to maintain the particle distribution from the original phase-space, it is necessary to either (1) read the same number of particles from either file and adjust the particle weights according to the original number distribution, or (2) adjust the number of particles to read from each PSL file according to the distribution. The second approach is used here, in order to maintain the original number distribution. Consider a patient plan with B fields, and take J_i to be the number of PSL files in the ROI *selected* for the field i. Then the total number of particles contained in the PSLs for all fields is the

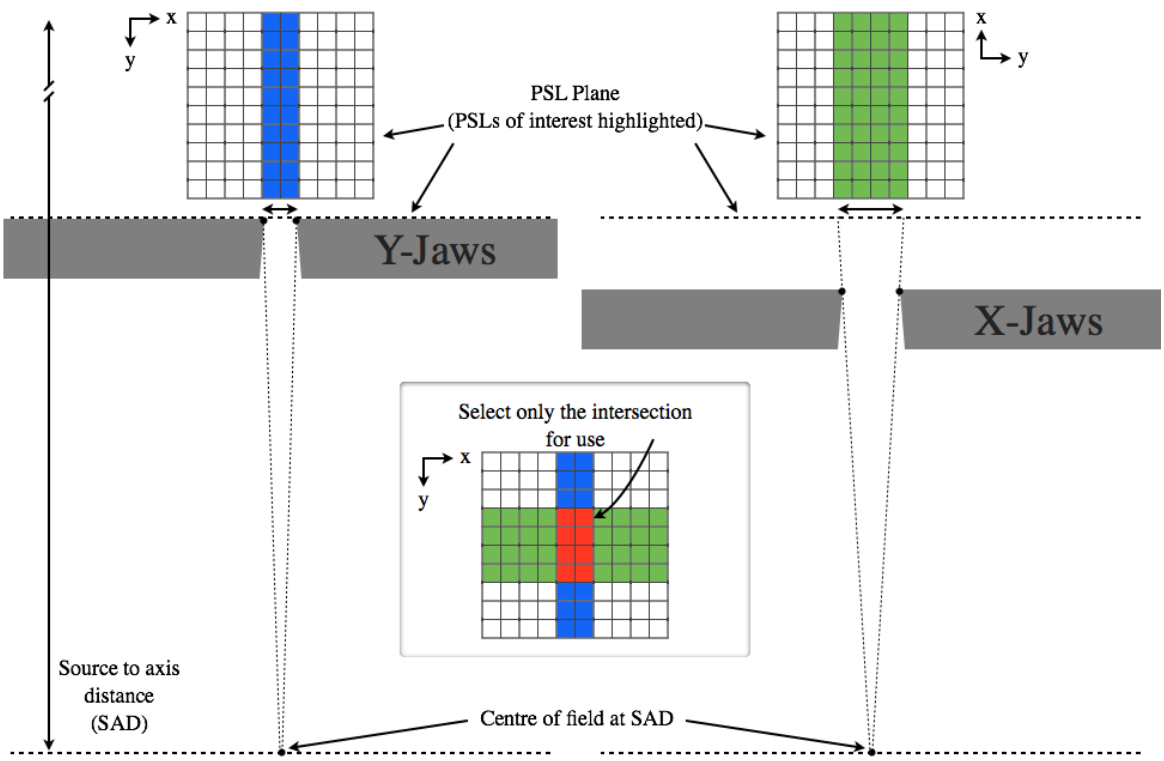

Figure 4.2: An illustration of how phase-space-lets (PSLs) are selected (in position space) based on the secondary collimator settings for each field (not to scale). The centre of the field at SAD is used to define the selected PSLs at the PSL plane, depending on the secondary collimator settings. The final group of selected PSLs is the intersection of those exposed by the X- and Y-jaws.

sum of these,

$$N_{total}^{psl} = \sum_{i=1}^{B} \sum_{j=1}^{J_i} N_j^{psl}, \tag{4.1}$$

where N_j^{psl} is the number of particles in the j^{th} PSL file. Given the total number of particles requested by the user to simulate, $N_{total}^{simulated}$, and the number of photons per PSL, P_j^{psl}, the number of photons to use for simulation from each PSL j per field is

$$P_j^{simulated} = N_{total}^{simulated} \frac{P_j^{psl}}{N_{total}^{psl}}. \tag{4.2}$$

Notice that all fields simulate the same number of particles. The number of electrons to simulate for a given PSL that contains E_j^{psl} electrons is determined by

preserving ratio of electrons relative to photons from the original phase-space:

$$E_j^{simulated} = P_j^{simulated} \frac{E_j^{psl}}{P_j^{psl}}. \tag{4.3}$$

Since the GPU has limited onboard memory, particle data are copied in chunks from CPU RAM to GPU global memory (as a texture memory type). The selected PSLs are used sequentially in the dimensions of particle type and energy. All of the required particles from the PSLs in one energy division are transported before proceeding to the next, to ensure particles of similar energy are transported together. This is illustrated in the flowchart in figure 4.3. For a given PSL, particles are read sequentially and data are added to a single buffer array stored in RAM.

Transporting particles grouped close in position, like in a PSL, can cause a bottleneck when many parallel threads attempt to deposit dose to the same voxel in the shared global dose matrix. This is because only a single thread can write to a particular element of a shared matrix at a time. It is therefore beneficial to transport distant particles in parallel to mitigate memory writing conflicts. To do this, all PSLs in the spatial dimension (for a given energy and particle type) are loaded concurrently, and only a few particles are read from each PSL at a time, iterating through the spatial dimension. Additionally, the beam number is sampled randomly and uniformly for each particle, so that sequential particles are spread between beams.

The first time opening each PSL file, the file pointer is directed toward a random particle in the file. When the end of the file is reached, the file pointer is simply directed back to the start of the file. This avoids the increase of latent variance when jobs are split over multiple GPUs, since it is less likely for the same source particles to be re-used. After each particle is assigned to a treatment field, it is ray-traced to check against the accept or reject secondary collimator model (each field will have corresponding collimator positions). If the particle passes the survival criteria for the field, the particle data are added to the memory buffer that is to be copied to the GPU. Once the buffer is full ($N^{GPU-batch}$ particles have been stacked), the data are copied into memory on the GPU device. When using a PSL source, no particle recycling is performed. This is because recycling with APR was already performed during generation of the PSL source itself.

Once the particle data are on the GPU, each particle receives gantry, collimator and couch rotations for the corresponding field. Its weighting factors are modified

to account for the MUs of the assigned field and any further beam modifiers using a fluence map (section 4.1.7). Finally, each particle is projected to the surface of the patient phantom and transport begins as described in Jia *et al.* (2011) [39]. Dose counters, that record the dose in each voxel of the phantom, are updated using atomic functions[2].

The statistical uncertainty of the dose in each voxel is estimated by considering the standard deviation over a number of independent simulations that are averaged to comprise the calculation result. These independent simulations are performed automatically, and are also referred to as independent batches (not to be confused with GPU-batches which have to do with on-board memory limitations and may not be independent). In the simulations presented in this chapter, 10 independent batches were used.

4.1.5 Dose normalization

The default output of gDPM v3.0 is a relative dose of Gy *per initial particle*, as in other dose engines such as DOSXYZnrc and VMC++ (Gardner *et al.* 2007 [28]). For a photon beam phase-space file generated by simulation of the actual linac geometry, the "initial particles" are the electrons incident on the target. Since not all particles in the phase-space are generally used, the number of initial particles must be adjusted accordingly. To achieve dose per electron incident on the target D, the original dose D_o is divided by the number of incident electrons used to generate the phase-space, $N^e_{incident}$, multiplied by the ratio of the total number of particles in the phase-space, N^{phsp}_{total}, with the number of particles actually simulated (including recycling), $N^{simulated}_{total}$:

$$D = \frac{D_o}{N^e_{incident}} \frac{N^{phsp}_{total}}{N^{simulated}_{total}}. \tag{4.4}$$

Absolute dose conversion is performed using equation 2.22, with measured S_b values.

4.1.6 Particle recycling and azimuthal particle redistribution

The statistical uncertainty that can be achieved in a dose calculation using plan-independent phase-space files is limited by the number of particles contained in the

[2]Atomic functions are specially designed mathematical operations in the CUDA programming language that avoid race conditions in parallel processing.

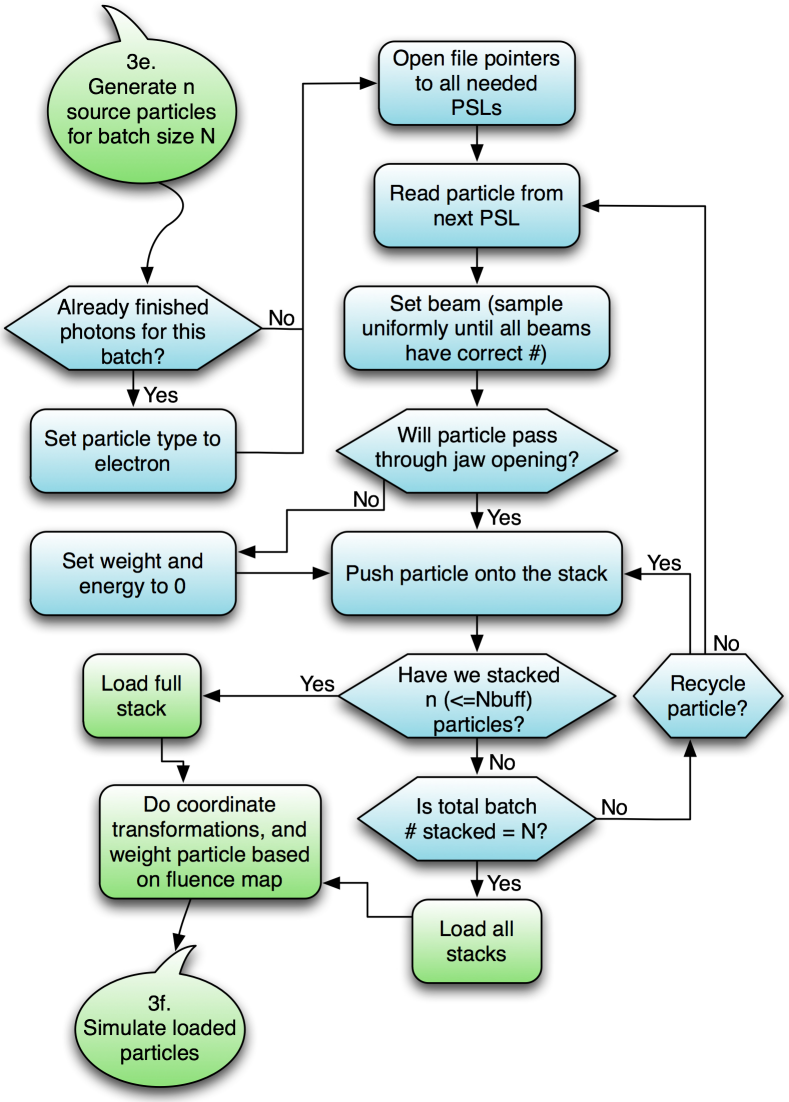

Figure 4.3: A flowchart of source generation in gDPM using PSLs, an addition to figure 2.2.

file. Commonly, phase-spaces contain too few particles to achieve small statistical uncertainty when each particle is only used once. It is therefore a standard practise to recycle particles from the phase-space a technique valid due to the subsequent random processes involved in particle transport. However, there is a limit how how many times a particle can be re-used and still improve statistical uncertainty in the dose distribution. At this limit, the result is subject to phase-space latent variance (Sempau *et al.* 2000 [75]) that can not be reduced by standard particle recycling. In-

creasing recycling beyond this limit results in systematic artefacts, usually noticeable as peaks and troughs in an otherwise uniform dose profile. The artefacts are most prominent near the surface of the phantom where duplicate particles tend to begin simulation in the same voxel.

There are two common solutions to solve this problem: (1) generate plan-independent phase-space containing a greater number of particles, or (2) apply azimuthal particle redistribution (APR) (Bush *et al.* 2007 [16], Brualla *et al.* 2010 [11]) or a similar VRT. The first option is often not possible, for example when the phase-space file has been downloaded from online phase-space repositories and therefore the number of independent particles in the file cannot be increased. The second option, APR, may only be applied if the phase-space is cylindrically symmetric, but this is commonly assumed to be true for modern linac design. APR works by assigning each particle in the phase-space new cylindrically symmetric positions and directions. That is, each particle is uniformly redistributed azimuthally (rotated about the beam axis) with its direction cosines adjusted to maintain the cylindrical symmetry of the phase-space. This allows particles to be recycled while avoiding latent phase-space variance artefacts.

Consider the case where a plan-independent phase-space is available, but it is of insufficient particle density for the desired application. Then the phase-space may be processed and re-written with higher particle density. The algorithm proceeds as follows:

1. read a particle from the input phase-space
2. duplicate the particle $N_{recycle}$ times
3. for each sub-particle:
   (a) decrease the weight by $1/N_{recycle}$
   (b) rotate the particle about the beam axis (update x, y, u and v as per APR)
   (c) write the particle to the output phase-space

The first two phase-space implementations presented (sections 4.1.2 and 4.1.3) perform APR during head model simulation automatically using BEAMnrc, on the plane directly above the upper secondary collimator. For the PSL method (section 4.1.4), APR was performed prior to generating the PSL database to ensure the PSL files contained sufficiency high particle density.

4.1.7 IMRT and VMAT simulation

All methods presented using gDPM share the same method of IMRT and VMAT modeling, described in Jia *et al.* 2011 [39]. Namely, planar fluence maps derived from the planned MLC motions are used to modulate particle weights during transport through a plane defined at the top of the MLC leaf bank. Each fluence map is a 2D matrix that represents cumulative transmission intensities through dynamic MLCs for one field (or control point for VMAT), and is assigned to a single beam angle (which may be an average position for a beam in motion). Fluence map values for a X MU beam range from 0 to X, where X is complete transmission and 0 is complete absorption.

During the source generation stage, each particle is projected to the fluence map plane to identify its corresponding fluence map intensity. The weight of the particle is multiplied by the fluence map intensity before subsequent radiation transport. The final weight is carried throughout subsequent transport process and scales all energy depositions.

4.2 Benchmarking with BEAMnrc + DOSXYZnrc using VIMC

While it would be possible to compare the results of the gDPM phase-space implementations with measurement, the agreement in that case would depend on the accuracy of the original plan-independent phase-space and model used to generate it. Instead, it is more enlightening to compare gDPM results with an independent and trusted MC simulation using the same phase-space source. This also allows us to test gDPM with phase-space sources that have not yet been commissioned to match measurements (of course, this comparison will not detect any systematic issues related to the code used for phase-space generation). To perform this benchmarking, linac head components above and including the secondary collimators were modeled in BEAMnrc using manufacturer specifications. The MLCs were modeled using the vcuDMLCcode software (Keall *et al.* 2001 [47], Siebers *et al.* 2002 [77]). The dose deposition in a voxellized phantom was simulated using DOSXYZnrc. This combination of software, designed as a part of the Vancouver Island Monte Carlo (VIMC) framework, has been extensively tested, and gives accurate results for IMRT and VMAT (Bush *et al.* 2011 [12]).

The DOSXYZnrc input file template is provided below. Substitutable parameters that depend on the treatment plan are shown using a dollar sign and brackets, such as ${PHANTOM} representing the path to the patient phantom file. These substitutions, along with required coordinate transformations (Thebaut and Zavgorodni, 2006 [83]), uncertainty estimations, etc., were performed by WebMC (section A.1.1).

```
DOSXYZ template                          #!GUI1.0
0
${PHANTOM}
0.7, 0.01, 0
0, 0, 0,
2, 2, ${ISOX}, ${ISOY}, ${ISOZ}, ${THETA}, ${PHI}, 55.0, ${PHICOLL}, 0
2, 2, 1, 60, 0, 0, 0, 0
${SOURCE}
${NCASE}, 0, 999, ${RAND1}, ${RAND2}, 100.0, 0, 0, 1, 1, 1, ${RECYCLE}, 0, 0, 1
 ########################
 :Start MC Transport Parameter:

 Global ECUT= 0.7
 Global PCUT= 0.01
 Global SMAX= 5
 ESTEPE= 0.25
 XIMAX= 0.5
 Boundary crossing algorithm= EXACT
 Skin depth for BCA= 0
 Electron-step algorithm= PRESTA-II
 Spin effects= On
 Brems angular sampling= Simple
 Brems cross sections= BH
 Bound Compton scattering= Off
 Pair angular sampling= Simple
 Photoelectron angular sampling= Off
 Rayleigh scattering= Off
 Atomic relaxations= Off

 :Stop MC Transport Parameter:
```

```
 ########################
```

where the substitutable parameters are:

```
${PHANTOM}: The path to the patient phantom file
${ISOX}: The x-coordinate of the isocentre
${ISOY}: The y-coordinate of the isocentre
${ISOZ}: The z-coordinate of the isocentre
${THETA}: The theta angle of the beam
${PHI}: The phi angle of the beam
${PHICOLL}: The phicoll angle of the beam
${SOURCE}: The source model parameters
${NCASE}: The number of histories to simulate
${RAND1}: A random integer
${RAND2}: A random integer
${RECYCLE}: Number of times to recycle particles.
```

4.3 GPU-based hardware

The GPU-based system for this research was named *cavigpumc01* as a contraction of Canada, Victoria and GPU-based Monte Carlo. The purchase of this hardware was supported by the Vancouver Island Center Medical Physics Research Fund. I designed and configured the system, which is a dedicated research resource at VIC. It contains one AMD FX-4100 3.2GHz quad-core processor, 6 GB DDR3 RAM, Kingston HyperX SATA3 Sandforce SSD and two GeForce GTX 580 1.5 GB video cards. The GTX 580 has 16 multiprocessors with 32 cores each and a clock speed 1.54 GHz per core (NVIDIA, 2010). The CUDA version 4.0 was used. When timing comparisons were being performed, only one of the two video cards was used at a time.

4.4 Results

The validity of the three phase-space implementations presented in sections 4.1.2, 4.1.3 and 4.1.4 has been demonstrated by comparison with the BEAMnrc and DOSXYZnrc codes. Profile and depth dose curves have been produced in a homogeneous water phantom along with γ-index tests for validation of the phase-space implementation

methods. Recall that the term used in the γ-index test to quantify spatial misalignment may not be needed when comparing doses that are perfectly registered. However, due to different modeling of geometry (in particular, the secondary collimators) there could be a spatial misalignment between gDPM and DOSXYZnrc results, so γ-index is a appropriate metric.

All simulations shared the same plan-independent PhspA source from a BEAMnrc model of a 6MV Varian Clinac 21EX. Recycling of 20$\times$ was performed in DOSXYZnrc, and automatic recycling for PSLs was used in gDPM.

Note that an independent but similar implementation of the simple secondary collimator model introduced in section 4.1.3 is also evaluated in chapter 3.

4.4.1 Open field profiles, PDDs and output factors

Open fields for 4×4, 10×10 and 30×30 cm^2 fields were calculated using 150 million, 1 billion and 9 billion particles transported in the dose engine, respectively. Depth dose and profile curves in water are shown in figures 4.4 and 4.5. All open field calculations were performed on a homogeneous water phantom comprised of $82 \times 82 \times 82$ voxels with 5 mm voxel size in each dimension. The source to surface distance used was 90 cm, with a source to axis distance of 100 cm.

For clarity, error bars are shown only on the results from the PhspB method (BEAMnrc + gDPM). Estimated statistical uncertainty near the isocentre is less than 1% for all cases. The PhspA and PSL methods are shown in comparison with the standard phase-space implementation (PhspB method) and the benchmark (BEAMnrc + DOSXYZnrc). Both the PhspB method and the benchmark used the same plan-dependent phase-space generated from BEAMnrc, so these results are expected to agree within statistical fluctuations. The PhspA method relies on a simplified jaw implementation that could introduce systematic artefacts, particularly in out-of-field regions. As illustrated in 4.5, these errors are small. The PSL method is expected to have small systematic artefacts from (1) the simple jaw implementation and (2) the exclusion of extra-focal radiation outside selected PSLs.

γ-index tests were performed in 3D on the open field cases, and results are shown in table 4.1. Over 98% of the voxels passed 2%/2mm criteria within the 10% isodose line in all cases. For the stricter criteria of 1%/1mm the success rate was over 95% for all cases. These results, consistent with the statistical uncertainty of the simulations, indicate that all methods are able to produce dose distributions of quality acceptable

for most clinical applications.

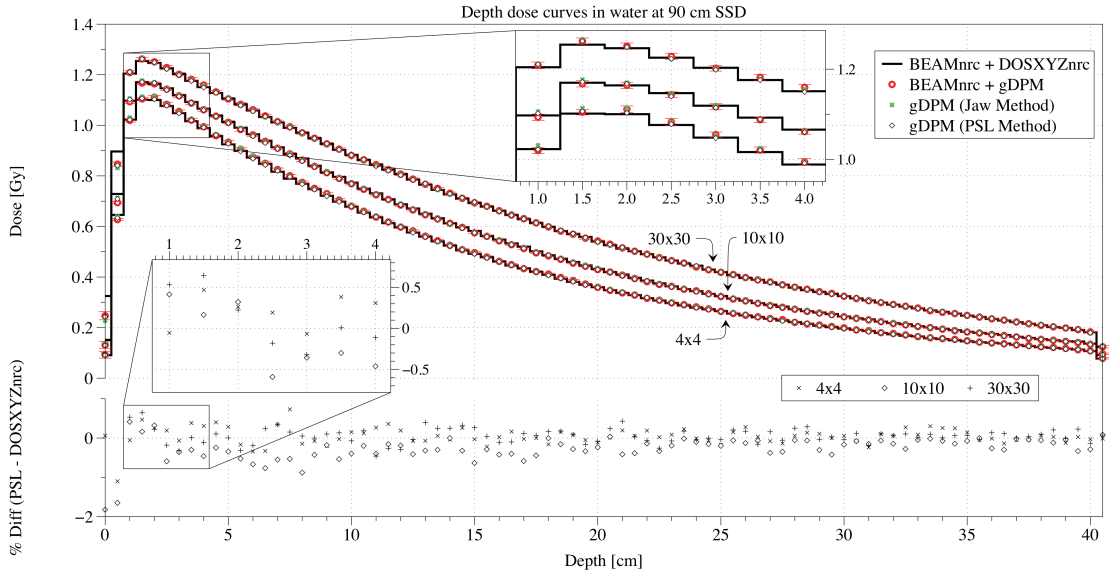

Figure 4.4: Depth dose curves in water for 100 MU at 90 cm SSD for 4×4, 10×10 and 30×30 cm^2 field sizes. The benchmark (BEAMnrc + DOSXYZnrc) is compared with the PhspB method, PhspA method and PSL method. For clarity, error bars are shown only for the PhspB method. The differences between the PSL method and the benchmark are shown for all field sizes, as a percentage of the maximum dose.

Field Size (cm^2)	**Method**	**γ-index 2%/2mm**	**γ-index 1%/1mm**
4×4	PhspB	100.00	99.27
	PhspA	100.00	97.86
	PSL	99.96	95.41
10×10	PhspB	99.66	98.99
	PhspA	99.93	99.08
	PSL	99.92	98.57
30×30	PhspB	99.61	98.84
	PhspA	98.66	95.94
	PSL	98.66	95.26

Table 4.1: γ-index test results for 4×4, 10×10 and 30×30 cm^2 field sizes are shown. Each of the gDPM phase-space implementation methods is compared with the benchmark (BEAMnrc + DOSXYZnrc). γ-indices were calculated in 3D. Two criteria conditions were applied, 2% / 2 mm and 1% / 1 mm, both using data above the 10% isodose only.

Calculation times of the above simulations are shown in figure 4.6. The vertical

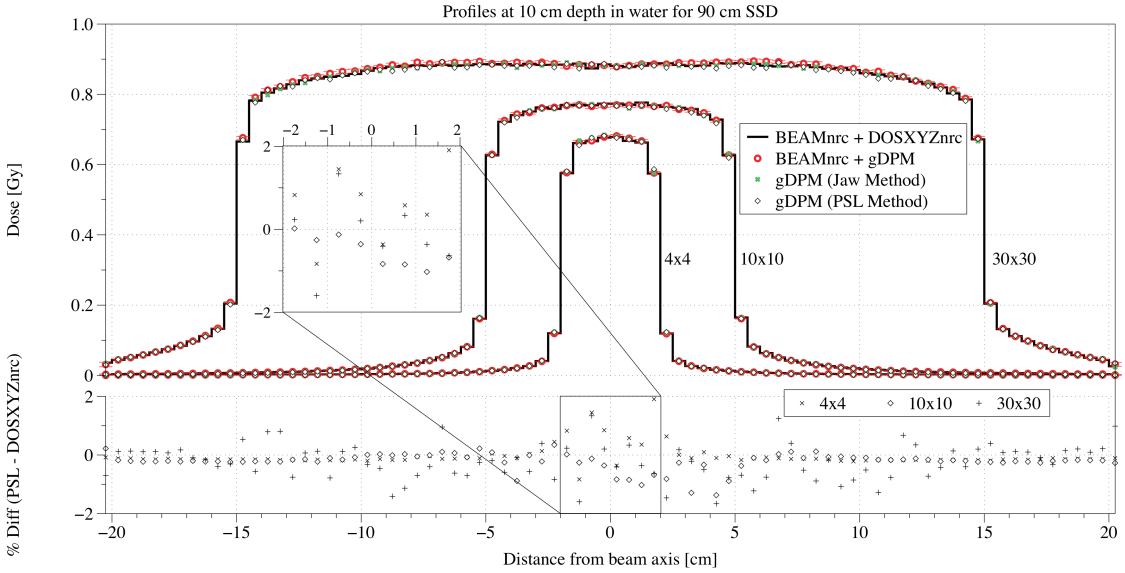

Figure 4.5: Cross-beam profiles in water for 100 MU at 90 cm SSD and 10 cm depth for 4×4, 10×10 and 30×30 cm^2 field sizes. The benchmark (BEAMnrc + DOSXYZnrc) is compared with the PhspB method, PhspA method and PSL method. For clarity, error bars are shown only for the PhspB method. The differences between the PSL method and the benchmark are shown for all field sizes, as a percentage of the maximum dose.

time axis is put on a log scale to better observe the GPU-based results. All times are scaled to correspond to a single processor (i.e. per CPU or per GPU). The number of CPUs used for parallelization was 10, 64 and 128 for the 4×4, 10×10 and 30×30 field sizes, respectively. The simulation time required to generate the PhspA (using BEAMnrc) is not shown, since it is only performed once per accelerator model. The "BEAMnrc" times in figure 4.6 include only simulation of the secondary collimators (since these are open fields, there was no simulation of MLCs).

As expected, the simulation times are longer for larger fields sizes, due to the larger number of particles simulated in order to maintain sufficient particle density and consequently statistical uncertainty in the dose distribution. Since the benchmark (BEAMnrc + DOSXYZnrc) and PhspB method (BEAMnrc + gDPM) use the same plan-dependent phase-space, the BEAMnrc portion of the calculation time is identical in both cases. The calculation times of the BEAMnrc + DOSXYZnrc for 4×4, 10×10 and 30×30 cm^2 field sizes were 4 + 4, 282 + 48 and 294 + 363 CPU hours, respectively (in each pair of numbers, the first is BEAMnrc, the second DOSXYZnrc). In the PhspB method the gDPM execution time was fast: the BEAMnrc + gDPM times were 4 hours + 16 seconds, 282 hours + 102 seconds and 294 hours + 830 seconds per CPU

or GPU for the three field sizes (again in each pair of numbers, the first is BEAMnrc, the second gDPM). The high speed of the gDPM component for the PhspB method is because secondary collimator simulation is performed by BEAMnrc, providing a phase-space for dose calculation that is already plan-dependent. The PhspA method, where secondary collimators are included in gDPM, has slower gDPM execution time, but there is no longer a need for BEAMnrc simulation of collimating devices (and this eliminates the requirement of a CPU-based computing resource). The calculation times were 73, 156 and 861 seconds for the three field sizes, respectively. The PSL method was the fastest of the dose calculation methods with calculation times of 17, 114 and 674 seconds for the three field sizes. The speed enhancement of the last method was particularly apparent for small field sizes, because the particles in PSLs outside the region of interest do not need to be processed. The PSL method also maintained high speed for large field sizes, primarily due to particles being sorted (by energy and particle type) prior to simulation.

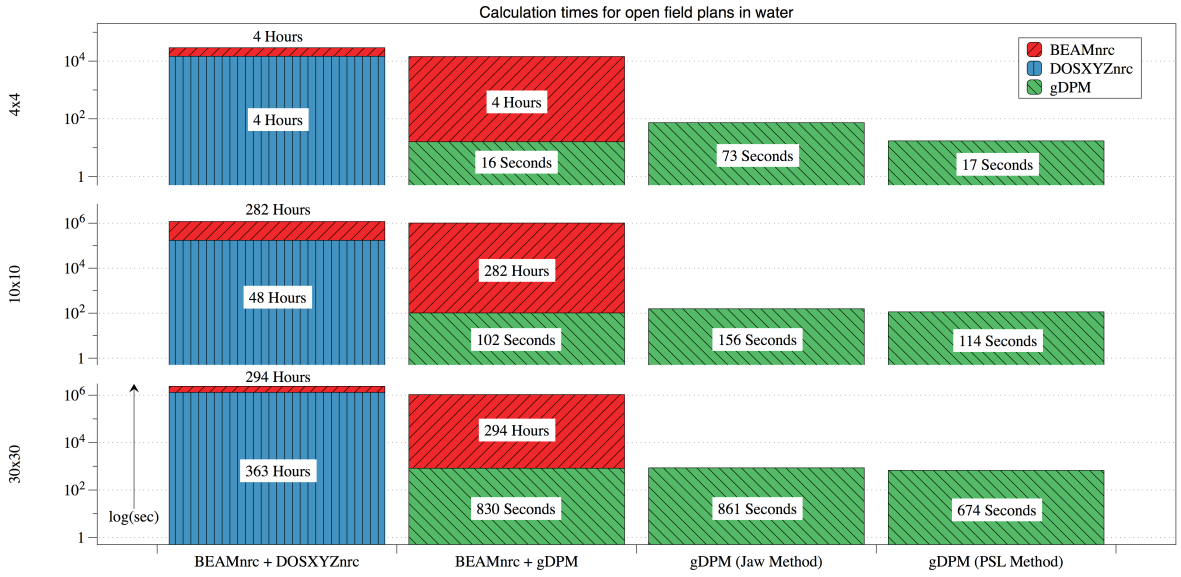

Figure 4.6: Calculation times for 4×4, 10×10 and 30×30 cm^2 open fields. The times are shown on a log scale and separated by software BEAMnrc, DOSXYZnrc and gDPM. The benchmark (BEAMnrc + DOSXYZnrc) is compared with the PhspB method, PhspA method and PSL method. All times are scaled to correspond to a single processor (i.e. per CPU or GPU).

For a range of field sizes, the relative output factors (ROFs) were determined to compare the PSL technique against BEAMnrc/DOSXYZnrc (figure 4.7). The ROF is the ratio of the evaluation dose with a reference dose at a point 10 cm deep along

the beam axis for a 10×10 field. Both methods used the same set of backscatter correction factors that were calculated from MC simulations of the monitor chamber (equation 2.22). The ROF percent differences between codes were less than 1% for all field sizes, which is within the estimated statistical uncertainty.

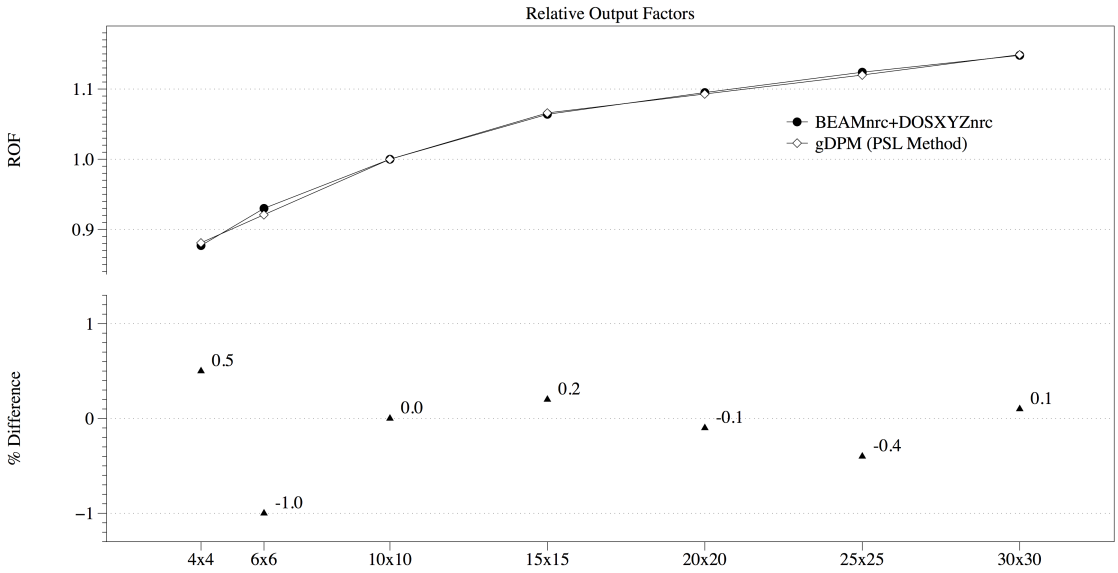

Figure 4.7: ROFs plotted for a variety of field sizes. The benchmark (BEAMnrc + DOSXYZnrc) is compared with the PSL method, and local percent differences are shown.

The ability of gDPM to produce accurate results in heterogeneities was tested using a virtual phantom of tissue, bone and lung. The phantom was $82 \times 328 \times 82$ voxels with $5 \times 1.25 \times 5$ mm^3 resolution. Slabs of different materials were varied along the y-axis: 2 cm of tissue, 3 cm of bone, 7 cm of lung and 29 cm of tissue, as shown in figure 4.8. A dose curve for a 10×10 cm^2 open field was produced for comparison of BEAMnrc and DOSXYZnrc with the PSL method, in figure 4.9. Agreement was within the statistical uncertainty of the simulations (<1%), except near the surface of the phantom. The γ-index and χ-index tests revealed agreement of 98.6% and 99.3%, respectively, above the 10% isodose of the benchmark and 1% / 1 mm criteria. The RMSD was 0.5%.

4.4.2 IMRT patient cases

When considering realistic patient cases to test the PSL source implementation, it is important to note that the gDPM MLC modeling method will result in discrepancies

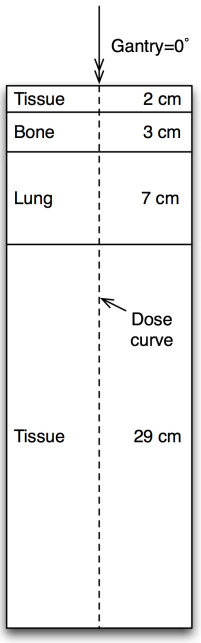

Figure 4.8: Diagram of the tissue-lung-bone phantom used for testing gDPM near heterogeneities. The dose curve location is illustrated with a dashed line.

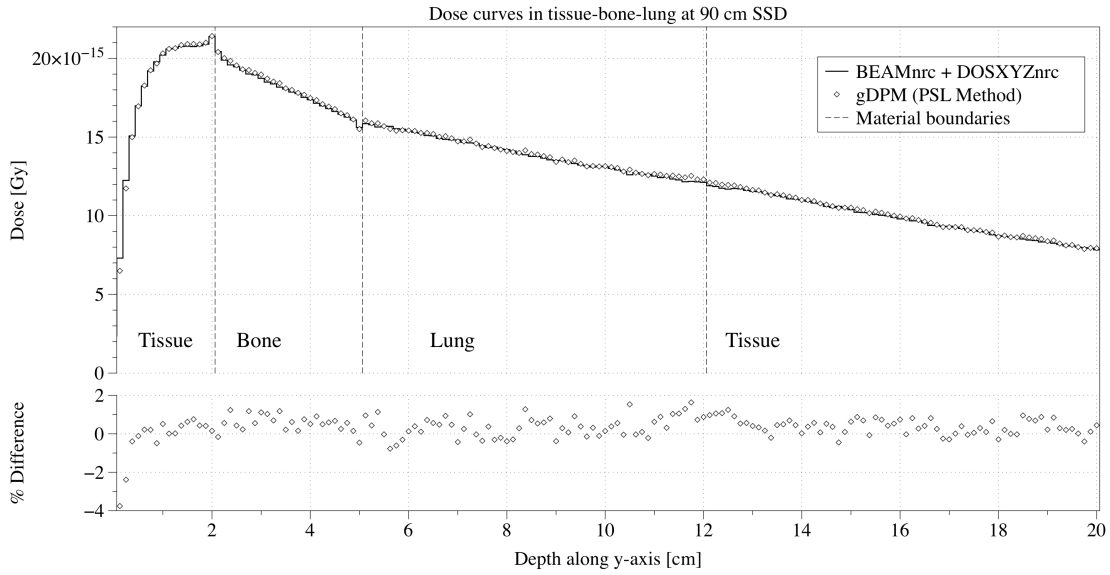

Figure 4.9: Dose curves for a 10×10 cm^2 open field from the BEAMnrc + DOSXYZnrc benchmark and the PSL method. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

when comparing with the VIMC system that was used for benchmarks. Additionally,

differences in transport mechanics have the greatest impact near heterogeneities. For this reason, two IMRT patient plans were simulated in a homogeneous 10 cm radius sphere of water before moving on to a heterogeneous case. The phantom contained $82 \times 82 \times 82$ cubic voxels with 0.25 cm side length. The centre of the phantom was positioned at isocentre, with a SAD of 100 cm.

To test a heterogeneous case, a patient phantom for an IMRT tongue treatment was built from CT scan. The resolution was down-sampled to $154 \times 90 \times 109$ voxels, each with a size of $3 \times 3 \times 3$ mm^3. In all cases, the estimated statistical uncertainty was <1% near the isocentre. Agreement statistics are presented in table 4.2, and figure 4.10 provides an isodose comparison. Notice that there are some dose differences in air near the trachea. For each of the IMRT cases, decreasing the statistical uncertainty (increasing the number of simulated particles) was not found to improve agreement, indicating that the discrepancies were systematic. Since heterogeneous open field cases provided good agreement, the discrepancies have been attributed to the differences in MLC modeling methods.

Case	**RMSD**	**γ-index**	**χ-index**
		2% / 2 mm	**2% / 2 mm**
Homogeneous 1	1.45	99.38	97.44
Homogeneous 2	1.84	98.51	94.25
Heterogeneous	2.00	96.67	94.30

Table 4.2: RMSD, γ-index, and χ-index tests using 2% / 2 mm criteria above the 10% isodose, comparing the PSL method against the VIMC benchmark of DOSXYZnrc + BEAMnrc + vcuDMLCcode.

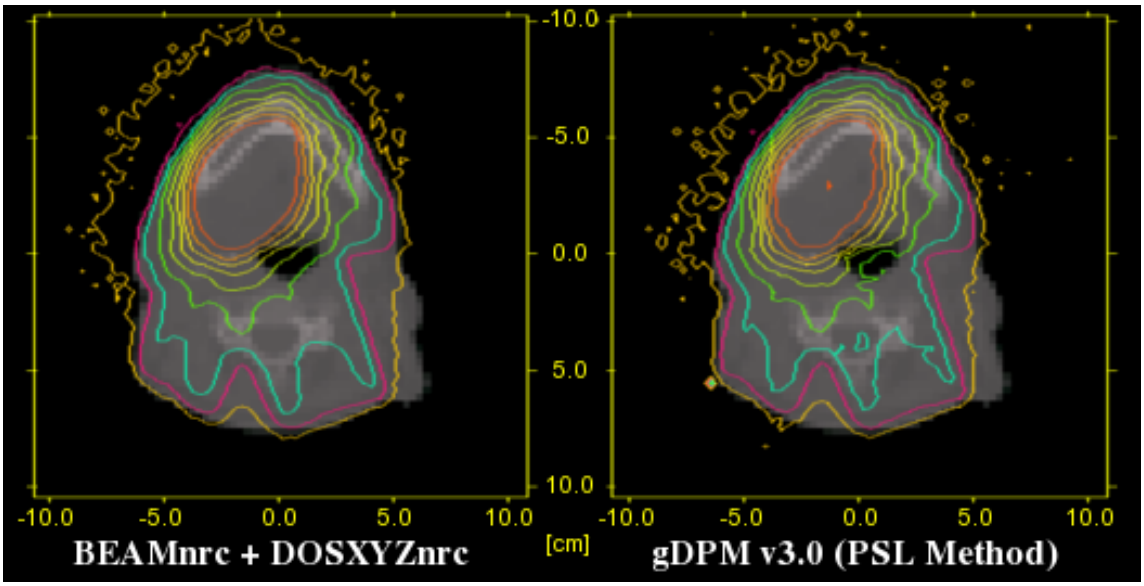

Figure 4.10: Isodose curves for a clinical IMRT tongue treatment, with BEAMnrc + DOSXYZnrc + vcuDMLCcode results pictured left and PSL results on the right. The voxel size of the phantom was $3 \times 3 \times 3$ mm^3, and estimated statistical uncertainty at the isocentre was 1% for both cases.

4.5 Discussion and conclusions

The test-cases presented above provide an initial validation of the capability of gDPM v3.0 using phase-spaces and PSL sources to reproduce dose distributions in agreement with standard MC simulations. The results for a single linac model were presented, but agreement is likely to be comparable for linacs of similar design. Due to the simple secondary collimator, models that contain larger extra-focal[3] radiation contributions may be less successful. However, since modern linac designs tend to be similar, and the energy range of therapeutic beams is not so large, it is expected that all of the source models presented above will retain good accuracy.

It was demonstrated that simulation times can be greatly decreased using the novel method for pre-processing phase-space sources, the phase-space-let (PSL) technique. This method potentially introduces a new source of systematic error (on top of the jaw model and fluence map approximations) because PSLs outside the field dimensions are excluded from the simulation. This is different from the perfect collimator approximation (evaluated in the previous chapter) because PSLs excluded from the

[3]Focal radiation is defined as bremsstrahlung photons resulting from interactions of the initial electron beam in the target. All other particles are then called extra-focal.

simulation may still include some particles that would pass through the secondary collimator openings. However, if PSL selection is performed conservatively, the lost extra-focal contributions should be small with respect to the total field dose and have negligible effect on most clinical cases. Our current strategy for selecting PSLs is to project from the field centre at SAD through the field boundaries and up to the PSL plane. This includes most of the extra-focal contributions for the source model tested. Note that the area of the selected PSLs can be reduced through use of smaller PSL spatial divisions. For example, using a bin size of 5×5 mm^2 instead of 1×1 cm^2 would improve the conformity of the PSL boundaries to the desired area of interest. However, whether or not this provides a performance gain will depend on the given phase-space and the server hardware. Hardware considerations play a similar role in deciding the size of energy bin to use when generating the PSL data structure. Using a small bin size results in improved hardware utilization on the GPU, but the extra processing overhead and increased memory usage can be drawbacks. These parameters are best determined empirically. For our system and 6MV phase-space, it was found that 1×1 cm^2 PSLs with 10 energy divisions provided the fastest simulation of a 10×10 cm^2 field. For other hardware configurations some benchmarking may be necessary to achieve the most desirable configuration. Since the treatment plan field sizes also play into this consideration, it may be optimal to generate several PSL data structures with varying parameters, but this would require significant hard-disk drive (HDD) storage requirements.

Chapter 5

A hybrid phase-space and histogram point source model

In chapter 4, the challenges of using phase-space sources in GPU-based dose calculation engines were presented. It was found that binning the phase-space particles according to type, energy and position using the phase-space-let (PSL) technique improved the efficiency of calculations. However, PSL source implementations are still a bottleneck in GPU-based Monte Carlo simulations (as much as 60% of the total simulation time). In this chapter, a new source modeling approach will be considered, called the hybrid source method. The PSL method will be used only for extra-focal radiation, a small component of the beam extracted from the original phase-space. The extra-focal PSL source will then be combined with a simple beam characterization of the focal source.

A subset of the results from this chapter were published in Townson and Zavgorodni (2014) [85].

5.1 The motivation for a hybrid focal/extra-focal photon radiotherapy source model

Recall that gDPM v2.0 was designed with only a photon point source. This source, while not an accurate representation of an actual treatment beam, generated photons faster than a PSL source, due to utilization of the GPU for generating photon data instead of reading it from a file. Since a significant portion of a phase-space is focal radiation (roughly 70-90%), replacing focal photon generation with a similar

GPU-based scheme would significantly reduce source generation time. Additionally, since focal photons originate from a small spot on the target, it may be possible to accurately represent the focal source with a simple point source.

For source models derived from phase-spaces, the capability to determine the interaction histories of particles is typically necessary. Since phase-spaces contain a randomly ordered assortment of particles, it is generally not possible to perfectly separate particles associated with focal or extra-focal components of the source unless these particles are marked in a specific way. For cases where the linac treatment head geometry is available, a particle transport code such as BEAMnrc (Kawrakow and Walters 2006 [46]) can be used to produce a phase-space where each particle contains an inheritable record of its history encoded in a 32-bit variable, called the LATCH. If configured to do so, the LATCH parameter records the components of the treatment head where each particle interacted. However, phase-space models provided by linac manufacturers (in place of treatment head geometry specifications) do not usually contain LATCH records. This makes it difficult to build phase-space based multi-component source models for modern machines.

This chapter presents a simple phase-space based model that splits the particle phase-space into focal and extra-focal components. The focal photons were represented as a point source while extra-focal radiation was modeled using a small phase-space. In section 5.2.1, a ray-tracing method is introduced that was used to identify focal photons in phase-spaces not containing LATCH records. The focal photons were then used to derive angular-dependent spectral properties that were represented in a 2D histogram. The remaining extra-focal particles from the original phase-space source were processed to form a PSL source. The hybrid source model was integrated into the GPU-based dose calculation engine gDPM v3.0 (Townson *et al.* 2013 [84]) and tested against standard PSL sources for both Varian 21EX Clinac and TrueBeam (Varian Medical Systems, Palo Alto, CA, USA) machines.

5.2 Dose calculations using a hybrid source model

When using the hybrid source model, dose calculation proceeds in two stages. First, the point source photons are generated on the GPU and transported through a voxelized phantom to calculate the focal component of the dose in each voxel, D^{focal}. The extra-focal particles, contained in a small phase-space, are sorted prior to dose calculation to form a PSL source. After the focal dose component has been calculated, the

PSL source is read from the HDD and transported on the GPU through the phantom to produce the extra-focal component of the dose per voxel, $D^{extra-focal}$. The total dose for the hybrid source, D^{hybrid}, is then the sum of these two dose components (in units of Gy per initial electron),

$$D^{hybrid} = D^{focal} + D^{extra-focal}. \tag{5.1}$$

5.2.1 Dividing a phase-space source into focal and extra-focal components

Typically, source models derived from phase-spaces use the LATCH variable to group particles based on their region of last interaction (e.g. target, primary collimator, flattening filter, etc.). The source particle groups can then be characterized independently. However, this process relies on LATCH records, which may not be available. Instead, a method has been developed to extract a large group of particles from a non-LATCH phase-space that can be modeled by a point or other target source. For each photon in the phase-space, (1) photons are ray-traced upstream to the bottom of the target, then (2) if the photon path intersects the target within a small circle of diameter d around the beam-axis, it is assumed to be a 'focal' photon. Photons that are projected to fall within d are extracted from the original phase-space and recorded into a new 'focal source' phase-space for further analysis, as illustrated in figure 5.1 (left). Since there is no way to verify that the selected photons only interacted in the target, there will be a small number of extra-focal photons included in the focal source. However, this does not compromise the model, as their energy-radial distribution will be accounted for automatically. The remaining particles in the original phase-space (extra-focal photons and electrons) are recorded to a second 'extra-focal' phase-space. While it may also contain a small number of primary photons, they similarly do not compromise the model. There is no double-counting of particles; each photon is used in only one of either the focal or extra-focal sources. The focal source phase-space is subsequently used to derive the energy and angular characteristics of the point source (as described in the following section), and particles from the extra-focal phase-space are used to generate a PSL source.

To better understand the composition of the photons selected for our focal source model (e.g. whether they indeed originate from the target or elsewhere), this procedure was tested on a phase-space containing LATCH records that allowed for photons

coming directly from the target to be separated from extra-focal photons. When performing the ray-tracing method to select focal photons, the LATCH was also analyzed to determine if the photon last interacted in the target (i.e. was actually a focal source photon). This analysis enabled the evaluation of the ray-tracing method in its ability to accurately separate focal and extra-focal components of the phase-space.

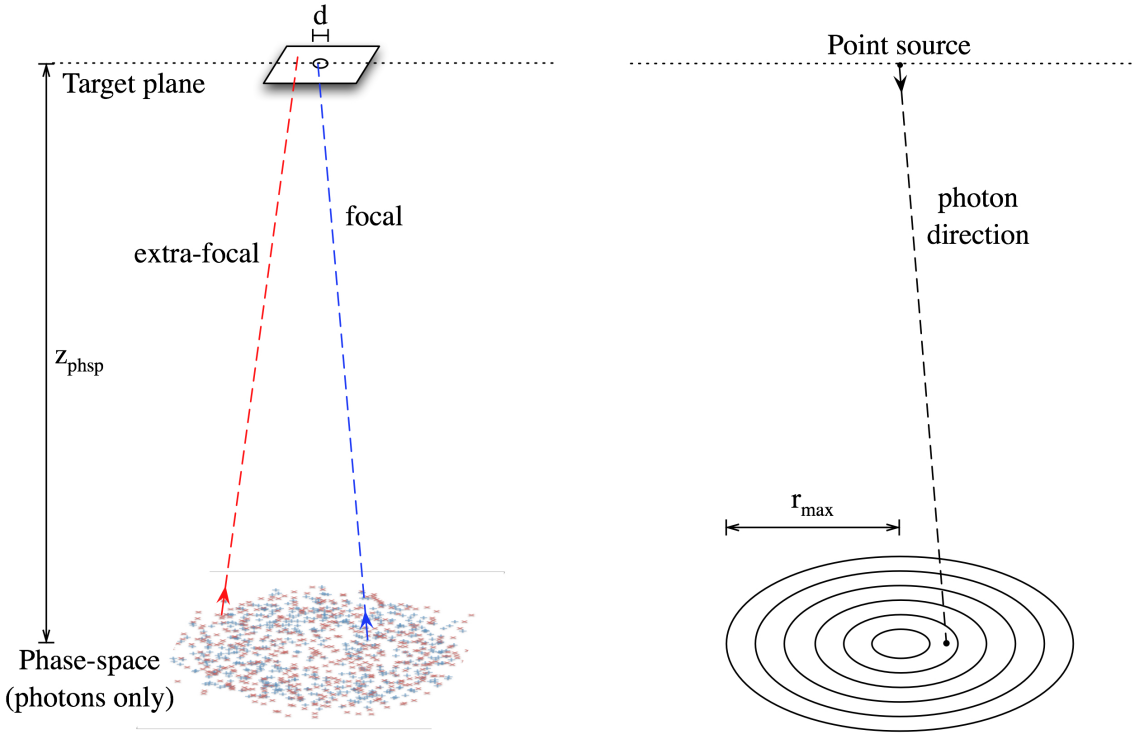

Figure 5.1: Photons selected for the point source model are illustrated (left). Only those photons that ray-trace within a circle of diameter d at the target are identified as representing the focal source. After characterizing these photons using radius and energy histograms, a point source model is used for photon generation (right). After the energy, radius and azimuthal angle are sampled, the point source assumption gives the photon direction. Not to scale.

5.2.2 A point source model derived from the focal radiation components of a phase-space

In a typical radiotherapy treatment beam, the photon energy spectrum varies with off-axis distance. Therefore, for a point source model, radius dependent photon energy sampling is required. In order to achieve this, the focal phase-space was processed and

2D histograms were constructed to provide an energy spectrum for each radial bin (on the same plane as the original phase-space). For the Varian linac models considered in this work, it is generally accepted practise to assume cylindrical symmetry of the source above the secondary collimators. With this assumption, the energy spectra of the photons in a phase-space can be considered to be independent of azimuthal angle (rotations about the beam axis).

To make the simulations efficient on a GPU, photons from the a given energy bin j were generated and transported simultaneously. When particles were generated for multi-beam treatment plans, the beam number was uniformly sampled for each unique photon. Once the energy bin was determined, the radial bin for each photon was sampled on the GPU using the Metropolis algorithm and the histogram N_{ij}^{focal}. The radius within the bin i was sampled uniformly. The exact energy within the energy bin j was also sampled uniformly. Finally, the location of each photon was determined by uniformly sampling its azimuthal angle about the beam axis. By assuming the photon originated from a point at the bottom of the target and on the central axis, the combination of radius and azimuthal angle determines the photon direction, as illustrated in figure 5.1 (right).

In order to maintain un-biased energy-radial characteristics of the sampled photons, the number of photons to be sampled from each energy bin must be calculated. To determine this quantity, first note that the original phase-space, containing N^{phsp} particles, was split into two phase-spaces: the first one including N^{focal} focal photons and the second including $N^{extra-focal}$ extra-focal particles. The focal photon phase-space is not kept it is used only to produce a histogram characterization of the source. The energy dependence on radius of these photons was characterized by building photon energy spectra for each of $N^{annuli} = 300$ annuli out to $r_{max} = 10$ cm radius from the beam axis at the phase-space plane (which was defined at a distance of $z_{phsp} = 27.3$ cm from the bottom of the target). In this work, $N^{energies} = 20$ energy bins were used for each energy spectrum. The relative dose contribution of the source component is proportional to the ratio

$$F = \frac{N^{focal}}{N^{phsp}} = 1 - \frac{N^{extra-focal}}{N^{phsp}}. \tag{5.2}$$

To account for the relative contributions of each energy bin in the 2D radial-energy histogram of focal photons, the number of photons were counted in each bin with radial index i and energy index j, the totals denoted per bin by N_{ij}^{focal} (an

example is provided in figure 5.2). The number of photons to simulate per energy bin is then given by

$$N_j^{focal/sim} = F \cdot N^{requested} \frac{\sum_{i=1}^{N^{annuli}} N_{ij}^{focal}}{N^{focal}}, \tag{5.3}$$

where the total number of particles the user requested for simulation is $N^{requested}$. The quantity $N_j^{focal/sim}$ may be greater than the number of particles that can be efficiently transported in parallel on the GPU. In this case, particle source generation proceeds in batches, alternating between generation and transport. To account for photon weights from the original phase-space, the average photon weight per bin was recorded during the initial phase-space processing as W_{ij}^{avg}. In subsequent simulations, each time a photon was generated for a given bin its weight was set to W_{ij}^{avg}.

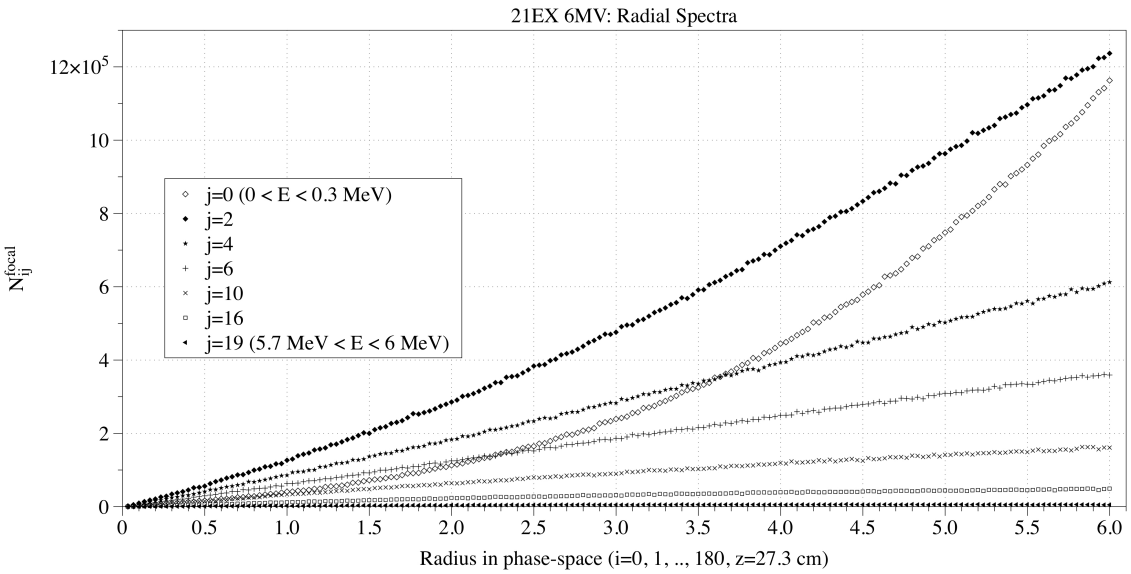

Figure 5.2: Radial distributions of focal photons plotted for a number of energy bins, as derived from a 21EX 6MV phase-space. The source originated from full MC simulation of the linac head model, and was scored just above the secondary collimators.

The dose calculation engine makes some approximations that allow for further efficiency enhancement: (1) scattering in air upstream of the phantom is ignored; (2) secondary collimators absorb 100% of particles; (3) multi-leaf collimators are modeled using a fluence map. This means that it is not necessary to generate photons directed outside the maximal collimator opening, since they would be guaranteed to not contribute to dose. Thus a maximum radial bin index i_b^{max} was defined for each beam b as the largest radial bin index i where a ray from the point source to

anywhere in the bin also passes through the collimator opening. Note that the radial bin indices are defined as increasing outward from the central axis. Focal photons were generated only in the reduced number of radial bins where $i \leq i_b^{max}$. If not corrected for, this would introduce a bias between beams, dependent on i_b^{max}, due to the uniform sampling of beam number. For example, smaller field sizes receive the same number of particles as larger field sizes, despite the smaller area. This bias was corrected for by modifying photon weights using the ratio of the number of photons inside i_b^{max} to the total for each energy bin. The weight of each particle was multiplied by the factor R_{jb}:

$$R_{jb} = \frac{\sum_{i=1}^{i_b^{max}} N_{ij}^{focal}}{\sum_{i=1}^{N^{annuli}} N_{ij}^{focal}}. \tag{5.4}$$

Particle weights were further modulated using fluence maps to simulate MLC motion for IMRT and VMAT treatment plans. Fluence maps included modeling of transmission and rounded leaf-ends. Since the method for fluence map generation is independent of the source model, it will not be considered in detail. The MLC model has not been changed from gDPM v2.0.

The pseudo-code in algorithm 2 shows how primary photons are generated using a point source. Photon energies are grouped by iterating through energy bins of the total energy spectrum using the CPU. During photon generation on the GPU, the treatment field and radial bin are sampled. Transport then proceeds, all in a batched format.

After transporting the focal photons, the dose deposited in each voxel, D_o^{focal}, is scaled by the ratio of the total number of photons in the original phase-space to the number of photons requested. This maintains the normalization of the dose as more or fewer particles are simulated. The dose is also normalized by the number of initial electrons incident on the target, $N^{initial}$ to obtain the standard output of MC dose calculation engines in Gy per initial electron. It is necessary to perform this normalization before adding to the total dose, D^{hybrid}, because the scaling factor may be different for hybridized phase-space or phase-space-let sources. The focal dose contribution per initial electron incident on the target is

$$D^{focal} = D_o^{focal} \frac{N^{phsp}}{N^{requested} N^{initial}}. \tag{5.5}$$

Algorithm 2 Hybrid Source: focal photon generation

```
1: procedure GenerateFocalPhotons
2:     for each integer j in N^energies do
3:         Calculate N_j^focal/sim;
4:         for each batch of photons until all N_j^focal/sim are simulated do
5:             for each photon <<< parallelized on GPU >>> do
6:                 Sample the beam number b uniformly;
7:                 Sample the radial bin i using the histogram N_ij^focal;
8:                 Sample radius within bin i uniformly;
9:                 Sample the energy within bin j uniformly;
10:                Sample the azimuthal angle φ uniformly;
11:                Calculate direction cosines, assuming a point source;
12:                Apply secondary collimator and MLC models;
13:                Store surviving photons in global GPU memory;
14:            end for
15:            Transport the batch <<< parallelized on GPU >>>;
16:        end for
17:    end for
18: end procedure
```

5.2.3 Hybridizing with phase-space-let sources

The number of particles to transport from the extra-focal PSL source, $N^{extra-focal/sim}$, is calculated by considering the number of extra-focal particles in the original phase-space source. However, it may be beneficial to also scale $N^{extra-focal/sim}$ independently of the focal source, potentially saving CPU time if the user chooses to adjust the statistical uncertainty of this portion of the simulation. This is achieved using the extra factor U. The total number of particles to simulate using PSLs was therefore calculated as

$$N^{extra-focal/sim} = \frac{(1-F)N^{requested}}{U}. \tag{5.6}$$

In the case where $N^{extra-focal/sim} > 2N^{extra-focal}$, particle recycling is used, and upon running out of particles in a given PSL recycling will restart from the first particle in the PSL. The extra-focal dose contributions $D_o^{extra-focal}$ were also normalized to Gy per initial electron before being added to the total dose distribution,

$$D^{extra-focal} = D_o^{extra-focal}\frac{N^{extra-focal}}{N^{extra-focal/sim}N^{initial}}. \tag{5.7}$$

Dose conversion of D^{hybrid} to absolute units was performed using equation 2.21.

Corrections for backscatter signal into the monitor chamber (equation 2.22) were not included in the results of this chapter.

5.3 Comparisons of the hybrid source with a PSL source

The hybrid source model, as implemented in gDPM v3.0, was compared with the PSL source model in terms of efficiency and accuracy. The PSL method was also used for the extra-focal component of the hybrid source. In this configuration, the expected accuracy of the hybrid source would collapse to that of the PSL source when the parameter d is reduced. The choice of PSL source was made in order to obtain the fastest possible calculation times.

5.3.1 Evaluation of contamination in focal source model

Phase-spaces with LATCH records for 6MV and 18MV Varian Clinac 21EX models were used to test the ray-tracing method for extracting a focal source. For the 6MV and 18MV beams, it was found by using the LATCH variable (section 5.2.1) that the percentage of particles in the phase-space that were photons originating from the target were 85% and 75%, respectively. Of these, 99.9% and 98.7% could be ray-traced to the target and fall within the circle defined by the diameter $d = 0.9$ cm, also respectively. Of all the photons within the focal spot, 0.18% and 0.28% for 6MV and 18MV beams respectively did not satisfy the LATCH criteria to be a focal photon (i.e. were extra-focal contamination). These results indicate that the ray-tracing method of identifying focal spot photons works well for the considered linac models and choice of d.

Similar tests were not performed on the Varian TrueBeam model, because geometry specifications are not available to enable full MC simulation with LATCH records included.

5.3.2 Energy distribution in the source

Recall that the point source model includes an energy-dependent radial component. While additional modeling complexity increases the simulation times, the radius-energy correlation was determined to be essential for accurate source reconstruction.

The mean energy is shown as a function of radius at the phase-space plane for focal photons (figure 5.3) and target plane (figure 5.4) for a variety of accelerator models. To obtain the distributions at the target, the photons were projected using their directions in the phase-space, without accounting for any scattering. Notice how the curves at the phase-space plane are notably flatter near the beam-axis when compared to the mean energies below the target, illustrating a correlation of the energy distribution with position and direction.

5.3.3 Dose calculation results

The hybrid source model was tested against gDPM v3.0 using PSLs, with both source models derived from the same initial phase-space. The focal component of the source was derived using the selection parameter $d = 0.9$ cm, and the extra-focal component was made up of the remaining particles that were processed to form a PSL source. The value of d was chosen to encompass the majority of the focal spot for a range of beam energies. The same value of d was used for all test cases, except where specified otherwise.

The beams modeled included 6MV and 18MV of a Varian Clinac 21EX as well as 6MV, 10MV, 10MV flattening filter free (10MV-FFF) and 15MV from a Varian TrueBeam linac. Profiles and depth dose curves for 4×4, 10×10 and 30×30 cm^2 open fields were obtained by simulating 400 million, 1 billion and 9 billion particles, respectively. A homogeneous virtual water phantom positioned at a 90 cm SSD and comprised of $82 \times 82 \times 82$ voxels with 5 mm voxel resolution. A water phantom with $133 \times 133 \times 82$ voxels of $3 \times 3 \times 5$ mm^3 resolution was also used. The focal component of the hybrid source was tested with focal photon recycling set to 0 and 32.

For all cases, the estimated statistical uncertainty near the isocentre was less than 1%. The energy cut-offs ECUT and PCUT were 0.01 MeV and 0.7 MeV, respectively. PSL sources used automatic recycling (calculated based on the PSL size and requested number of particles). MC simulations used a block size of 64 threads for each of photon and charged particle transport, which are performed sequentially. The source and secondary particles of each type were stored in buffers on the GPU containing up to $N^{GPU-batch}$ particles.

Open field results are shown in figures 5.5-5.16 for the 21EX 6MV, 18MV and TrueBeam 6MV, 10MV, 10MV-FFF, and 15MV accelerators. All test cases achieved greater than 97% χ-index test (Bakai *et al.* 2003 [4]) agreement above the 2% isodose

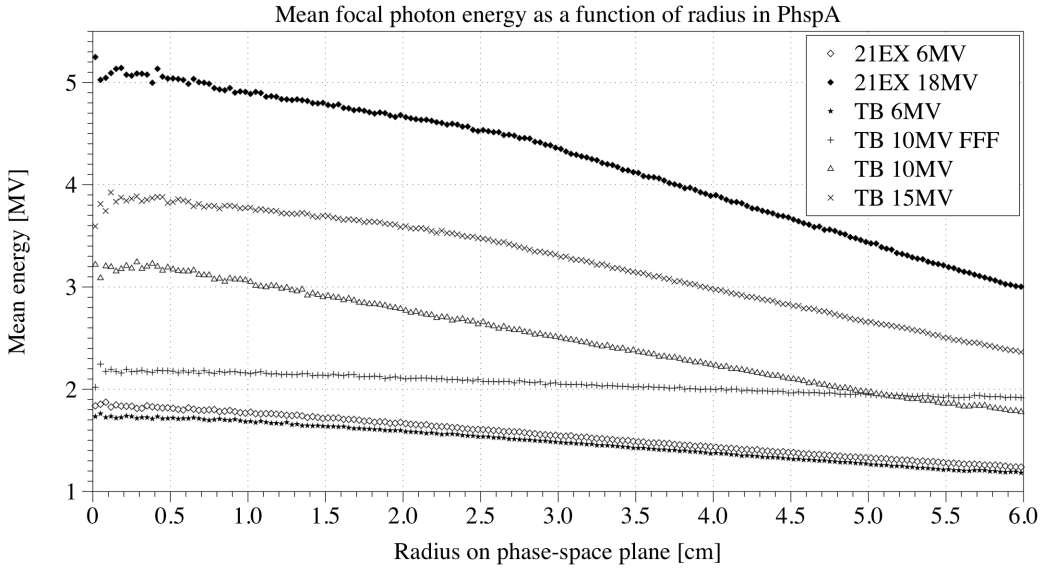

Figure 5.3: Mean focal photon energy as a function of radius for a variety of machines and energies. All phase-space planes were at 27.32 cm from the bottom of the target.

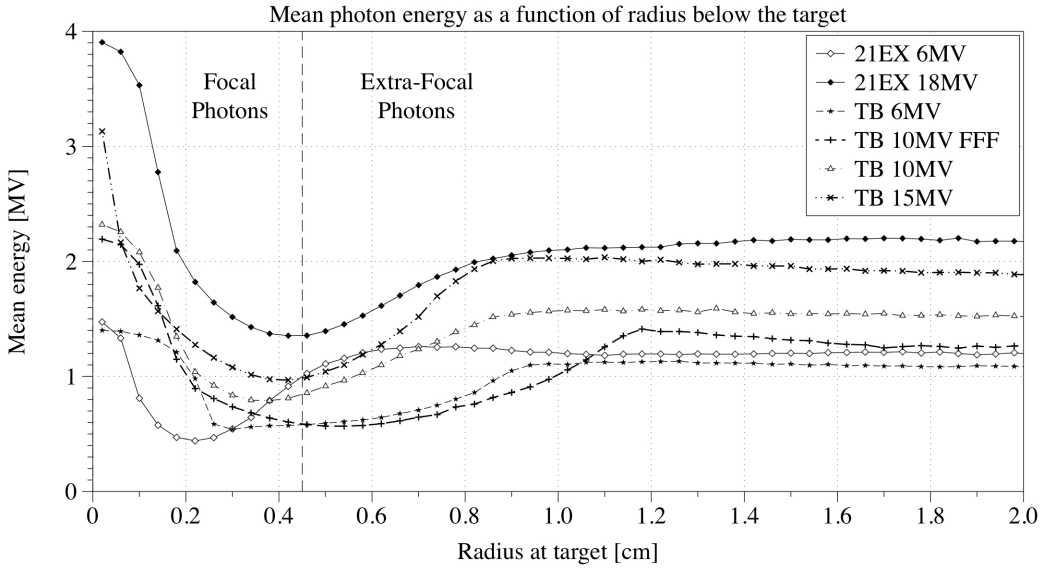

Figure 5.4: Mean photon energy as a function of radius for a variety of machines and energies. Photons were ray-traced to the target based on their positions and directions in the original phase-spaces. A vertical dashed line has been included to illustrate the division of focal and extra-focal particles for the hybrid source with $d = 0.9$ cm.

with 1% / 1 mm criteria. The mean χ-index test agreement was 99%. The RMSDs were less than 1%, with a mean value over the test cases of 0.5%. The largest

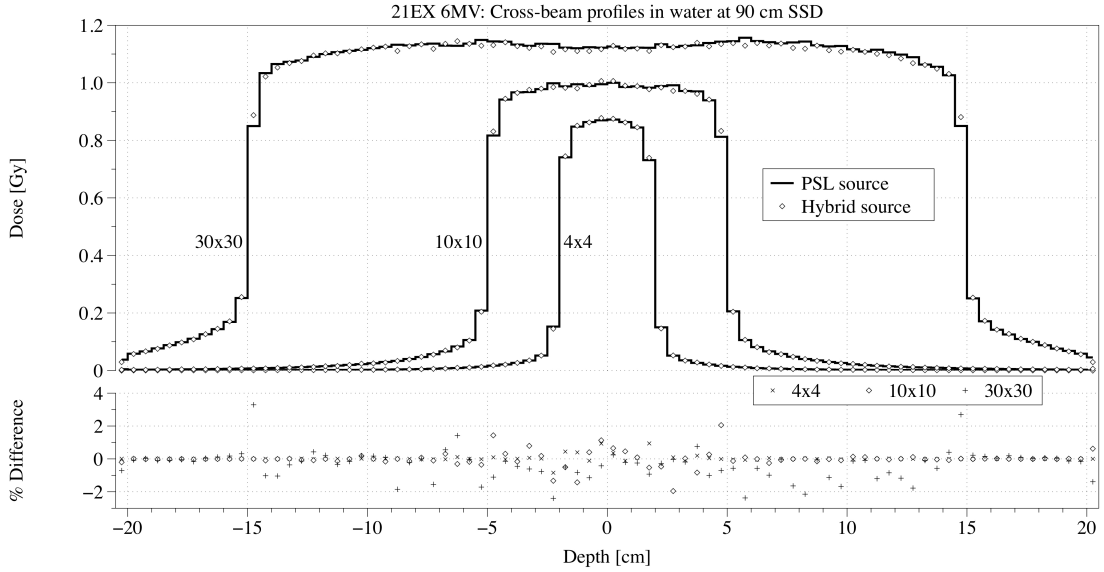

Figure 5.5: Cross-beam profiles for the 21EX 6MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

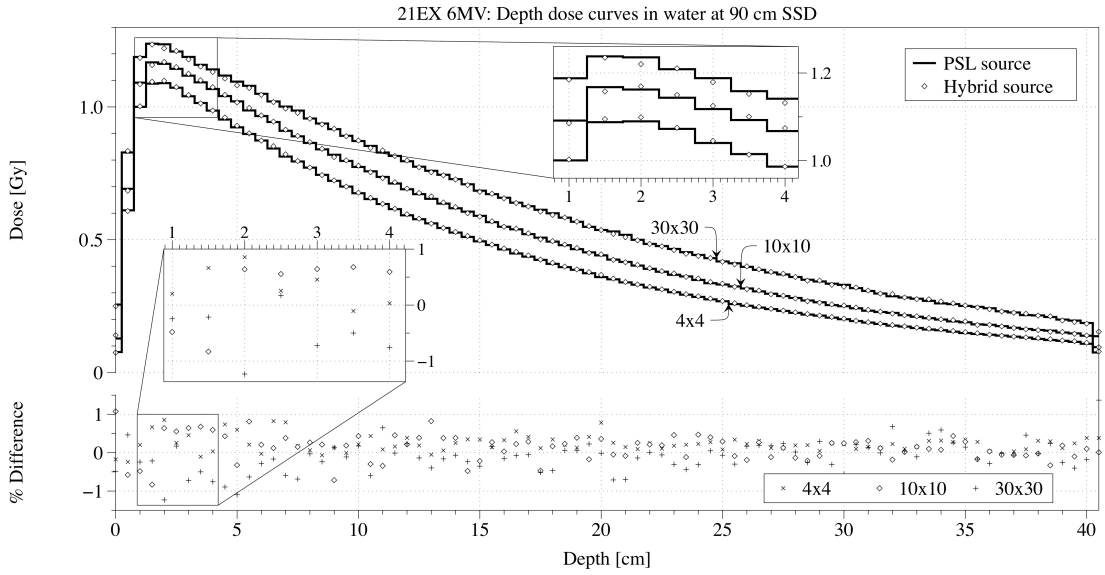

Figure 5.6: Depth dose curves for the 21EX 6MV are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

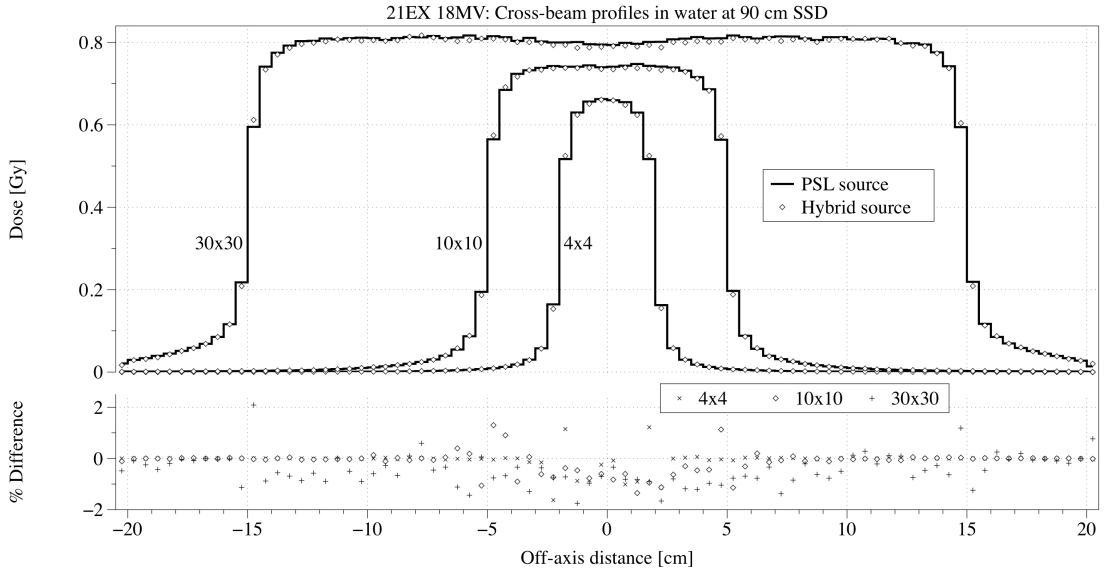

Figure 5.7: Cross-beam profiles for the 21EX 18MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

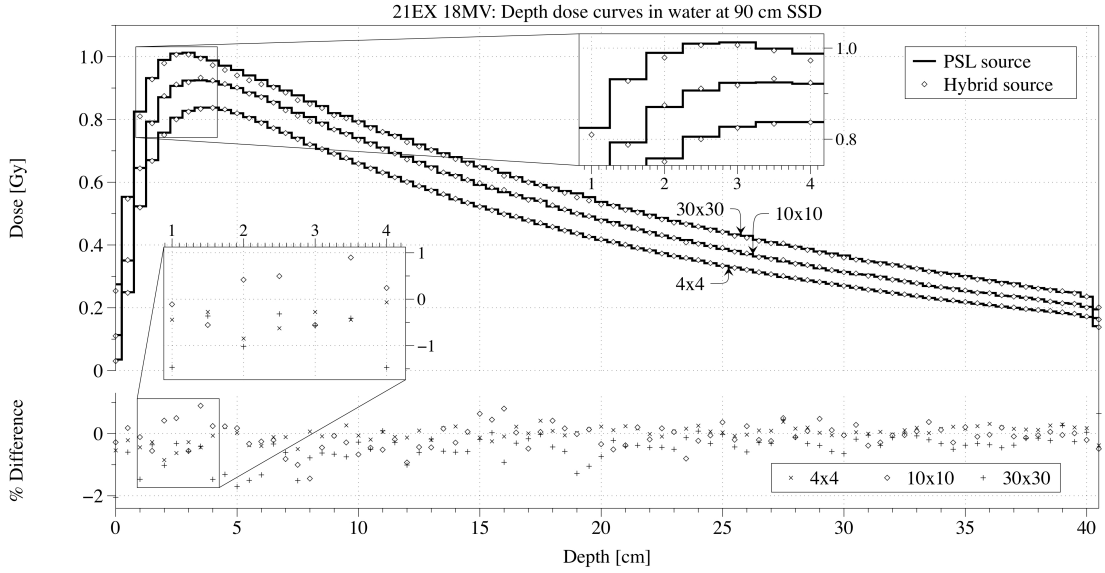

Figure 5.8: Depth dose curves for the 21EX 18MV are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

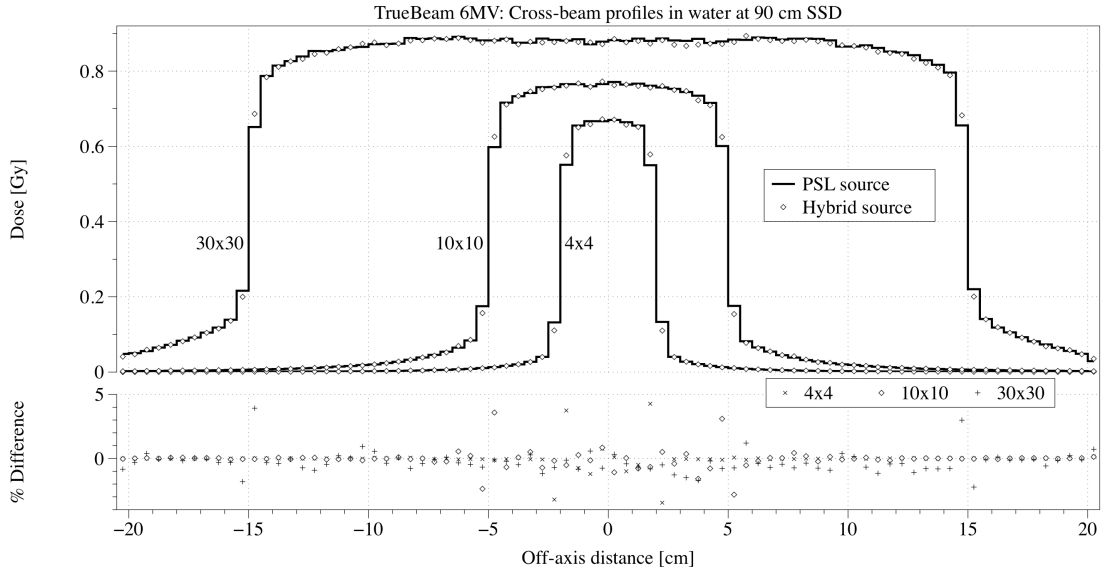

Figure 5.9: Cross-beam profiles for the TrueBeam 6MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

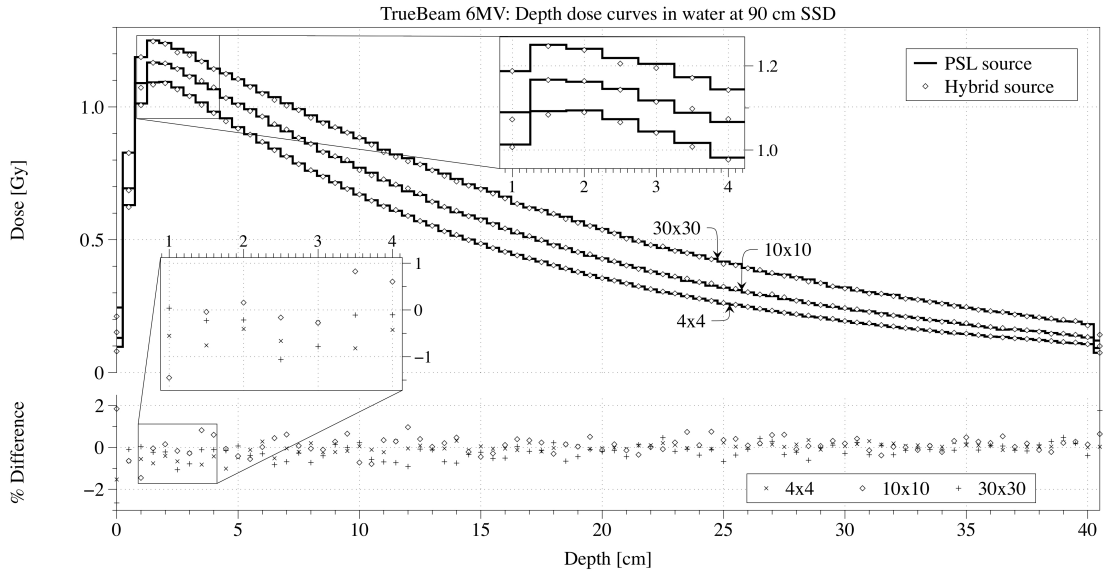

Figure 5.10: Depth dose curves for the TrueBeam 6MV are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

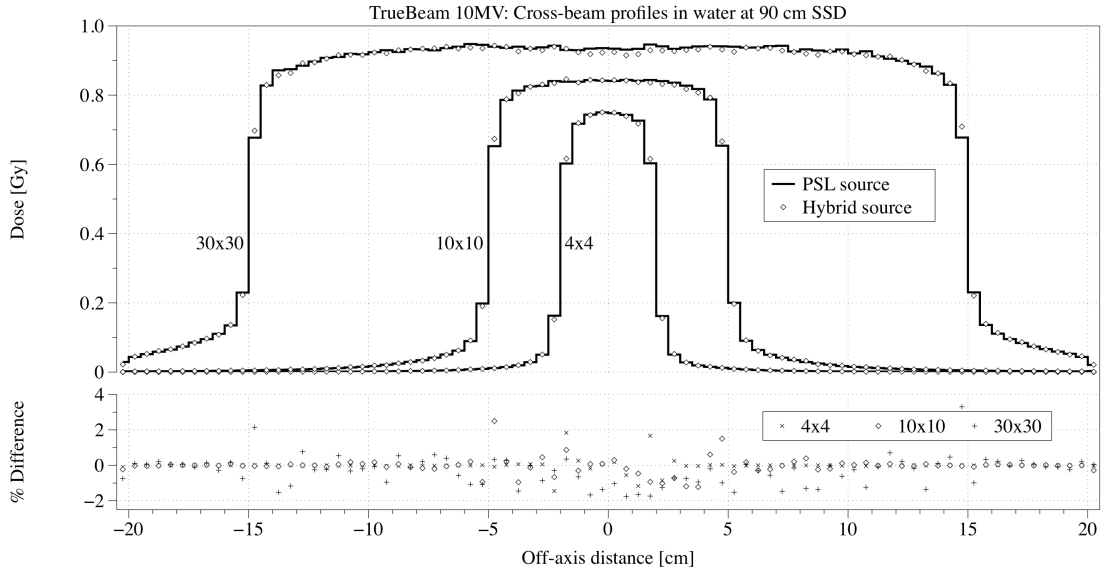

Figure 5.11: Cross-beam profiles for the TrueBeam 10MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

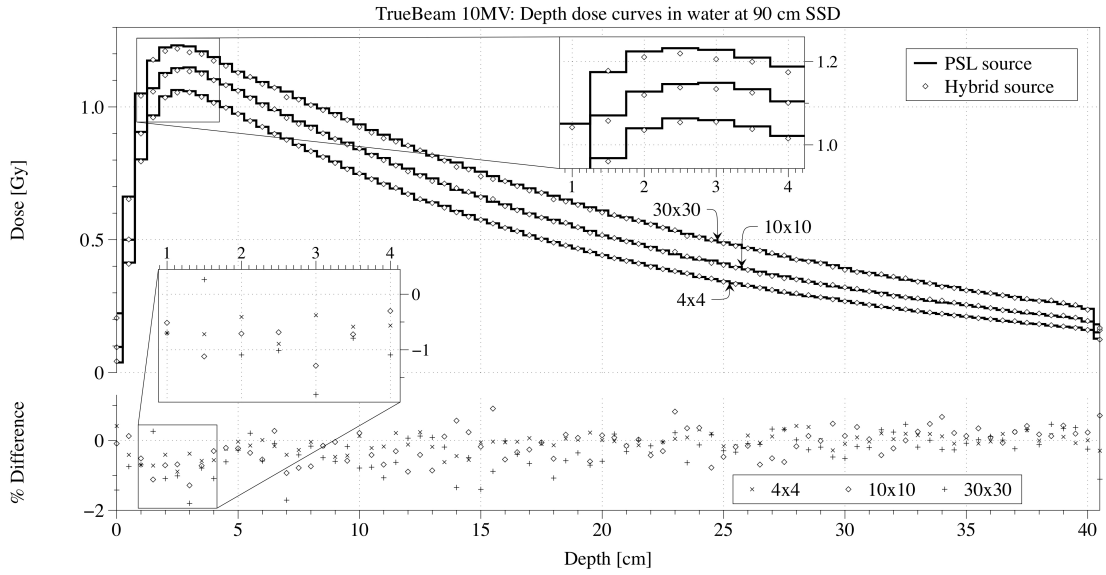

Figure 5.12: Depth dose curves for the TrueBeam 10MV are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

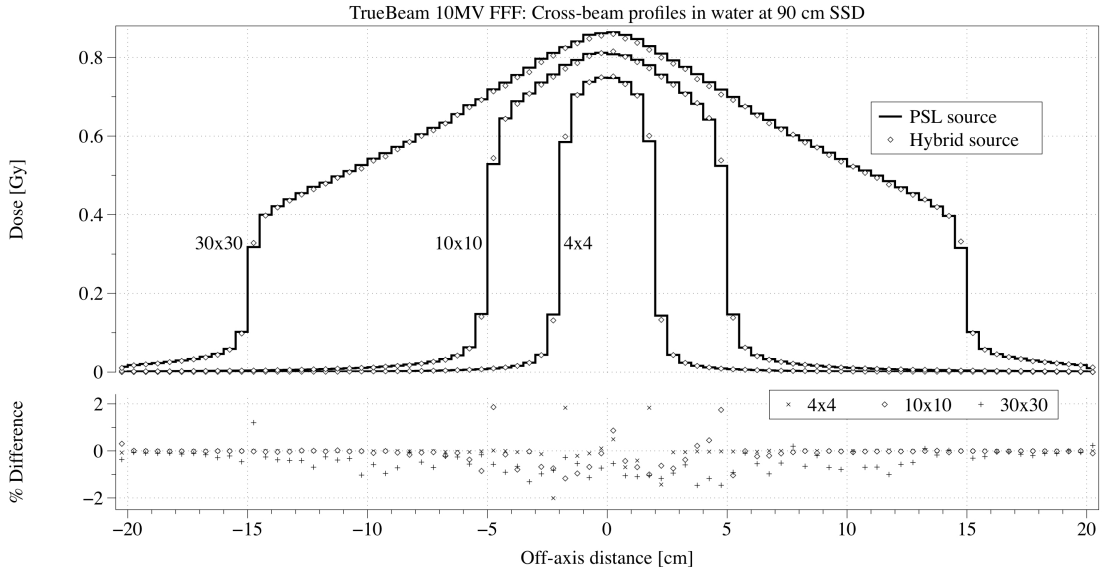

Figure 5.13: Cross-beam profiles for the TrueBeam 10MV FFF at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

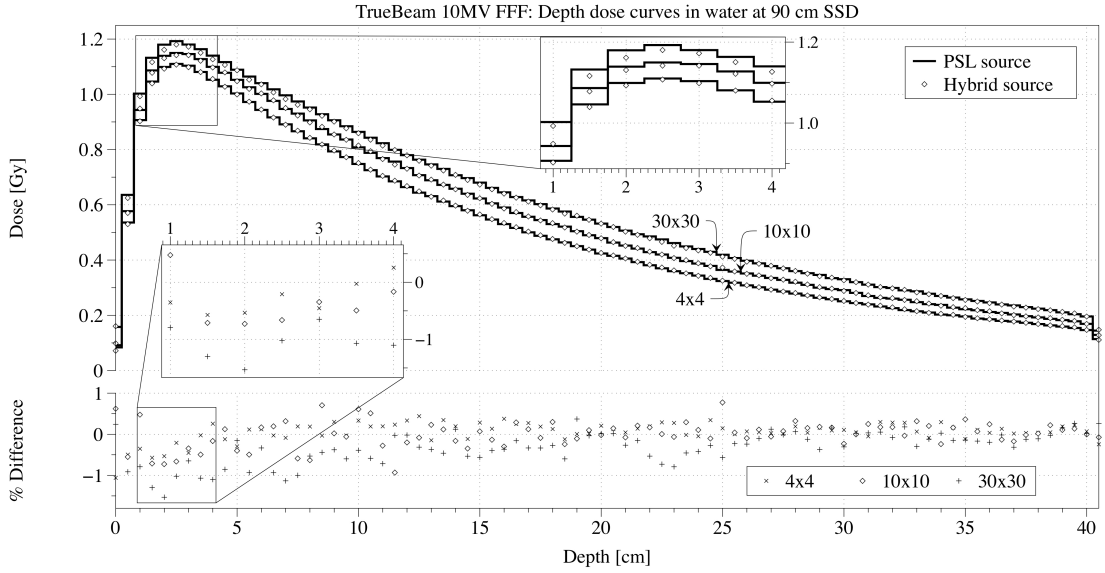

Figure 5.14: Depth dose curves for the TrueBeam 10MV FFF are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

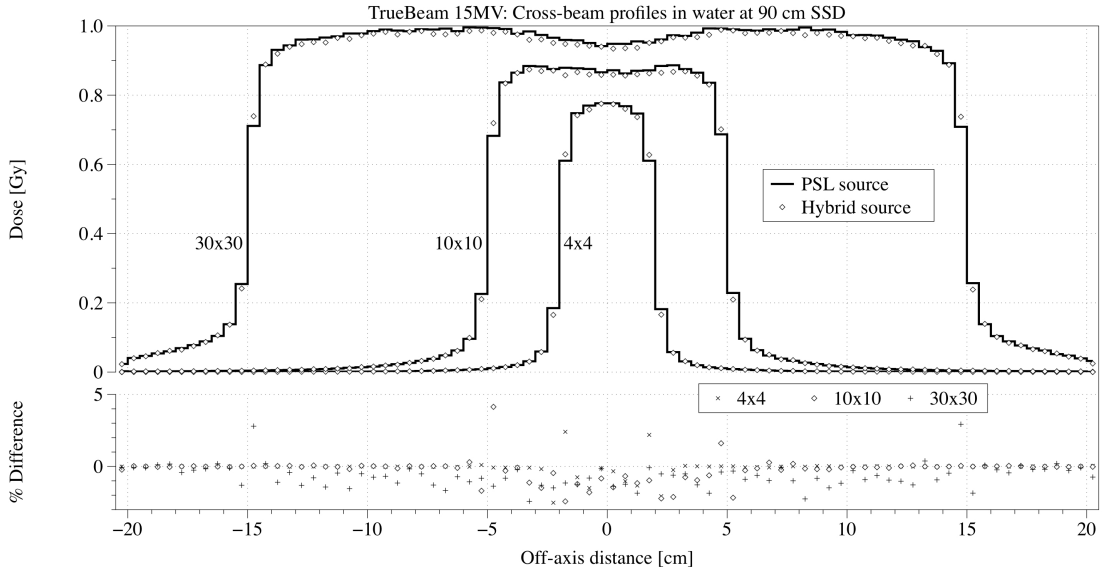

Figure 5.15: Cross-beam profiles for the TrueBeam 15MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

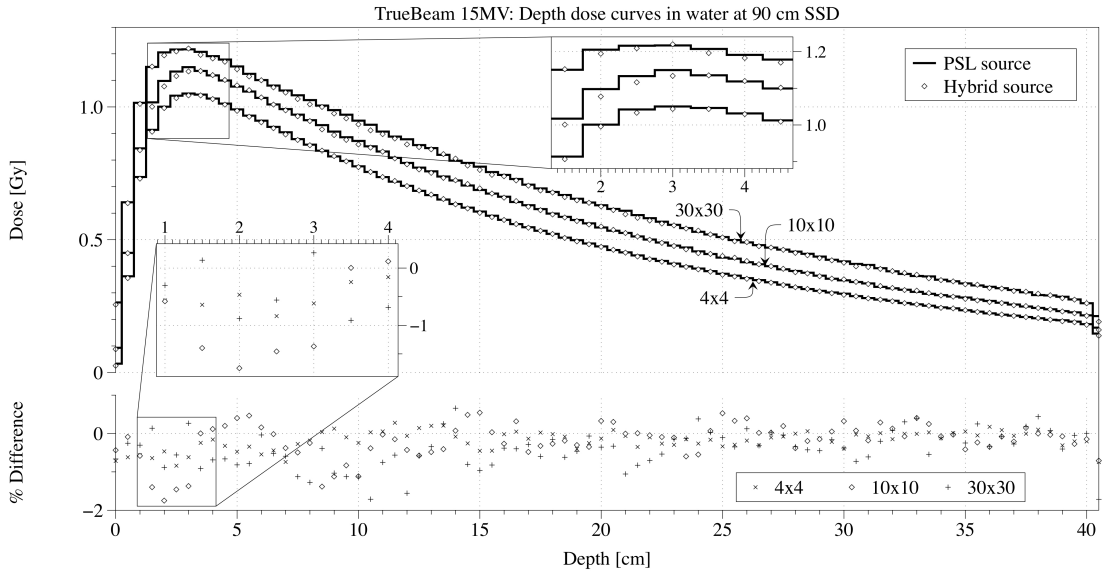

Figure 5.16: Depth dose curves for the TrueBeam 15MV are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

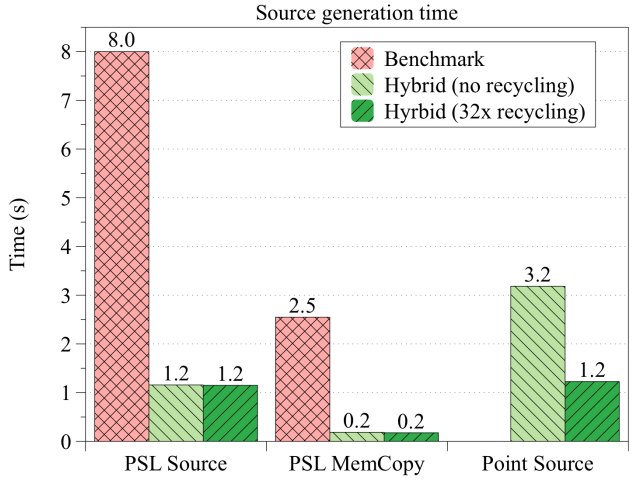

Figure 5.17: A breakdown of the source generation time for a 4×4 cm^2 open field using the 21EX 6MV model. The 'PSL Source' time refers to the time spent reading and processing particles from the PSL source, while 'PSL MemCopy' is the time needed to transfer that data to the GPU. The 'Point Source' time refers to photon generation from the point source component of the hybrid model, which occurs on the GPU (so there is no memory transfer needed).

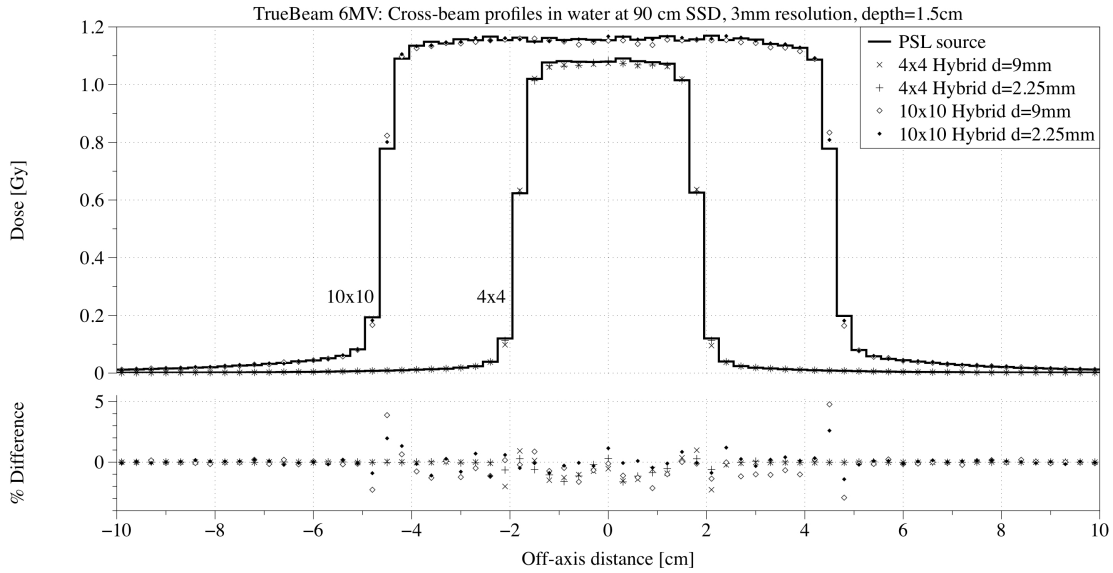

Figure 5.18: Cross-beam profiles for the TrueBeam 6MV at 1.5 cm depth are shown for the hybrid source model (points) and a PSL source (lines) derived from the same initial phase-space. Two choices for d are shown, to illustrate the change in agreement in the penumbra region. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

dose differences were observed in the penumbra regions of the open fields. For these simulations, the hybrid model source generation time was faster than a PSL source by factor of 2-3 for the same number of generated particles. However, this comparison is unfair, since the focal source component of the hybrid source did not use any recycling. In the PSL source, recycling is determined automatically and independently per PSL (ranging from 22 to 68 in these tests). When recycling was set to 32 in the focal source component, the source generation speed-up factor became 4-5. Approximately half of the source generation time was spent generating extra-focal particles, but increasing U could be used to mitigate this. A breakdown of source generation times are shown for the 4×4 cm^2 21EX 6MV case in figure 5.17.

The differences in the penumbra regions were investigated further using a higher resolution phantom and varying the parameter d. As shown in figure 5.18, the agreement in the penumbra region for smaller values of d is improved. This is expected not only because a smaller circular area has been collapsed to a point in the source model, but also because the point source now comprises a smaller component of the total hybrid source model. In this case, the number of annuli in the histograms was increased to $N^{annuli} = 400$, for reasons explained in the next section.

There was also a discrepancy for large fields near the depth of maximum dose on the beam axis, seen in the 30×30 cm^2 fields for 10MV and 10MV-FFF accelerators. The trend was for the hybrid source to underestimate the dose near the beam centre when compared with the PSL source. The systematic error, which depends on the source composition, would converge on the PSL model upon reduction of d. Smaller focal source bin sizes in energy and radius would also improve agreement. However, evaluation of these adjustments was left for future work.

A clinical 7-field IMRT plan using a 6MV Varian Clinac 21EX beam was calculated on a virtual CT phantom with $3 \times 3 \times 3$ mm^3 voxel size, and $N^{annuli} = 400$. Simulations for both the hybrid source model and PSL benchmark were run with 2 billion particles. Focal photon recycling was set to 32. This case achieved 95% χ-test agreement above the 10% isodose with 1% / 1 mm criteria, and 99.8% agreement for 2% / 2 mm. The RMSDs were 0.8%. The source generation time of the hybrid source was a factor of 2.5 faster than the PSL source method.

5.4 Accuracy and efficiency implications of varying N^{annuli}

The radial resolution of the histograms, determined by the parameter N^{annuli}, plays an important role in the accuracy and efficiency of the results. In particular, as N^{annuli} is decreased, an artefact appears near the beam axis (figure 5.19). Using a water phantom with $3 \times 3 \times 3$ mm^3 voxel size, open fields were used to demonstrate this effect for N^{annuli} ranging from 100 to 400. The artefact is likely due the uniform sampling of radius within a selected radial bin in algorithm 2. As the radial bins get larger, this approximation of uniform photon density and constant energy spectrum within an annulus is no longer valid. This has a particularly large effect on the beam-axis, observed as a peak at the centre of the fields in figure 5.19.

The larger values of N^{annuli} also appear to reduce the efficiency of the simulations by a small amount, as shown in figure 5.20 for a 10×10 cm^2 open field. Both the photon and electron transport times increased with N^{annuli}, though source generation time stayed roughly the same. While this effect has not been investigated in detail, the increased spatial resolution leads to improved conformity of the outermost annulus (given by i_b^{max}) with the field size. This means that less particles are generated that will subsequently be absorbed be secondary collimators, some of which get passed along to transport despite carrying zero weight. When the conformity is reduced by decreasing N^{annuli}, more zero weight particles get included in the transport portion of the simulation (and immediately thrown away), leading to reduced simulation times but proportionately higher statistical uncertainty.

Since a smaller N^{annuli} reduces simulation times, and resolution near the beam axis appears to be a limiting factor on accuracy, it would be an effective solution to use a multi-resolution radial spectrum. That is, adjust the radial bin sizes to be smaller near the beam axis. As observed in figure 5.20, the speed-up between $N^{annuli} = 400$ and $N^{annuli} = 100$ is moderate, making this worth considering for future work.

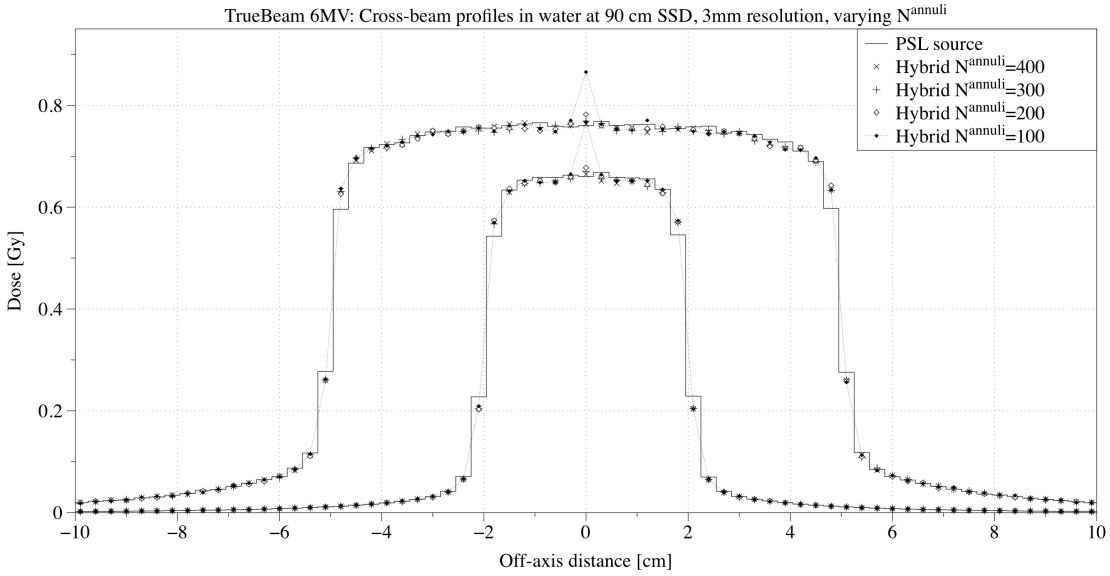

Figure 5.19: Cross-beam profiles for the TrueBeam 6MV at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The spatial resolution of the histograms in the focal source model, N^{annuli}, was varied.

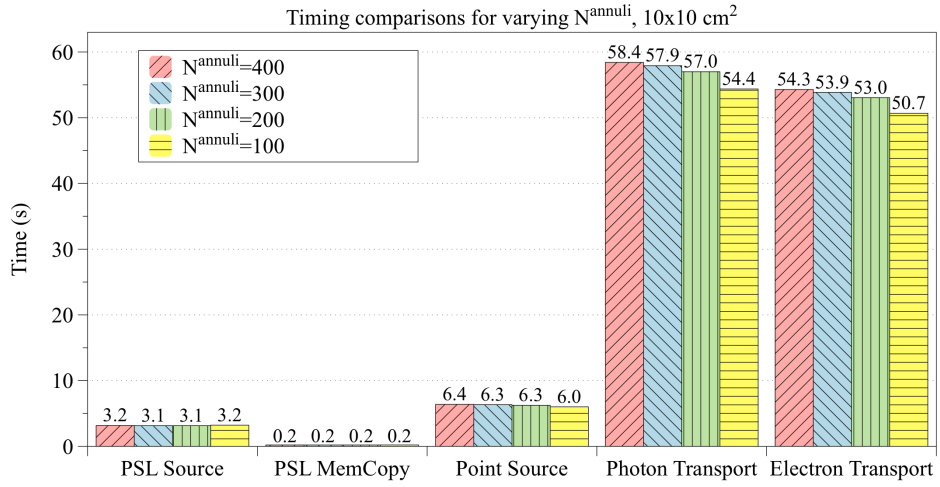

Figure 5.20: The spatial resolution of the histograms in the focal source model, N^{annuli}, was varied. Timing comparisons are shown for a 10×10 cm^2 open field using a water phantom with $3 \times 3 \times 3$ mm^3 voxel size and the TrueBeam 6MV accelerator.

5.5 An alternative focal source generation algorithm

The algorithm described above was the final result of some experimentation with alternative implementations. The next most successful algorithm is presented below for

comparison. Consider the modification where the number of particles to simulate per ij bin, $N_{ij}^{focal/sim}$, is calculated prior to photon generation (instead of being sampled from the histograms using a Metropolis algorithm). Additionally, the beam number b is not sampled on the GPU, but rather particles are distributed evenly between beams (see algorithm 3).

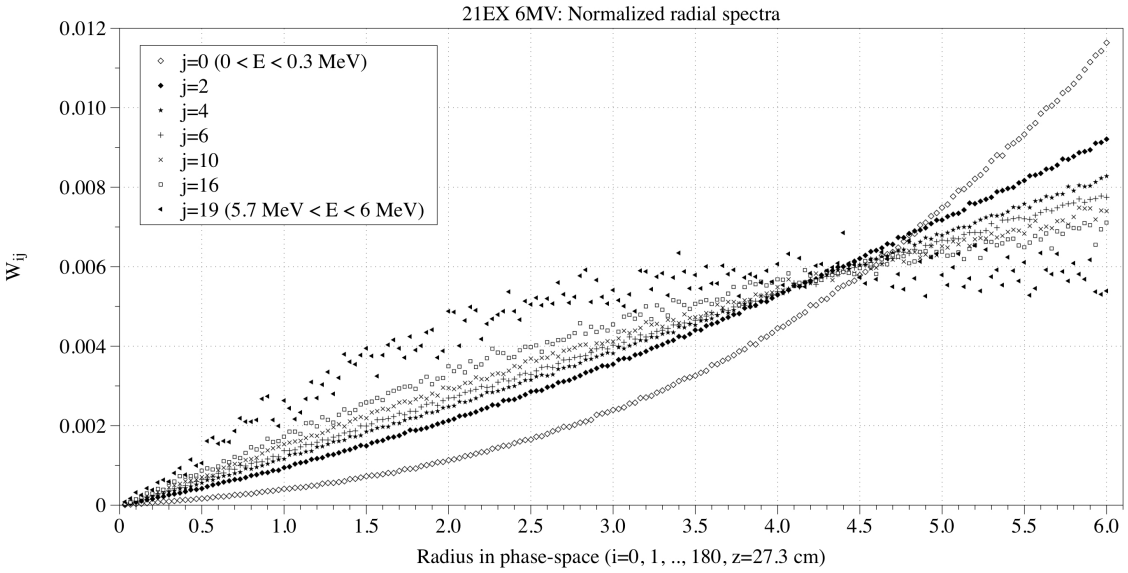

Figure 5.21: Normalized radial distributions of focal photons plotted for a number of energy bins, as derived from a 21EX 6MV phase-space. The source originated from full MC simulation of the linac head model, and was scored just above the secondary collimators. Used for algorithm 3.

For this method, $N_{ij}^{focal/sim}$ is calculated using the number of photons in each ij division of the original phase-space. This is calculated using the curves W_{ij} that are normalized using the number of photons in each energy bin (shown for the Varian 21EX 6MV in figure 5.21).

$$N_{ij}^{focal/sim} = FN^{requested}\frac{W_{ij}}{N^{beams}N^{focal}}, \tag{5.8}$$

where

$$W_{ij} = \frac{N_{ij}^{focal}}{N_{j}^{focal}}. \tag{5.9}$$

Algorithm 3 achieved accuracy effectively identical to algorithm 2. This was tested for the full range of accelerator models, but for brevity only the results for the True-Beam 6MV are shown (figures 5.22 and 5.23). For the same set of open fields and

Algorithm 3 Hybrid Source: alternative focal photon generation

```
1: procedure GenerateFocalPhotons
2:     for each integer j in N^energies do
3:         for each integer i in N^annuli do
4:             Calculate N_ij^focal/sim;
5:             for each integer b in N^beams do
6:                 if i > i_b^max then
7:                     continue to next b;
8:                 end if
9:                 for each batch of photons until all N_ij^focal/sim are simulated do
10:                    for each photon <<< parallelized on GPU >>> do
11:                        Sample radius within bin i uniformly;
12:                        Sample the energy within bin j uniformly;
13:                        Sample the azimuthal angle φ uniformly;
14:                        Calculate direction cosines, assuming a point source;
15:                        Apply secondary collimator and MLC models;
16:                        Store surviving photons in global GPU memory;
17:                    end for
18:                    Transport the batch <<< parallelized on GPU >>>;
19:                end for
20:            end for
21:        end for
22:    end for
23: end procedure
```

accelerators as in section 5.3.3, the mean 1% / 1 mm χ-index test agreement was 99%. The RMSDs were all less than 1%, with a mean RMSD over the open field cases of 0.5%.

Due to efficiency dependencies on the number of batches and beams in algorithm 3, discussed below, the recommended hybrid source implementation is algorithm 2.

5.5.1 Efficiency dependency on $N^{batches}$ and N^{beams}

Thus far in this dissertation, the impact on efficiency of the number of independent batches used in the simulation has not been mentioned. In this context a batch refers to step 3 in figure 2.2, and is essentially a completely independent dose calculation to allow for the determination of statistical uncertainties over the batches. This dependency is different for each of the source models discussed, and in the case of algorithm 3, also includes a dependency on the number of beams, N^{beams}.

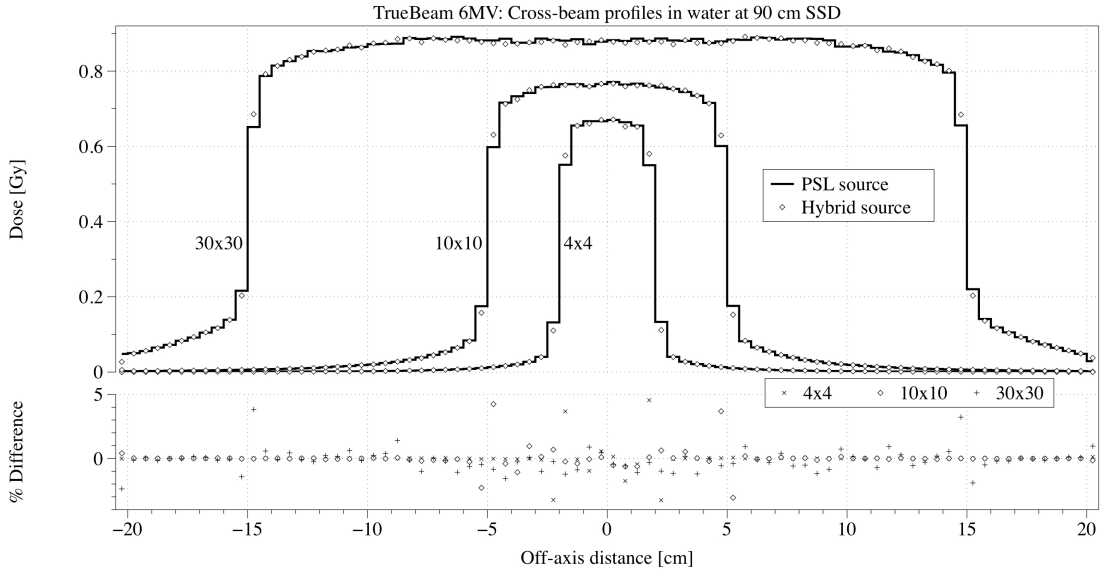

Figure 5.22: Using algorithm 3 for the TrueBeam 6MV accelerator, cross-beam profiles at 10 cm depth are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

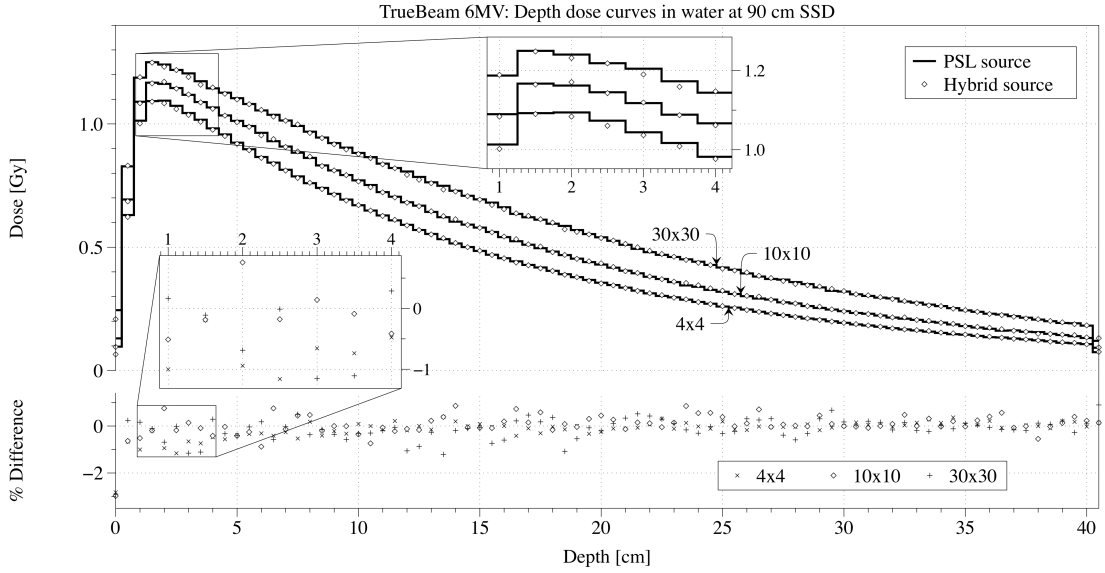

Figure 5.23: Using algorithm 3 for the TrueBeam 6MV accelerator, depth dose curves are shown for the hybrid source model (dots) and a PSL source (lines) derived from the same initial phase-space. The percentage differences are also shown, relative to the maximum PSL source dose in the curve.

For the PSL method, the source generation time strongly depends on the number of batches, but particle transport times do not change. This is because the function for PSL source generation has a relatively high computational cost each time it is called, even when a smaller number of particles are being generated.

With the hybrid source algorithm 2, the source generation time does not change when $N^{batches}$ is increased - the overhead cost of calling this function is much lower. Additionally, the particle transport times remain similar.

With algorithm 3, the source generation time increases only slightly with increased $N^{batches}$, but the electron transport times increase substantially, particularly when N^{beams} is also increased. This is due to the number of particles simulated at once, $N_{ij}^{focal/sim}$, tending to be small, resulting in low GPU hardware utilization during transport of secondary charged particles. Recall that $N_{ij}^{focal/sim}$ depends on N^{beams}. This efficiency dependency is a critical drawback, since some treatment techniques such as VMAT use a large number of beams.

The resulting simulation times for changing $N^{batches}$ from 1 to 10 are shown in figure 5.24. No recycling was used for the hybrid sources in this figure (because it has not yet been implemented for algorithm 3), while automatic recycling was used for the PSL sources. It is worth noting the reduced transport times of algorithm 3 compared to algorithm 2 for $N^{batches} = 1$. This is likely due to reduced thread divergence resulting from all photons in $N_{ij}^{focal/sim}$ being simulated simultaneously for the same beam b. Since the photons enter the surface of the phantom from the same general direction given by the beam angle, they are more likely to follow similar interaction tracks, compared to the other methods where the beam number is sampled uniformly for each particle. This hypothesis was not tested and may be worth investigating in future work. It is worth noting that the long calculation time for electron transport for algorithm 3 where $N^{batches} = 10$ in figure 5.24 is not a result of an overall increased number of secondary electrons being produced. This quantity is tracked in the software - the number of both photons and electrons actually transported in the phantom was very similar between all methods.

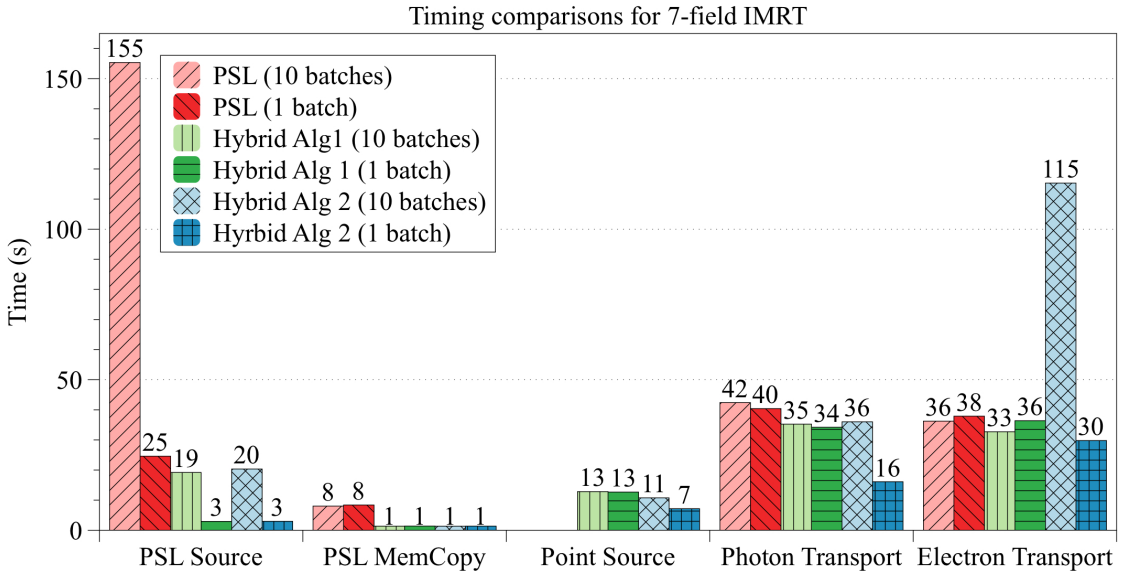

Figure 5.24: A breakdown of the simulation times for the PSL method and two hybrid source implementations (algorithms 2 and 3. These times are for a 7-field IMRT case - the same plan as in figure 4.10. No recycling was used for the hybrid sources, and automatic recycling was used for the PSL method.

5.6 Discussion and conclusions

The proposed hybrid source model has two major benefits compared to using a standard phase-space or PSL source for GPU-based MC calculations: (1) faster calculation time compared to the PSL method, and (2) roughly a six-fold reduction in the size of the required phase-space. As shown, these improvements have been achieved while maintaining a good level of accuracy.

It is helpful to note that, in our simulations, the precise choice of diameter d (for selecting focal spot photons) mostly affected accuracy in the penumbra regions and was not critical in obtaining generally accurate results. The trade-off for increasing d was improved performance (due to a greater portion of the original phase-space being including in the point source model), against introducing bias into the particle distribution due to collapsing a larger source area into a point. The maximum value of d that should be used depends on the application and source in question. Regardless, for sufficiently small d, the ability of the ray-tracing method to precisely select true focal spot photons is not essential because inclusion of contamination particles does not reduce its accuracy. Rather, it is the ability of the method to extract a large portion of a phase-space that can be well modeled by a simple source that is important.

Our results indicate that the bias introduced by the point source, which mostly affects the out-of-field regions, is dosimetrically small for the accelerator models considered. Improvements to the model in the form of a non-point source distribution would help to reduce errors, but would require accounting for position/direction correlations.

The statistical uncertainty of extra-focal contributions in the hybrid source model can be tuned independently of the focal component using the parameter U. This is particularly useful to reduce calculation times in cases where these contributions are less significant; for example, when the desired result is only a point dose at depth along the central axis.

The accuracy of the hybrid source method is comparable with previously reported models in the literature. The phase-space derived multiple source model described by Fix *et al.* (2004 [24]) included target, primary collimator, flattening filter, and charged particle sources. This achieved agreement of 1% or 1 mm for 99% of the dose voxels (compared to MC simulation with the full phase-space) for a series of open field cases using a Varian Clinac 21EX 6 and 18 MV. Fippel *et al.* (2003 [23]) used Gaussian-shaped target and flattening filter sources combined with a uniform electron source, and additional fluence variation parameters to model the treatment head, based on standard measurements in water and air. Output factors for a range of field sizes were compared in this study, showing percent difference with measurement of better than 2% in most cases for Elekta (6 and 15 MV) and Siemens (6 and 10 MV) machines.

The current implementation of the hybrid source model is most efficient for square fields centred on the beam axis. Since photons are generated within a circle centred on the beam axis (the radius determined by collimator positions), off-axis and asymmetric fields result in generating a greater number of photons that will subsequently be absorbed by the collimators. Modifying the algorithm to generate photons within a cylindrical sector that is centred on the field would mitigate this. In contrast, the PSL source method uses rectangular spatial discretization of the PSLs to select only those containing particles likely to pass through the collimator openings and therefore is well adapted for use with asymmetric and off-axis fields.

The focal source model presented has the advantage of being able to efficiently produce photons that are statistically independent, unlike recycling particles in a phase-space. For a sufficiently large initial phase-space that accurately captures the energy and angular distributions of photons (and when no recycling is used), the focal source component can be modeled accurately with negligible error and no latent vari-

ance. The extra-focal component that is modeled by a small PSL database will still maintain its latent variance, though its contribution is considerably reduced compared to that of the full phase-space, and it will mostly affect the dose uncertainty outside of the secondary collimator aperture. Additionally, provided an accurate phase-space source, the hybrid model requires no further tuning, in contrast to analytical source models. This was shown in our tests by using identical parameters for a variety of beam models.

Chapter 6

A GPU-based implementation of photon splitting

In this chapter the photon splitting variance reduction techniques (VRTs) will be investigated, to the aim of decreasing dose calculation times in GPU-based engines. The photon splitting technique presented below is similar to photon recycling, but is repeated at each transport step and involves spatial redistribution of the particles. While VRTs of this sort have been previously designed for CPU-based MC software (Kawrakow and Fippel, 2001 [41]), little work has been done with GPU-based codes. Since algorithm design plays such a important role in the efficiency of GPU codes, novel solutions may be required. In the work that follows, the speed-up of one photon splitting implementation in gDPM v3.0 was determined, and the experience was used to suggest an alternative algorithm.

6.1 Photon splitting and redistribution

Photon splitting is designed to increase the number of photons interacting in the patient phantom, reducing the number of photons required to be incident on the phantom to achieve a particular statistical uncertainty. At the surface of the phantom, each source particle is replaced by N^{split} "sub-photons", each weighted by $1/N^{split}$. Initially, each sub-photon contains identical properties. Then, to achieve a more uniform distribution of interaction sites (Kawrakow and Fippel, 2001 [41]), the number

of mean-free-paths (MFPs), ν_i, for the i'th sub-photon is calculated by,

$$\nu_i = -\ln\left(1 - \frac{r+i}{N^{split}}\right), \tag{6.1}$$

where r is a uniformly distributed quasi-random number ($0 \leq r \leq 1$), and $0 \leq i \leq N^{split} - 1$.

Recall that gDPM v3.0 uses the Woodcock tracking VRT for determining photon interaction sites, as illustrated by the pseudocode in algorithm 4. In the Woodcock tracking algorithm, photons are transported using the maximum cross section in the geometry. This introduces additional interactions, called fictitious interactions. Normally, when a fictitious interaction takes place, the photon direction and energy are left unchanged and the photon continues to the next transport step. In the context of photon splitting, each of the sub-photon interaction sites has a chance to be a fictitious interaction. Sub-photons that undergo fictitious interactions have their directions and energy left unchanged, and are stored for later transport (similar to Compton scattered photons). This is different from photon splitting by Kawrakow and Fippel, where photon transport was performed without Woodcock tracking. Pseudocode for photon splitting in gDPM v3.0 is provided in algorithm 5.

After determining the photon interaction sites for all sub-photons, the interaction types are sampled (Compton, pair production or photo-absorption). For a given set of sub-photons, the event parameters of each interaction type are calculated at most once and then re-used. However, since the sub-photons have different interaction locations, some of the sub-photons may reside in different materials. In order to re-use event data in these cases, the assumption is made that the differential cross sections for all photon processes are close to independent from the material. This is generally true at radiotherapy energies (Kawrakow and Fippel, 2001 [41]). There are two beneficial results of this assumption: (1) reduced computation time and (2) for a given interaction type, the electrons produced will all have the same energy (with different positions and directions).

Scattered photons are terminated using Russian Roulette with probability $1/N^{split}$, and the weight of surviving photons is multiplied by N^{split}. This restores the original weight for scattered photons. All scattered photons and electrons are stored in global GPU memory to be transported in the next batch.

Photon splitting was selected as a worthwhile VRT for investigation for a number of reasons. Firstly, source generation time is a bottleneck in GPU-based dose

calculations, so reduction of the number of independent source particles required is important. Secondly, photon splitting produces secondaries with identical energy, ideal for reduced thread divergence. Finally, photon splitting is necessary for the STOPS VRT, which also lends itself well to the GPU (see section 6.3).

Algorithm 4 Photon Transport

```
1: procedure PhotonTransport
2:     Load particle(x, y, z, u, v, w, E, wt);
3:     if E ≤ 0 or wt ≤ 0 then
4:         Break;
5:     end if
6:     while true do
7:         r =a uniform random number on (0, 1];
8:         Lookup λ_min(E)                          ▷ Minimum mean free path [cm]
9:         s = −λ_min ln(r);        ▷ Pathlength to the next potential interaction [cm]
10:        x = x + su;                          ▷ Transport the photon to a new voxel
11:        y = y + sv;
12:        z = z + sw;
13:        Lookup ρ(voxel);                  ▷ Density in the new voxel [g/cm^3]
14:
15:        r_2 =a uniform random number on (0, 1];
16:        Lookup η_total(E, voxel);          ▷ Total inverse mean free path [cm^2/g]
17:        P = 1 − λ_min ρ η_total;          ▷ The probability of a fictitious interaction
18:        if r_2 < P then
19:            Continue;                          ▷ A fictitious interaction occurs
20:        end if
21:
22:        Lookup η_compton(E, voxel);     ▷ Inverse Compton mean free path [cm^2/g]
23:        P = P + λ_min ρ η_compton;
24:        if r_2 < P then
25:            Simulate Compton interaction;
26:            if Photon is absorbed then
27:                Break;
28:            else
29:                Continue;
30:            end if
31:        end if
32:
33:        Lookup η_pair(E, voxel); ▷ Inverse pair production mean free path [cm^2/g]
34:        P = P + λ_min ρ η_pair;
35:        if r_2 < P then
36:            Simulate pair production;
37:            Break;
38:        end if
39:
40:        Simulate photoelectric absorption;
41:        Break;
42:    end while
43: end procedure
```

Algorithm 5 Photon Transport with Splitting

procedure PHOTONTRANSPORTSPLIT
  Load particle$(x, y, z, u, v, w, E, wt)$;
  if $E \leq 0$ or $wt \leq 0$ **then**
    Break;
  end if
  $wt = wt/N^{split}$;
  $r =$a uniform random number on $(0, 1]$;
  Lookup $\lambda_{min}(E)$ ▷ Minimum mean free path [cm]
  for Each i in N^{split} **do**
    $s = -\lambda_{min} \ln(1 - \frac{r+i}{N^{split}})$; ▷ Pathlength to the next potential interaction [cm]
    $x_i = x + su$; ▷ Transport the photon to a new voxel
    $y_i = y + sv$;
    $z_i = z + sw$;
    Lookup $\rho_i(\text{voxel}_i)$; ▷ Density in the new voxel [g/cm^3]
    $u_i = u$;
    $v_i = v$;
    $w_i = w$;
    $E_i = E$;

    $r_i =$a uniform random number on $(0, 1]$;
    Lookup $\eta_{i,total}(E, \text{voxel}_i)$; ▷ Total inverse mean free path [cm^2/g]
    $P_i = 1 - \lambda_{min}\rho_i\eta_{i,total}$; ▷ The probability of a fictitious interaction
    if $r_i < P_i$ **then**
      stackPhoton$_i$ = true; ▷ The interaction was fictitious and the photon will be stored for later
      Continue;
    end if
  end for

  for Each i in N^{split} **do**
    if stackPhoton$_i$ == true **then**
      Continue;
    end if
    Lookup $\eta_{i,compton}(E_i, \text{voxel}_i)$; ▷ Inverse Compton mean free path [cm^2/g]
    $P_i = P_i + \lambda_{min}\rho_i\eta_{i,compton}$;
    if $r_i < P_i$ **then**
      Simulate Compton interaction;
      if Photon is absorbed **then**
        Continue;
      else
        stackPhoton$_i$ = true; ▷ The Compton scattered photon will be stored for later
        Continue;
      end if
    end if

```
44:
45:         Lookup η_{i,pair}(E_i, voxel_i);          ▷ Inverse pair production mean free path [cm^2/g]
46:         P_i = P_i + λ_{min} ρ_i η_{i,pair};
47:         if r_i < P_i then
48:             Simulate pair production;
49:             Continue;
50:         end if
51:
52:         Simulate photoelectric absorption;
53:     end for
54:
55:     m = 1/N^{split};                    ▷ The rejection parameter for Russian Roulette
56:     for Each i in N^{split} do
57:         r_{i,rej} =a uniform random number on (0, 1];
58:         if stackPhoton_i == true AND r_{i,rej} < m then
59:             wt_i = wt_i N^{split};      ▷ Increase weight to account for Russian Roulette
60:             Add photon_i to the stack, to be processed in the next batch;
61:         end if
62:     end for
63: end procedure
```

6.2 Results

6.2.1 Open field comparisons with photon splitting

The validation experiments for the photon splitting method include comparisons using the same PSL source model with either photon splitting turned on, or using the version of gDPM v3.0 before photon splitting was introduced. For all of the photon splitting test cases, N^{split} was set to 20. In order to maintain a similar level of statistical uncertainty as in the benchmark cases, the number of source particles was reduced by a factor of N^{split}. The number of particles was chosen in all cases to provide an estimated statistical uncertainty of less than 1%.

The initial validation was performed using open fields in homogeneous virtual water phantoms. First, the TrueBeam 6MV PSL model was tested using a water phantom with $133 \times 133 \times 82$ voxels of $3 \times 3 \times 5$ mm^3 resolution, positioned at 90 cm SSD. For 4×4 and 10×10 cm^2 open fields, 2.9 billion and 4.3 billion particles were simulated in the benchmark with no photon splitting. In the $N^{split} = 20$ case, the number of particles simulated was a factor of 20 less. These results are shown as cross-beam profiles at 1.5 cm depth and depth dose curves in figures 6.1 and 6.2, respectively. Additionally, the out-of-field regions were highlighted in figure 6.3. As shown in the figures, the agreement was very good and within the estimated statistical uncertainty.

A higher energy beam was also tested, the TrueBeam 10MV FFF using a homogenous water phantom comprised of $82 \times 82 \times 82$ voxels with 5 mm voxel resolution at 90 cm SSD. For 4×4, 10×10 and 30×30 cm^2 open fields, 1.45 billion, 2.15 billion and 16.7 billion particles were simulated in the benchmark cases, respectively. With $N^{split} = 20$, 20$\times$ less particles were simulated. Cross-beam profiles at 10 cm depth and depth dose curves are shown in figures 6.4 and 6.5. Again, the agreement is within the estimated statistical uncertainty.

A more challenging heterogeneous phantom was also used. Since material boundaries present a challenge for accurate dose calculation, these regions are ideal for testing a new transport algorithm. The phantom was $82 \times 328 \times 82$ voxels with $5 \times 1.25 \times 5$ mm^3 resolution. Slabs of different materials were used, varying along the y-axis. The phantom consisted of slabs of 2 cm of tissue, 3 cm of bone, 7 cm of lung and 29 cm of tissue, as shown in figure 6.6. Two 10×10 cm^2 fields were tested, one in the standard orientation with gantry = couch = collimator = 0°, and

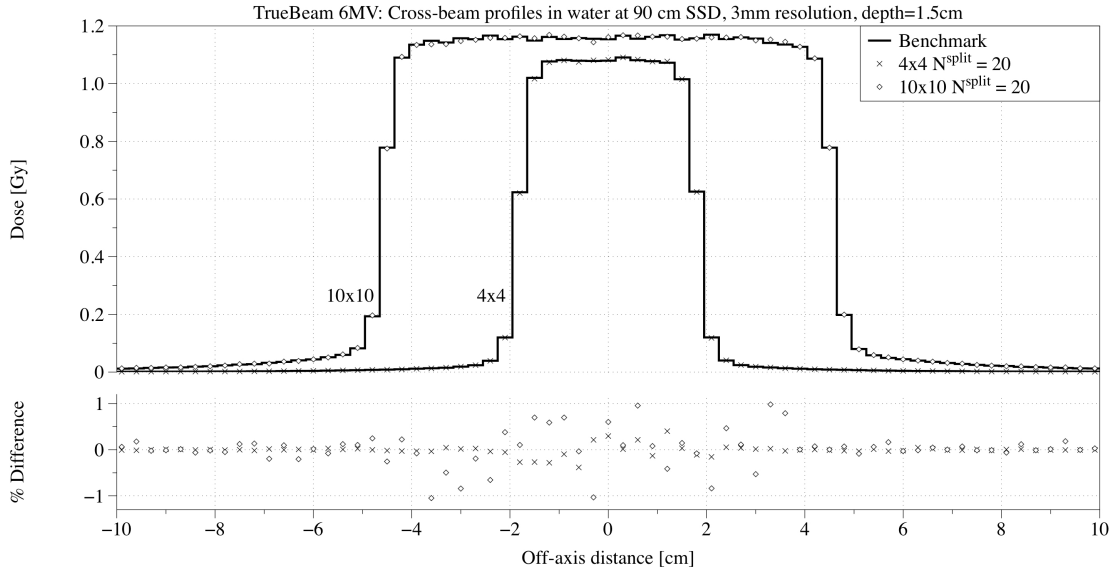

Figure 6.1: For the TrueBeam 6MV accelerator: cross-beam profiles at 1.5 cm depth are shown for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines) using the same PSL source model. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

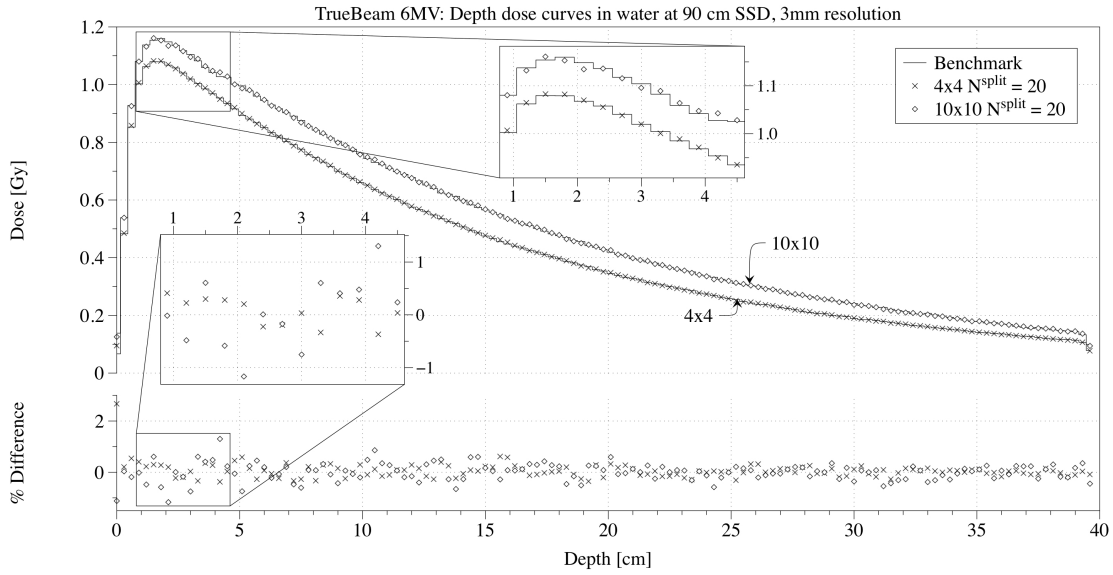

Figure 6.2: For the TrueBeam 6MV accelerator: depth dose curves are shown for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines), using the same PSL source. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

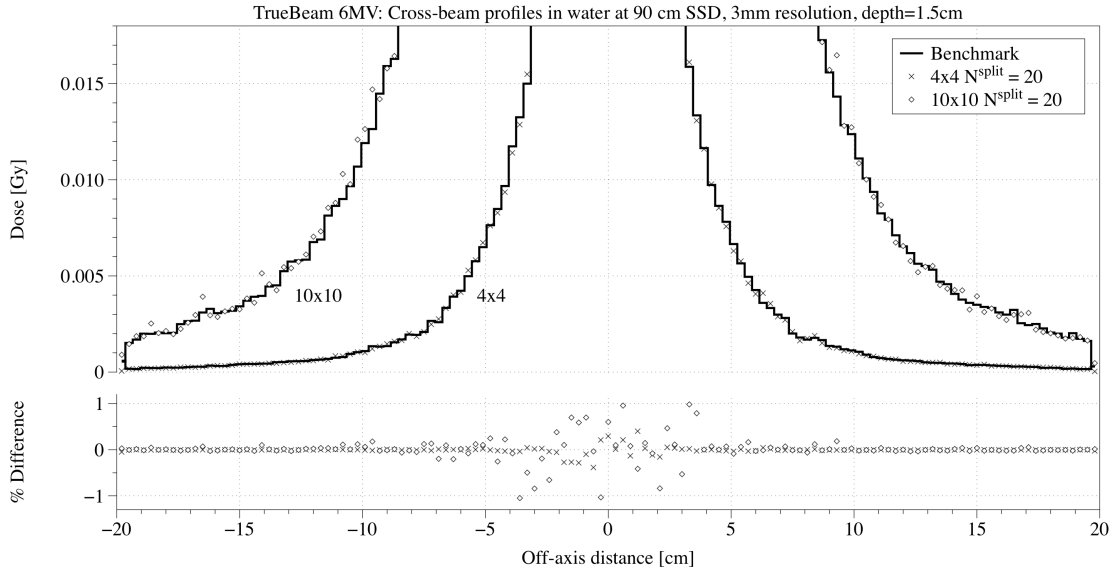

Figure 6.3: For the TrueBeam 6MV accelerator: cross-beam profiles at 1.5 cm depth are shown for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines) using the same PSL source model. The curves have been zoomed in to better visualize the out-of-field regions. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

one with gantry = 90° and the isocentre shifted to $y = -15$ cm. Dose curves are shown in figure 6.7 for both set-ups, oriented along the y-axis through the centre of the phantom. The γ-index and χ-index test results for gantry= 0° were 99.9% and 99.1%, respectively above the 10% isodose for 1% / 1 mm criteria, with an RMSD of 0.4%. For gantry= 90° and the same criteria, the γ-index and χ-index success was 99.2% and 97.2%, respectively, with an RMSD of 0.6%.

6.2.2 Comparing and combining the hybrid source method with photon splitting

A realistic IMRT tongue treatment plan was used to compare the efficiency and accuracy of the hybrid source method (chapter 5) and photon splitting when implemented separately and concurrently. First, a benchmark simulation was performed using a PSL source of the TrueBeam 6MV accelerator. A large number of particles were simulated for the benchmark, $N^{sim} = 11.3$ billion particles. This was compared with various configurations of photon splitting and the hybrid source model, all using clinically realistic statistical uncertainty requirements of $< 1\%$ near the isocentre,

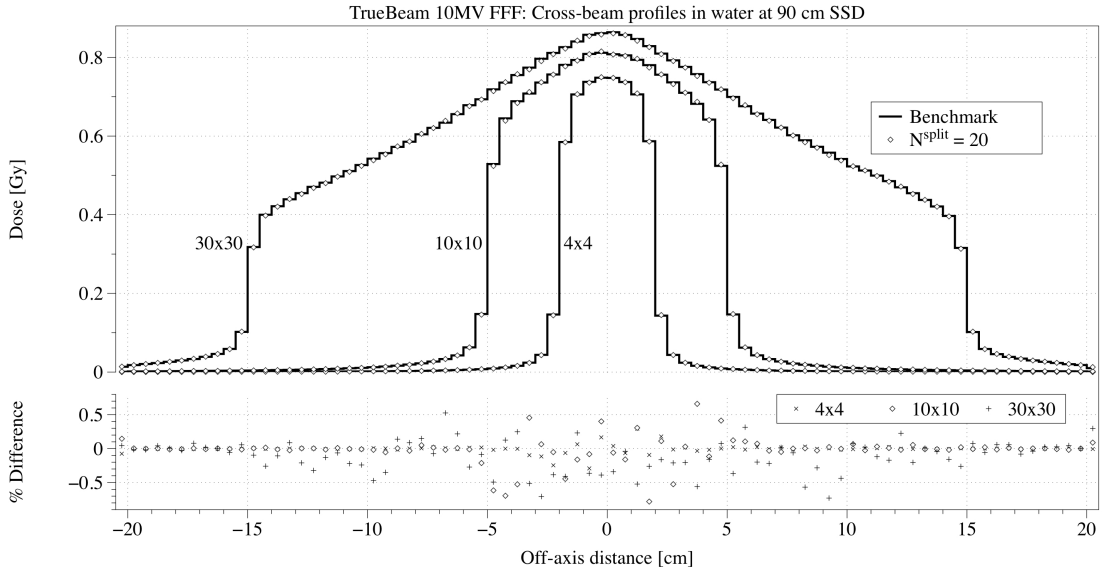

Figure 6.4: For the TrueBeam 10MV FFF accelerator: cross-beam profiles at 10 cm depth are shown for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines) using the same PSL source model. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

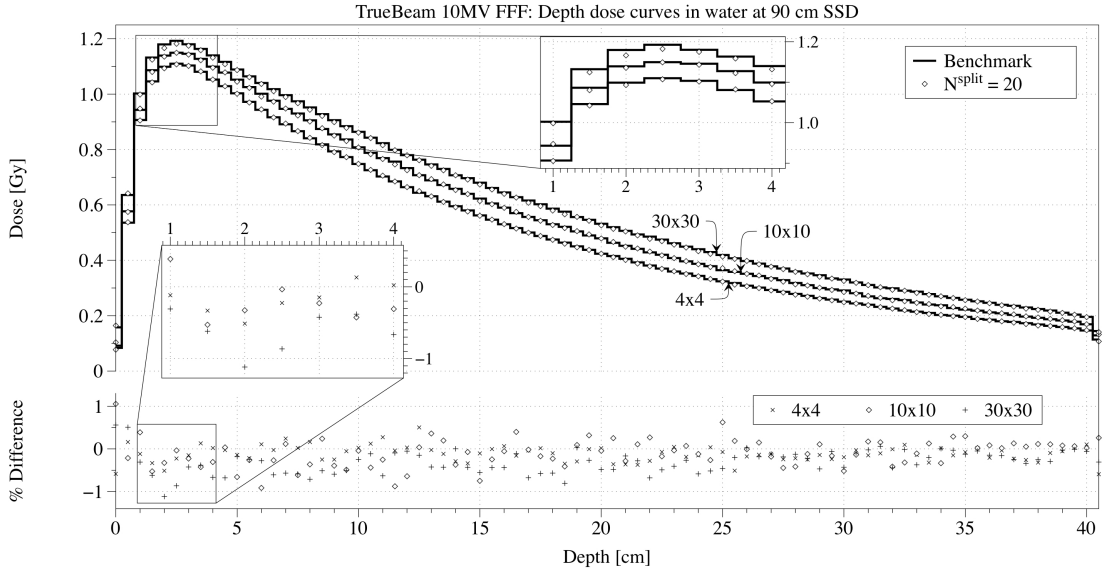

Figure 6.5: For the TrueBeam 10MV FFF accelerator: depth dose curves are shown for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines), using the same PSL source. The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

which corresponds to simulating $N^{sim}/4$ histories. The parameters varied were N^{split} (which requires a change in the number of simulated particles to maintain a similar statistical uncertainty), and $N^{recycling}$ in the hybrid source. The test cases were: (1) the PSL source with $N^{split} = 20$ and $N^{sim}/80$ particles, (2) the hybrid source with $N^{recycle} = 32$ and $N^{sim}/4$ particles, (3) the hybrid source with $N^{recycle} = 32$, $N^{split} = 20$ and $N^{sim}/80$ particles, (4) the hybrid source with $N^{recycle} = 1$, $N^{split} = 20$ and $N^{sim}/80$. Finally, the original PSL source, as in the benchmark, was simulated with $N^{sim}/4$ particles to illustrate the expected agreement.

The results of γ-index tests, χ-index tests, RMSDs and maximum percent differences are shown in table 6.2.2. In the IMRT patient case tested, the PSL source with $N^{split} = 20$ achieved very similar agreement with the benchmark as the PSL source without photon splitting. As expected, the hybrid source with $N^{recycle} = 32$ was less successful. Combining the hybrid source with a high amount of recycling ($N^{recycle} = 32$) with photon splitting did particularly poorly. This is an effect of excessive recycling - the number of independent source particles is reduced both by recycling and photon splitting (recall that the hybrid source model does not perform APR when photons are recycled). It is therefore more sensible to turn off hybrid source recycling when photon splitting is turned on. Under these conditions, the results improve to match those of the hybrid source alone without photon splitting.

Simulation times for each of these cases are presented in figure 6.8. Notice that with $N^{split} = 20$, the photon transport times increase in all cases. This is due to the increased complexity of the photon transport algorithm. The electron transport times are decreased slightly in the hybrid source and photon splitting cases. This is a result of photon repetition producing secondary electrons of similar energy, reducing thread divergence. The total simulation times were similar and shortest for the hybrid source configurations. For this reason, the recommended configuration for the highest speed while maintaining reasonable accuracy is the hybrid source with recycling turned off, and photon splitting turned on. The best choice of photon splitting N^{split} will be determined in the following section.

6.2.3 Determining the optimal choice of N^{split}

The heterogeneous tissue-bone-lung phantom described in section 6.2.1 was also used to test variations of N^{split} on the efficiency and accuracy of the simulations. All other parameters were held constant, and only N^{split} was varied while using a PSL source

Table 6.1: Statistics comparing various configurations of photon splitting, a hybrid source, and PSL source. Comparisons were performed only in voxels containing > 20% of the dose in the benchmark. The γ-index, χ-index, RMSD and maximum percent difference are all shown. Statistical uncertainty was < 1% in all cases.

Plan	**Gamma (%) 1% 1 mm**	**Chi (%) 1% 1 mm**	**RMSD (%)**	**Max %Diff (%)**
PSL	99.7	99.0	0.48	2.56
PSL; $N^{split} = 20$	99.6	98.7	0.52	2.66
Hybrid 32x	88.44	93.97	0.86	4.46
Hybrid 32x; $N^{split} = 20$	68.37	78.71	1.64	8.19
Hybrid 1x; $N^{split} = 20$	90.73	94.12	0.86	5.73

for the TrueBeam 6MV accelerator. The estimated uncertainty near the isocentre was less than 1%, and the number of simulated particles was scaled by $1/N^{split}$. It was found that the efficiency was reduced for values of $N^{split} > 10$, and the accuracy was not substantially effected for $N^{split} \leq 35$. The simulation times are presented in figure 6.9. The shortest simulation time was found for $N^{split} = 10$. The primary cause for the increase in simulation times was the photon transport algorithm. This is likely due to a reduction in the speed of memory access in response to the increased memory requirements with N^{split}. This is discussed in the next section.

The best choice of N^{split} does not depend entirely on the fastest transport time. There may be source models where only a small number of independent photons are available (e.g. when a phase-space source was provided by a manufacturer, or where hard-drive storage space is limited). In these cases, larger values of N^{split} may help to reduce the variance for a fixed number of source particles.

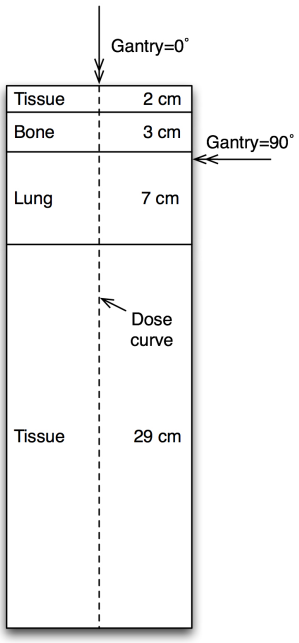

Figure 6.6: Diagram of the tissue-lung-bone phantom used for testing the photon splitting method near heterogeneities. The two gantry angles used to produce figure 6.7 are shown by arrows along the corresponding beam axis. The dose curve location is also illustrated with a dashed line.

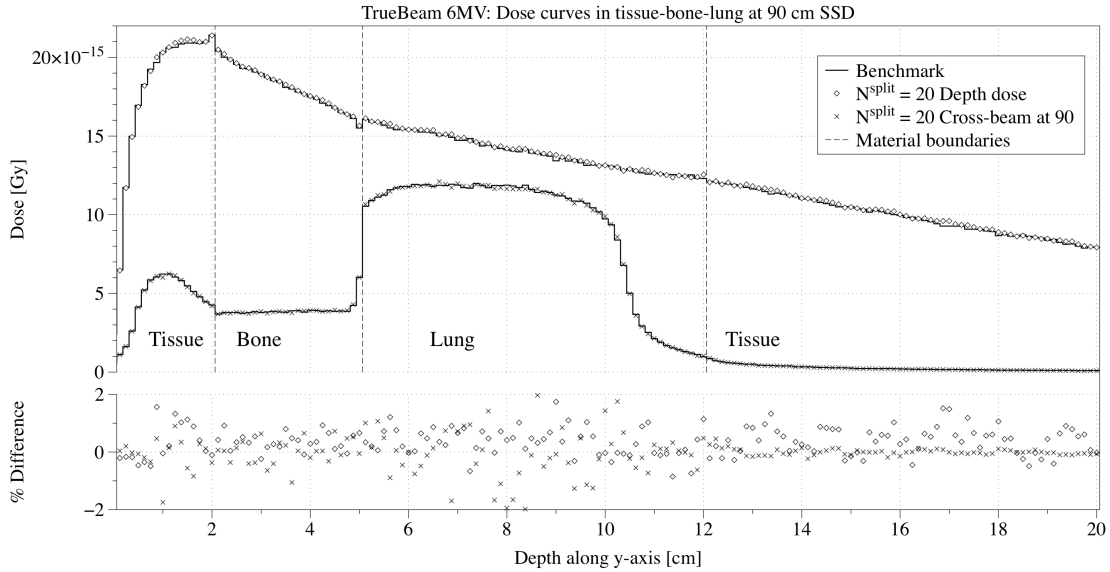

Figure 6.7: For the TrueBeam 6MV accelerator: two dose curves are shown for a 10×10 cm^2 open field for $N^{split} = 20$ (points) and the benchmark with no photon splitting (lines) using the same PSL source model. The cross-beam curve involved rotating the gantry by 90° and shifting the isocentre y-coordinate to -15 cm (this corresponds to a point on the beam-axis at depth of 5.5 cm from the top surface). The percentage differences are also shown, relative to the maximum benchmark dose in the curve.

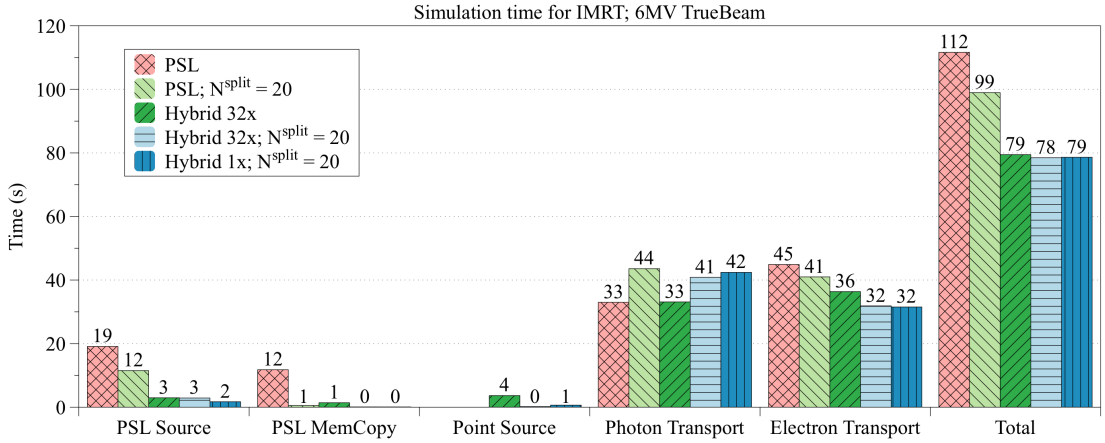

Figure 6.8: A breakdown of the simulation time for an IMRT case using the TrueBeam 6MV source. The 'PSL Source' time refers to the time spent reading and processing particles from the PSL source, while 'PSL MemCopy' is the time needed to transfer that data to the GPU. The 'Point Source' time refers to photon generation from the point source component of the hybrid model, which occurs on the GPU (so there is no memory transfer needed). The photon transport, electron transport and total simulation times are also shown. The left-to-right order of the bars in the graph is the same as the top-to-bottom order in the legend.

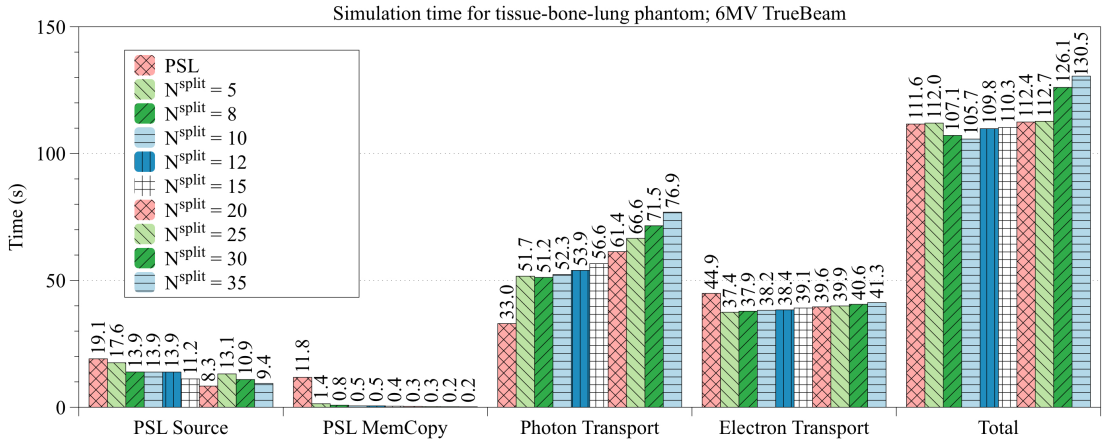

Figure 6.9: A breakdown of the simulation times for various N^{split} values using an artificial tissue-bone-lung phantom and the TrueBeam 6MV PSL source. The left-to-right order of the bars in the graph is the same as the top-to-bottom order in the legend.

6.3 Discussion and conclusions

The simulation times using photon splitting were limited by the photon transport time, which increased with N^{split} for splitting numbers > 10. This is a result of the algorithm design, which relies on split photons being stored and transported on the same thread as the original photon. Unfortunately, the memory requirements increase with N^{split} and may exceed what can be stored locally for the thread, where memory access is fastest. This problem is unique to the GPU-based implementation of photon splitting.

An alternative photon splitting algorithm could involve performing the photon splitting and ray-tracing steps (lines 6-27 in algorithm 5) in a separate kernel. The split photons could be stored in global memory to be accessed by the next kernel which performs the interaction simulation and generation of secondary particles. Since each split photon would be simulated on its own thread (instead of all on the same thread, as in algorithm 5), the required amount of locally stored data would be reduced. There would be additional global memory transfers of data generated in the first kernel (P_i, λ_{min}, ρ_i, $voxel_i$, etc.), but this would be small compared to the speed-ups as a result of the reduced memory requirements per thread and reduced thread divergence. Thread divergence should be small when sub-photons are transported together in a warp and interaction data could be shared using shared memory.

One advantage of the photon splitting VRT, is that it can be used to produce sets of electrons (from Compton interactions of the sub-photons) which all have the same energy. This has an immediate benefit of reduced thread divergence. However, it could also be used in an additional history repetition technique known as simultaneous transport of particle sets (STOPS) (Kawrakow and Fippel, 2001 [41]). In STOPS, the interaction data for electrons is shared for each set of electrons with the same energy. That is, the track of one electron is repeated for all electrons in the set, despite their different initial locations and directions. It could be possible to do this by keeping a record of these sets and then using memory that is shared between threads to share interaction data. Similarly, the precalculated track Monte Carlo algorithm (Renaud *et al.*, 2015 [67]) could be coupled with photon splitting. These challenges are left for future work, and have promise to provide an efficiency increase to the electron transport portion of the simulation.

In conclusion, the photon splitting algorithm as currently integrated into gDPM v3.0 provides a slight efficiency boost without detectable loss of accuracy. It is ex-

pected that with some re-structuring of the algorithm (separating the photon ray-tracing and interaction calculations, as described above) a greater efficiency boost could be obtained.

Chapter 7

Monte Carlo doselets for pre-treatment QA in a spherical phantom

So far in this dissertation, the solutions to high-speed dose calculation have required access to either (1) a powerful CPU computing resource, or (2) a computer with specific GPU hardware. However, for certain applications it may be possible to achieve fast dose calculations on a standard workstation using alternative algorithmic approaches. In this chapter, the specific case of pre-treatment dose verification will be investigated in this context; a novel algorithm has been developed that leverages symmetries in dose calculations using a spherical virtual water phantom.

7.1 Quality assurance and dose verification

Modern radiotherapy treatment is a complex computer controlled process that involves multiple dynamically moving components. For example, techniques such as IMRT and VMAT are now standard in clinical practice. The computerization of radiation therapy through these technologies has resulted in a complex treatment system that may obfuscate some components of the process, and despite considerable safety measures normally implemented, errors can slip into the treatment and undesirable dose can be delivered accidentally (Bogdanich 2010 [8]). Independent verification methods have therefore become essential to ensure that the treatment machines are delivering the expected radiation dose distributions to each patient.

This has motivated individual dose verification to be included as a part of some quality assurance (QA) programs (Greer, 2013 [33]). There are two levels of such dose verification: pre-treatment and *in-vivo* dosimetry.

In pre-treatment dose verification, linear accelerator (linac) output for a patient treatment plan is measured before treatment and compared with what was planned. Since pre-treatment verification occurs prior to patient set-up, actual patient positioning and treatment parameters are not captured. *In-vivo* dose verification methods use measurements during treatment to verify dose delivery. While *in-vivo* methods are rapidly evolving and likely to be used increasingly in the future (AAPM Report 87, 2005 [1]), pre-treatment dose verification is widely used.

In this dissertation, a strategy for high speed pre-treatment dose verification is proposed.

7.1.1 Pre-treatment dose verification

A variety of methods currently exist for pre-treatment verification of radiotherapy. In general, such a process involves experimental measurements of machine output that are compared to pre-calculated expectations. A typical pre-treatment procedure follows. This will usually be performed by a physicist in the days before treatment:

1. Set up the apparatus (if necessary)
2. Transfer the treatment plan files to the treatment machine
3. Deliver the treatment plan and measure the machine output
4. Compare the measurements to pre-calculated expectations
5. If an error is detected, follow the appropriate response procedure
6. If not, the plan is approved. For treatment, the treatment plan files will be transfered to the treatment machine again

Note that it is possible for data corruption to occur during the final transfer (Step 6) to the treatment machine (after the plan has already passed QA). However, a 'checksum' is transferred with a plan, re-calculated from the file received on the machine and compared to what is expected. In general, pre-treatment verification in an arbitrary 3D phantom has the ability to check for (Van Elmpt *et al.*, 2008 [87]):

- Data corrupted prior to or during transfer to the treatment machine

- MLC leaf sequencing, position and speed
- Collimator angle
- Wedge presence and direction
- Beam flatness and symmetry
- Changes in the configuration or calibration of the treatment machine since the beginning of treatment
- A treatment plan different from what was planned (for example, if the wrong patient was selected)
- An incorrectly configured TPS

Pre-treatment verification techniques vary extensively by their robustness, comprehensiveness and accuracy. Common approaches include the use of ionization chamber arrays (van Esch *et al.* 2007 [89], Saminathan *et al.* 2010 [72], Godart *et al.* 2011 [29]) and diode arrays (Feygelman *et al.* 2011 [22]). Such detector arrays provide effective 3D measurements, but tend to be expensive, heavy devices that make QA setup less than ideal. In addition, most of the current detector arrays are intended exclusively for use with coplanar treatment deliveries. Alternatively, direct analysis of machine log files can provide valuable information on treatment delivery (Litzenberg *et al.* 2002 [51] and Stell *et al.* 2004 [79]), but does not reveal the impact of potential errors on the dose to a patient. Log files in combination with MC calculations, using the DOSXYZnrc code, have also been used for pre-treatment QA (Teke *et al.* 2010 [80]) and provide accurate calculation of the dose delivered at the QA session. However, MC based systems require significant expertise and dedicated computing resources, limiting their use to relatively few clinics worldwide. Due to the near ubiquitous adaptation of EPIDs by linac vendors, electronic portal image (EPI)-based verification techniques have become an active area of research (for literature reviews of the subject, see van Elmpt *et al.* (2008a [87]) and Greer (2013 [33])). These techniques utilize linearity of EPI dose response (Greer and Popescu 2003 [34]) and have the advantage of using a device that is already present on the linac.

EPID dosimetry can be carried out using either transmission (with a patient or phantom in place) or non-transmission measurements. Both schemes can be used for either 2D fluence comparisons or 3D reconstructions to achieve verification. However, it was shown by Kruse (2010 [49]) that 2D verification of IMRT plans is insensitive and not sufficient to detect important dosimetric inaccuracies. Using non-transmission

EPIs, Ansbacher (2006 [2]) developed a technique for 3D dose reconstruction in a cylindrical water phantom, van Elmpt *et al.* (2008b [88]) reported a method for calculating 3D dose (MC) inside inhomogeneous phantoms, and van Zijtveld *et al.* (2009 [90]) used non-transmission EPIs to produce fluence maps for dose calculation in the TPS. Yeo *et al.* (2009 [98]) presented an algorithmic method for dose reconstruction in the patient from transmission EPIs, as did Mans *et al.* (2010 [57]) using a back-projection method. Dose reconstructions in 3D tend to be more complex, since it is necessary to account for scattering that occurs in both the reconstruction volume and the imager itself. Solutions commonly assume coplanar treatment geometry (Ansbacher 2006 [2]) and therefore only allow verification of coplanar treatments. While non-coplanar treatments are typically uncommon in IMRT and VMAT, they have demonstrated considerably improved dose conformity (Pugachev *et al.* 2001 [66], Wang *et al.* 2005 [95], Llacer *et al.* 2009 [52], Clark *et al.* 2012, Panet-Raymond *et al.* 2012 [63]) that is likely to be exploited increasingly in the near future. The dosimetric advantages are clear (normal tissue dose is spread over a larger volume), but implementation has generally been limited by the difficulties involved with treatment couch motion, which requires increased staff involvement, treatment time and more complex QA procedures. A modern generation of linacs with integrated robotic couch motion reduces the difficulties associated with treatment delivery, but the problem of robust QA procedure remains.

In a previous work, our group presented an effective solution called the phase-space modulation (PSM) method (Berman *et al.* 2010 [6]). In the PSM method, the electronic portal image (EPI) signal was deconvolved to remove imager scatter and produce a fluence map representing the radiation intensity from the treatment beam incident on the imager. EPIs were collected prior to treatment, without the patient in place. Each particle in a plan-independent phase-space was then weighted according to the value of the fluence element intercepted by its projected path (assuming no scattering in air). The resulting phase-space was used for MC dose calculation to reconstruct the dose distribution that would have been delivered to a phantom or patient, allowing for pre-treatment verification. The success of this strategy has led us to develop a new method that avoids time-consuming MC dose calculation during pre-treatment QA.

The following sections describe an EPI-based pre-treatment verification technique capable of fast reconstruction of 3D dose distributions from both coplanar and non-coplanar treatments. This will be referred to as the spherical doselet mod-

ulation (SDM) method. The dose reconstruction is based on EPI measurements, and performed by modulating pre-calculated MC doselets in a virtual spherical water phantom. The novelty of this technique is in essentially eliminating the statistical uncertainty of MC dose calculations by combining azimuthal symmetry in a plan-independent phase-space with the spherical symmetry of a water phantom to derive plan-independent radial beamlet dose distributions (doselets). Only a small number of doselets are necessary for accurate dose reconstruction, compared to thousands when this symmetry is not exploited. The plan-dependent components of the linac are accounted for by weighting the doselets based on EPI signals. Doselets of this type have not been used before in radiotherapy calculations, though they offer considerable advantages for QA applications.

7.2 The spherical doselet modulation (SDM) method

7.2.1 Phase-space sorting using azimuthal particle redistribution

The MC particle transport code BEAMnrc (Kawrakow and Walters 2006 [46]) was used for generating a planar phase-space upstream of all plan-dependent beam-shaping apertures in a 6 MV Varian Clinac 21EX (Varian Medical Systems, Palo Alto, CA, USA) treatment head. An azimuthally symmetric circular source of electrons incident on the target was used. When transported through the azimuthally symmetric physical geometry of the accelerator, the resulting particle distribution shares the same symmetry. The phase-space was spatially discretized into beamlets, which, after having been transported through a phantom produced doselets in units of Gy per incident electron. In the context of this work, a beamlet refers to a single unique spatial region of the phase-space (no two beamlets may contain the same particles). Usually beamlets are defined to fill a Cartesian grid for a planar phase-space (Bush *et al.* 2008a [13]), but in this work beamlets were produced on a cylindrical grid defined by annular sectors. Due to rotational symmetry, all beamlets from a given annulus are dosimetrically equivalent. Therefore, using a modified azimuthal particle redistribution (APR) technique (see section 4.1.6 for the standard implementation), phase-space particles were redistributed into a single annular sector for each annulus, as illustrated in figure 7.1, dramatically increasing the particle density in the beamlet. Using a 1° azimuthal size for a sector, the particle density will increase by 360 times,

resulting in nearly 19-fold dose calculation uncertainty reduction.

At the centre of the grid a circular region was defined as an additional beamlet, instead of being divided azimuthally. This was done because of the small area of the region. Further divisions into smaller beamlets would increase dose reconstruction time with little effect on the resulting distribution. In order to increase the particle density in this region equivalently to the outer annuli, standard APR was performed (as in section 4.1.6) with recycling equal to the number of sectors, N_ϕ.

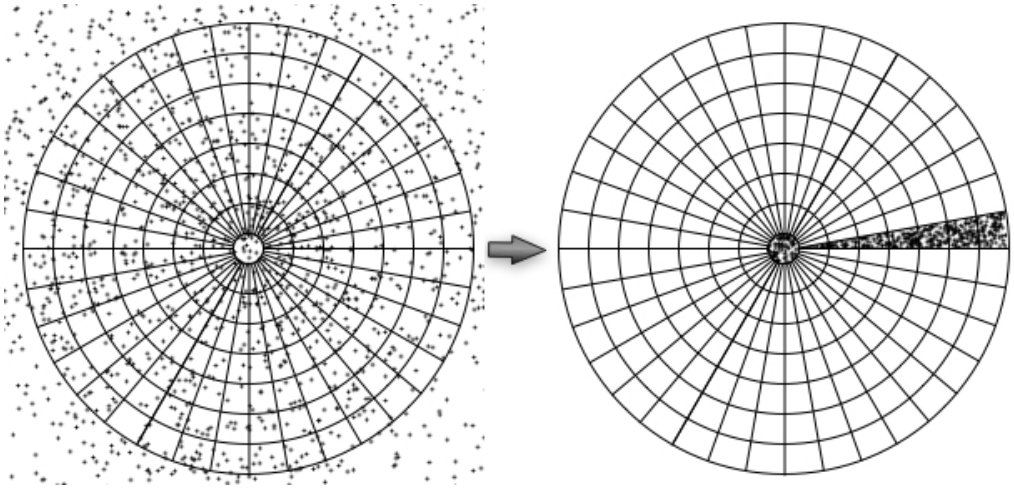

Figure 7.1: This figure visually illustrates the beamlet generation procedure. Particles in the phase-space were sorted using a cylindrical grid. APR was used to rotate particles into a single annular sector per annulus. In the central circular beamlet, recycling with APR is performed to similarly increase the particle density.

For this work, the phase-space was divided into $N_a = 50$ annuli and one central circular sector within an outer circular boundary of radius 10.1 cm (projected to the isocentre). The central circular beamlet was 0.2 cm in diameter at the isocentre, the same as the radial thickness of each annulus. The particles in each annulus were compressed into a single corresponding annular sector beamlet of approximately $1.42\,^\circ$ in angular size. This size corresponds to dividing each annulus into $N_\phi = 254$ sectors. The number of annuli and annular sectors were chosen by trial-and-error to achieve a reasonable balance of speed and accuracy for our system.

7.2.2 Generation of plan-independent Monte Carlo doselets

A virtual phantom was created that contained a water sphere of 20 cm diameter, surrounded by a $20.25 \times 20.25 \times 20.25$ cm^3 cube of air. Voxel dimensions of $0.125 \times$

0.125×0.125 cm^3 were used in the phantom.

For each of the beamlets defined by the phase-space sorting algorithm described above, DOSXYZnrc (Kawrakow and Walters 2006 [46]) was used to generate doselets in this phantom. The doselets were initially calculated in Cartesian coordinates, and then converted into spherical coordinates (with origin at the centre of the water sphere) using tri-linear interpolation. In subsequent plan verification calculations, the plan isocentre was set to be at the centre of the spherical phantom at the source-to-axis distance (SAD). This allows us to define a spherical voxel system that, in the azimuthal plane, aligns with projected beamlets discussed in the previous section. That is, the radial discretization in the spherical voxel system corresponds 1-to-1 with the annuli used to derive beamlets. Similarly, the azimuthal discretization of the voxel system corresponds with beamlet annular sectors. The spherical voxel system is a key component in our method, as it removes the necessity for time-consuming dose interpolations during dose reconstruction (section 7.2.5) that could also introduce interpolation artefacts.

To correspond with the beamlets, the spherical voxel system was defined with 0.2 cm radial divisions, $N_\phi = 254$ ($\sim 1.42\,^\circ$) azimuthal elements and $N_\theta = 134$ ($\sim 1.34\,^\circ$) polar elements. Since the central circular beamlet was chosen as 0.1 cm in radius, the first radial division in the coordinate system was similarly reduced to 0.1 cm.

The entire process thus far is plan-independent and needs to be performed only once per phase-space (i.e. linac beam). One set of doselets can be re-used for all dose reconstructions so long as they require the same source model and spherical phantom. Thus high particle densities and small voxel resolution can be used during MC dose calculation to ensure the initial doselets are accurate and precise, at no sacrifice to the speed of later dose reconstruction.

Cross-beam profiles of a few selected doselets are shown in figure 7.2. Doselets are normalized independently with the maximum set to 1.

7.2.3 Construction of fluence maps from EPIs

Similar to other pre-treatment verification methods using EPIs, plan-dependent portal images were collected prior to treatment. The portal image acquisition strategy described by Ansbacher (2006 [2]) was followed. The EPID used was a Varian aS500 amorphous silicon device attached to a 6 MV Varian Clinac 21 EX. This has an active imaging area of 512×384 pixels^2 with aSi light sensitive photodiodes and a pitch of

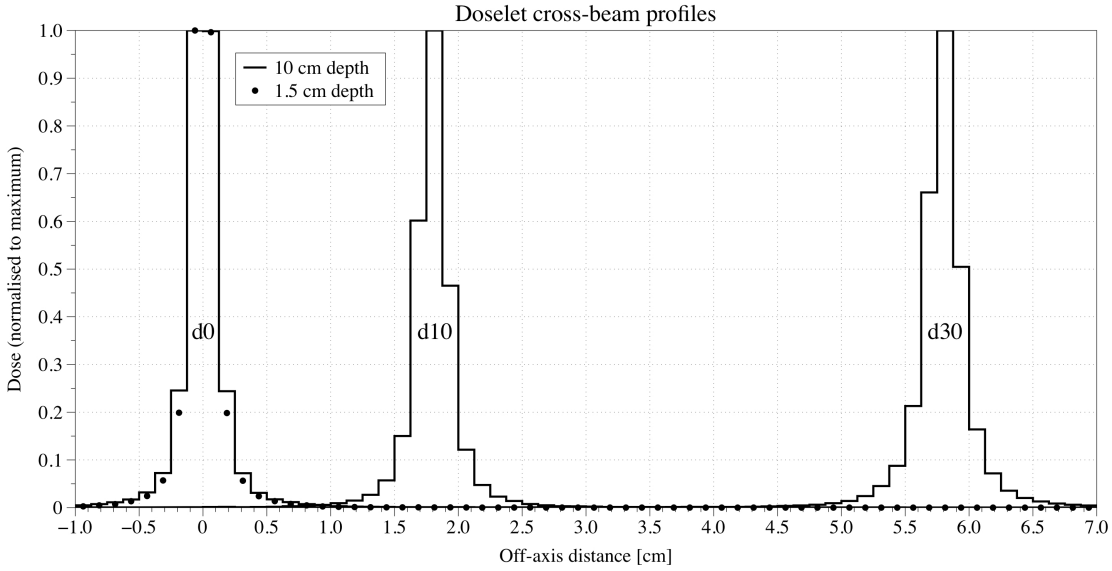

Figure 7.2: Cross-beam profiles for the doselets indexed from smallest radius outward as the $d0$, $d10$ and $d30$, where $d0$ is the central-axis (first) doselet, $d10$ is the 11th doselet, and so on. For the central-axis doselet, two profiles are shown at a depth of 10 cm (lines) and 1.5 cm (dots). Doselets are normalized independently with the maximum set to 1.

0.784 mm. The EPID was positioned at a source-to-image distance (SID) of about 110 cm (where the effective resolution at the isocentre is approximately 0.7 mm) and then repositioned longitudinally in order to optimally include the area of all treatment fields. The specific SID used is not critical for the method since this is accounted for during calibration. Using a delivery rate of 400 MU/min to avoid possible image saturation, three images were acquired from dark, flood and 10×10 cm^2 calibration fields. The treatment field images were then collected using the gantry angles specified in the treatment plan, and corrected by subtracting the dark field and dividing by the flood field images. Since couch rotations can result in collisions with the EPID, the couch was not rotated during image acquisition. Instead, couch rotations were performed during dose reconstruction, as described in section 7.2.5. Collimator rotations were executed for the acquisition. The images were normalized by the calibration field. Fluence maps were constructed by deconvolving the corrected EPIs using a kernel that accounts for scatter in the imager. This process was described in Berman *et al.* 2010 [6]. For the purpose of the current work, the fluence map represents a weighting matrix with elements geometrically defined to be identical to the cylindrical grid used to generate the beamlets as projected to the isocentre.

The deconvolution of the image was performed using the Fourier convolution theorem, according to which the convolution operation of two functions can be calculated as the Fourier transform of pointwise product of Fourier transformed functions. The effect of scatter is removed by dividing the 'blurred' image in the frequency domain $\hat{f}$ by the Fourier transformed scatter kernel $\hat{k}$. The Fourier transformed image with scatter removed, $\hat{g}$ is calculated as,

$$f = g \otimes k \equiv \int_{-\infty}^{\infty} g(\tau)k(t-\tau)d\tau \tag{7.1}$$

$$\hat{f} = \hat{g} \cdot \hat{k} \Rightarrow \hat{g} = \frac{\hat{f}}{\hat{k}}. \tag{7.2}$$

7.2.4 Doselet dose conversion to absolute units

The phase-space source used for doselet generation was calibrated using a standard MC simulation under calibration conditions (Popescu *et al.* 2005 [64]). The normalization factor $D_{10\times10}$ was derived as the MC dose (in Gy/e-) at the central axis of a 10×10 cm^2 field at 10 cm depth in water. The tissue maximum ratio (TMR) was used with a measured reference dose $D^{measured}$ as calibration. The SDM method requires two further factors for absolute dose normalization: the number of azimuthal sectors, N_ϕ, to account for increased particle density from APR, and the monitor units used for producing the 10×10 cm^2 calibration EPI, MU_{cal}. The fluence map elements $w_{\phi r}$ that were derived from EPIs already account for the monitor units as well as monitor chamber backscatter (Zhu *et al.* 2009 [101], Zavgorodni *et al.* 2014 [99]) from each field. Recall that the number of fluence map elements is equal to the number of doselets used for dose reconstruction, namely $N_\phi N_a + 1$. The conversion of the doselet dose $D_{\phi r}$ from the relative units of Gy/e- to absolute dose $D_{\phi r}^{abs}$ in Gy is performed as

$$D_{\phi r}^{abs} = D_{\phi r} w_{\phi r} \frac{MU_{cal} TMR(10, 10\times10) D^{measured}(d^{max}, 10\times10)}{D_{10\times10} N_\phi}. \tag{7.3}$$

7.2.5 Re-constructing 3D dose in a sphere

Once we have plan-dependent fluence maps derived from EPIs, dose reconstruction can begin, as outlined by the flowchart in figure 7.3. Since the doselets are generated only for one annular sector per annulus from the phase-space, each doselet must be

azimuthally rotated and re-used N_ϕ times to fill an annulus. Since elements on the azimuthal plane of the spherical coordinate system of the dose distributions correspond with beamlet divisions, the dose from a given annulus can be collected by: (1) repeatedly re-indexing the azimuthal coordinates of the corresponding doselet, (2) multiplying the original dose values by the fluence element weighting factor and (3) adding the dose in each re-indexed voxel to the cumulative dose matrix for the field. There is one exception to this process - the central doselet was defined corresponding to only a single element in the fluence map, so no rotation is necessary. The doselet rotation/modulation/summation procedure is the most computationally intensive part of the SDM method, since the entire 3D dose distribution for each doselet must be re-indexed, scaled and summed N_ϕ times. To reduce the required number of operations, voxels in a doselet containing zero dose are ignored. Once every non-zero fluence map element has been used, the sphere will be filled with a patient-specific dose distribution from one treatment field. Exactly the same procedure is done to derive the dose distributions for each treatment field in the plan.

The only machine rotation included in the fluence map implicitly is that of the collimator rotation; gantry and couch rotations must be performed on the reconstructed dose directly. The gantry angle used is that from the delivery (rather than the planned rotation). Since the couch rotations are not performed during acquisition, the planned rotations must be used.

With the sphere centred at the isocentre, a simple method to account for gantry and couch rotations would be to re-indexing the dose coordinates along the polar and azimuthal directions, respectively. This technique would provide fast dose rotations, but is a 'nearest neighbour' approximation. Since rotations are necessary only once per field (after dose reconstruction has already been performed) the contribution to the overall calculation time is small. Thus more advanced interpolations were used with little sacrifice to the calculation time. The dose for each field was first converted back into Cartesian coordinates, and then the delivered gantry and planned couch rotations were induced using cubic spline interpolation. Recall that the original dose calculations from the treatment planning system are in Cartesian coordinates, so this conversion is necessary for 1-to-1 voxel comparison. The conversion from spherical to Cartesian coordinates was performed using tri-linear interpolation. After the above procedure was completed for each field, the contributions were summed to obtain the total dose.

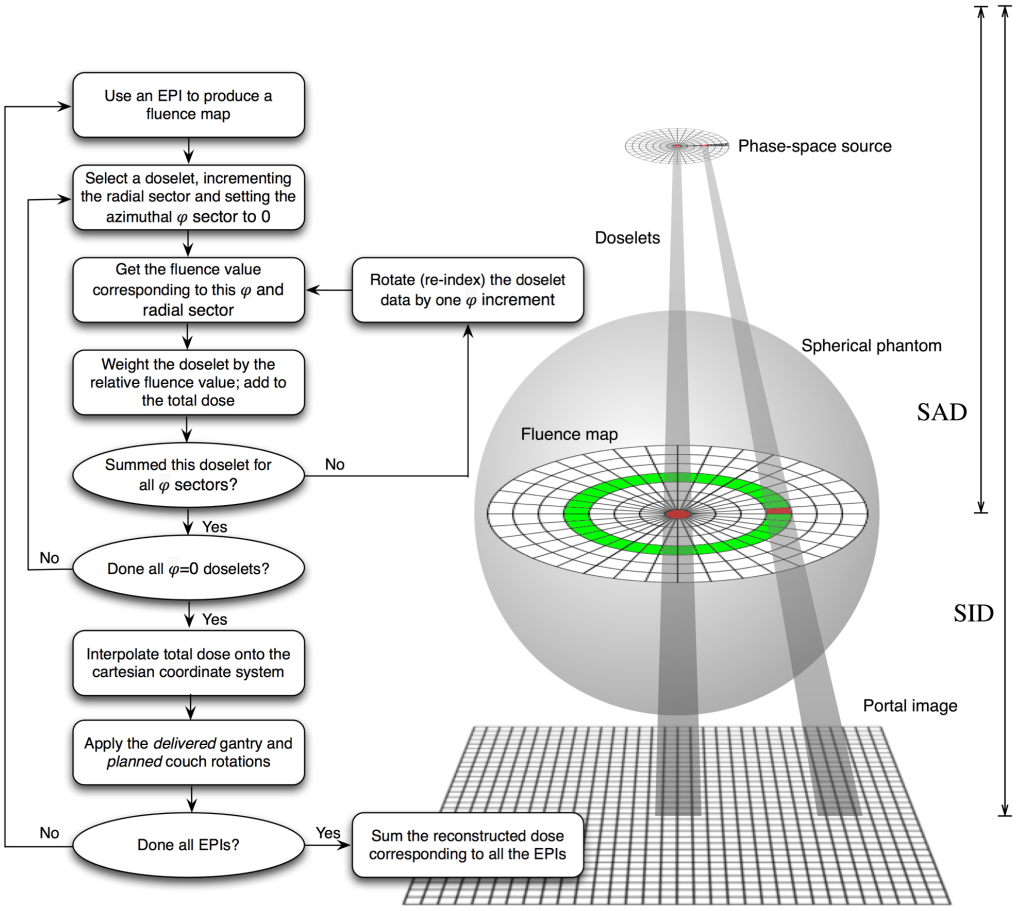

Figure 7.3: A flowchart, starting at the top-left of the image, and diagram illustrating how beamlets from the phase-space source were used to calculate doselets in a spherical phantom (not to scale). In this process, an electronic portal image (EPI) was deconvolved to produce a fluence map projected to the SAD. The fluence map elements corresponding to two doselets are highlighted in the figure.

7.3 Benchmarking against standard MC and the PSM method

The SDM method has been tested against standard MC simulations and the PSM method. Since the purpose of these tests was to illustrate the accuracy of the SDM method, all simulations shared identical treatment parameters. In other words, the same gantry angles, MUs, etc were used in all cases (rather than comparing planned versus delivered, as in a true pre-treatment verification).

The PSM and MC benchmark utilized DOSXYZnrc for dose calculation in a cubic

phantom with $2.5 \times 2.5 \times 2.5$ mm^3 voxel resolution containing a 10 cm radius water sphere. After dose reconstruction in spherical coordinates, the SDM dose results were also interpolated to match this phantom. The original doselets were calculated on a higher resolution phantom, described in section 7.2.2.

All methods used the same phase-space source scored just above the secondary collimators from a BEAMnrc model of a 6 MV Varian Clinac 21EX (Varian Medical Systems, Palo Alto, CA, USA). This model has been validated previously (Bush *et al.* 2007 [16], 2009 [15], 2011 [12], Gagne and Zavgorodni 2007 [27]). In the benchmark MC calculations, secondary collimator simulation was performed using BEAMnrc. For the PSM and SDM methods, beam-shaping information is inherent to the measured EPIs.

The energy cut-offs (PCUT and ECUT) in DOSXYZnrc for photons and electrons were 0.010 MeV and 0.700 MeV, respectively. In every plan the isocentre was set to the centre of the spherical phantom at 100 cm SAD. Computations were performed on the cavivake CPU cluster (section 3.3). The benchmark and PSM simulations utilized the Vancouver Island Monte Carlo (VIMC) framework for streamlined dose calculation (Zavgorodni *et al.* 2007 [100], Bush *et al.* 2008b [14]). The SDM method has now also been integrated into this framework.

Open fields of size 1×1, 3×3, 5×5, 10×10, 15×15 and 20×20 cm^2 were considered, along with seven clinical head and neck IMRT cases (five brain, one larnyx and one left tonsil). One of the brain cases, 'Brain5', included two non-coplanar fields.

The average total calculation time for dose reconstruction *per EPI* was 96 s using a single 2.1 GHz processor. This time includes applying portal image calibration corrections, deriving a fluence map, processing doselets (re-indexing voxels, modulating and summing), and applying gantry and couch rotations. The bulk of this time (81 s) is spent re-indexing voxels, modulating and summing doselets.

Depth dose and cross-beam profiles for a variety of open field sizes are provided in figures 7.4 and 7.5, respectively. The percentage difference between the SDM method and MC benchmark was plotted in these figures, relative to the maximum dose in the benchmark. The SDM method exhibits systematic errors in penumbra regions due to doselet discretization. In the PSM method, errors near the field edges arise from the fact that collimators are modeled using a planar fluence map. Recall the PhspC model, where two rejection planes are used for collimation, located at the top surface of each secondary collimator. The fluence map collimation method instead weights particles all on the same plane, resulting in a different (and less accurate) particle

distribution due to the motion of particles before reaching the plane.

χ- and γ-index tests were performed using 2%/2 mm criteria using voxels above the 2% isodose in the MC reference, and limited to 9 cm radius from the centre of the sphere. The results of these tests are provided in table 7.1, and isodose images for the 'Brain5' case (with two non-coplanar fields) are shown in figure 7.6. Over the IMRT cases, the averages of the χ- and γ-index test results for the SDM method were 98.7% and 98.8%, respectively, and the average RMSD was 1.6%.

	Gamma (%)		**Chi (%)**		**RMSD (%)**	
Plan	**SDM**	**PSM**	**SDM**	**PSM**	**SDM**	**PSM**
1×1	96.4	95.3	100.0	100.0	3.2	2.5
3×3	95.6	94.0	100.0	98.8	2.0	1.8
5×5	96.3	94.7	99.9	98.5	1.8	1.7
10×10	96.2	95.6	99.6	99.5	2.2	1.6
15×15	94.6	89.1	96.4	91.3	2.0	4.0
20×20	97.0	100.0	96.0	100.0	1.1	0.6
Larynx	97.9	98.6	97.9	98.2	1.4	1.5
LT Tonsil	99.1	99.2	99.2	99.0	1.2	1.3
Brain1	99.6	99.0	99.7	98.4	1.9	1.4
Brain2	96.5	96.0	97.4	94.9	2.2	2.3
Brain3	99.4	99.4	99.5	99.2	1.4	1.5
Brain4	99.5	99.0	99.5	98.3	1.5	1.5
Brain5	98.7	98.1	98.7	97.9	1.8	1.9

Table 7.1: Results comparing the SDM method and PSM method to a standard MC calculation. All simulations were performed in a 10 cm radius spherical phantom. χ- and γ-index tests used 2%/2 mm criteria restricted to voxels within 9 cm radius of the sphere centre above the 2% isodose in the reference. The RMSDs are also shown.

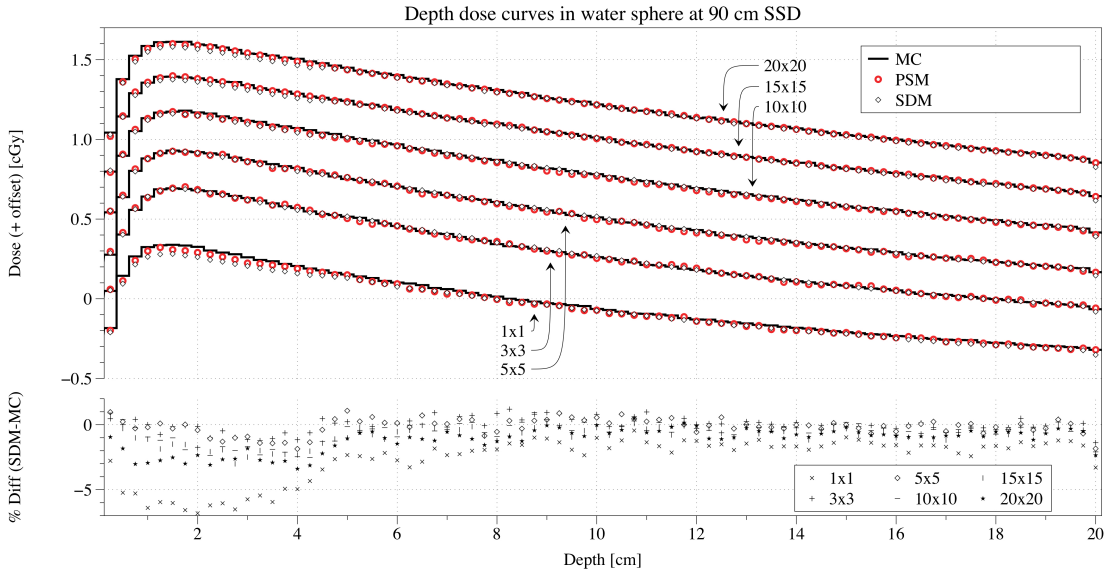

Figure 7.4: Depth dose curves are shown for a variety of field sizes for the SDM method, PSM method, and a standard MC simulation. Curves were artificially offset by 0.2 cGy increments for clarity, except for the 10×10 cm^2 curve. The percentage differences between the SDM method and MC simulation are shown, relative to the maximum MC dose. All simulations were performed in a spherical 10 cm radius water phantom at 90 cm SSD.

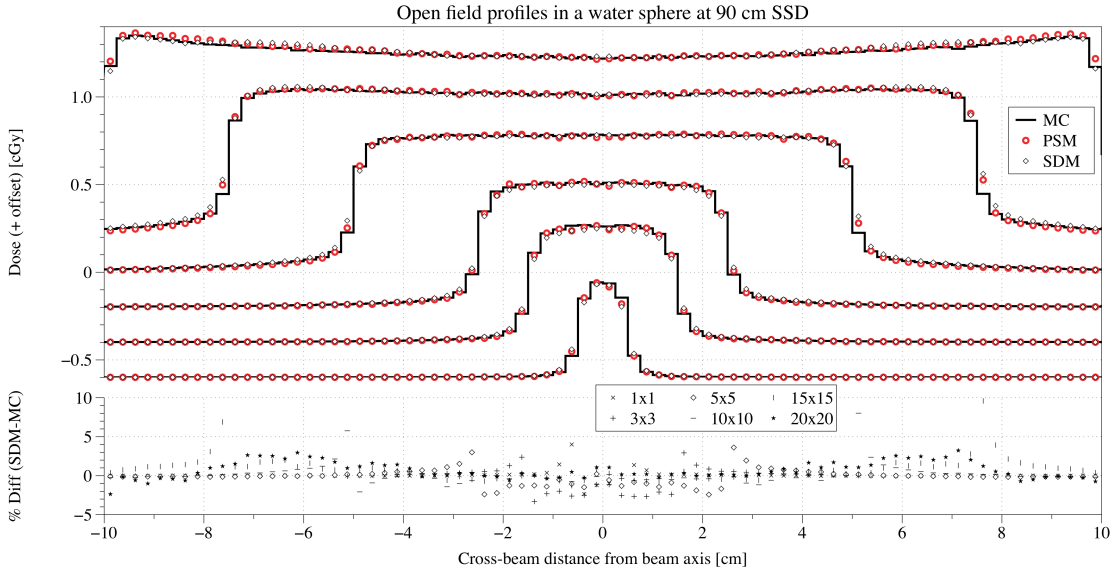

Figure 7.5: Cross-beam profile curves at 10 cm depth are shown for a variety of field sizes for the SDM method, PSM method, and a standard MC simulation. Curves were artificially offset by 0.2 cGy increments for clarity, except for the 10×10 cm^2 curve. The percentage differences between the SDM method and MC simulation are shown, relative to the maximum MC dose. All simulations were performed in a spherical 10 cm radius water phantom at 90 cm SSD.

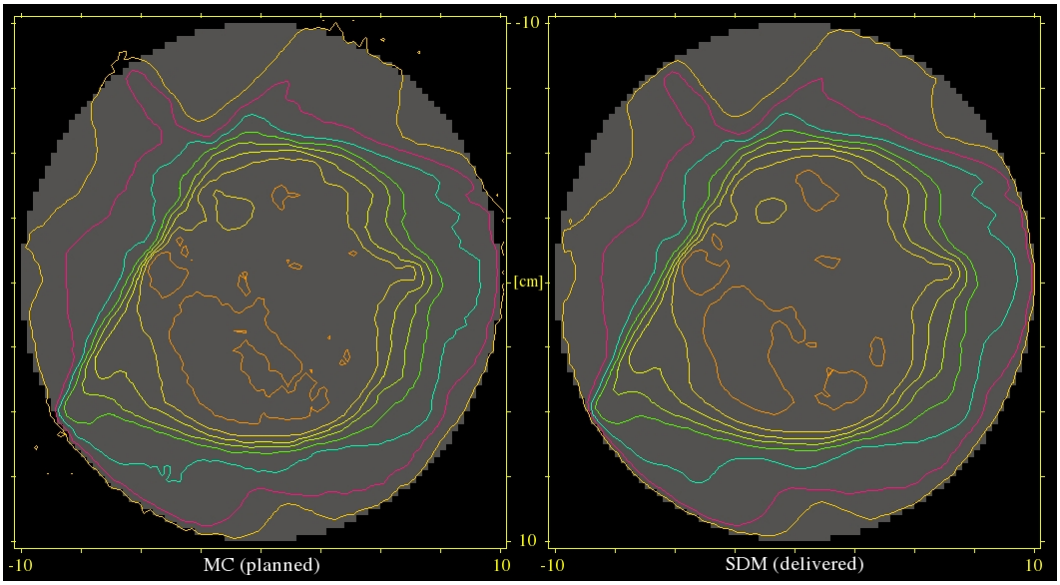

Figure 7.6: Isodose curves for the non-coplanar 'Brain5' IMRT treatment plan, the MC benchmark on the left and SDM method on the right. The slice shown is in the coronal plane at the centre of the sphere. Isodose lines start at 10% of the maximum dose and increment in 10% intervals up to 90%.

7.4 Discussion and conclusions

Existing dose verification methods that use convolution to account for EPI and phantom scatter commonly use an invariable scatter kernel, resulting in calculation accuracy loss off-axis and at-depth (Ansbacher 2006 [2]). The SDM method avoids these problems by using MC calculated doselets that inherently account for fluence spectral changes as photons get transported through the linac head and phantom. Since the doselets are plan-independent and only calculated once, it is also possible to achieve very low statistical uncertainty using long initial simulation times. Further use of the doselets for QA calculations does not require any MC simulations to be performed, so the necessary computational overhead and operator expertise is minimized. This makes it possible to share the doselet database between institutions, where a MC program may be difficult to establish. The SDM method also has the advantage of minimal experimental set-up, requiring only measurements in air using an EPID. Non-coplanar treatments are supported even in cases where the couch would collide with the imager, since the in-air measurements do not require rotation of the couch (rather, it is sufficient to simply rotate the reconstructed field dose distribution).

For this work, algorithm parameters were selected to provide a reasonable trade-off between dose reconstruction speed and accuracy. For different hardware configurations and accuracy requirements, these parameters could vary. The most important of these is the dose voxel resolution in the spherical coordinate system. The dose reconstruction time is approximately inversely proportional to the voxel volume. To mitigate accuracy dependency on voxel dimensions, alternative interpolation methods such as cubic spline could be considered. Additionally, modeling particle transport directly in spherical coordinates would eliminate the need for Cartesian to spherical interpolation (though conversion from spherical to Cartesian coordinates after dose reconstruction would still be necessary).

The rotational symmetries utilized in this method require that the MC source model also exhibit such symmetries. However, EPID measurements will contain asymmetries of the physical beam (e.g. an asymmetric focal spot) that are not modeled. As a result, comparisons between the SDM method and a benchmark will include systematic delivery discrepancies. Of course, the asymmetry errors would be removed if the benchmark model was commissioned asymmetrically.

The SDM method uses the delivered gantry rotation according to the delivery log files produced by the treatment machine. Assuming the log files are correct, this pro-

vides validation of the gantry motion. However, the method would be susceptible to errors in the gantry position readout. The collimator rotations are directly measured by the EPID, so validated effectively. Since the couch is not rotated during EPID measurements, the delivery couch position cannot be tested by this method.

The spherical geometry of the verification phantom used in this study is different to the commonly used cylindrical water phantoms. The reasons for the more conventional cylindrical shape of verification phantom include similarity to human geometry, rotational symmetry that simplifies dose calculations for coplanar treatments, and ease of physical construction when used in experimental verifications. In our technique the ease of construction is irrelevant, as the phantom is 'virtual' and does not actually need to be manufactured. Also, so long as the attenuation of the radiation field is not excessive (nor insufficient) near the regions of interest, the shape of the phantom does not need to mimic the human form to achieve meaningful dose verification. A spherical surface simplifies the dose reconstruction algorithm by avoiding the necessity of corrections for irregularities (unlike the edges of a cylinder, that can be encountered in non-coplanar treatments). For the SDM method, the spherical shape was also essential for the high efficiency of the technique.

In the case of large field sizes or treatment areas, it might be necessary to increase the radius of the sphere to encompass the entire field. However, doing so would increase beam attenuation, potentially reducing the sensitivity of the method to capture treatment errors. The virtual nature of the SDM method could offer one solution - simply reduce the phantom density as the radius is increased. So long as the same phantom is used both for the planning and verification dose calculations, a reasonable change in water density would not inhibit the usefulness of the SDM method. Investigation of this idea is left for future work.

The SDM method could also be adapted for post-treatment verification, where measurements during treatment are used for dose reconstruction. Dose reconstructions in the patient geometry would not be possible, since the SDM method is fundamentally restricted to using a spherical phantom, but measurements during treatment could be used instead of pre-treatment EPID signal. For example, log files of the delivered MLC motions could be used to reconstruct a fluence map for use in SDM. In reality, any measurements that can be used to create a fluence map could be used with SDM, so long as the resolution of the fluence maps elements is sufficiently high to provide an accurate fluence estimation.

In summary, the SDM method has been shown to provide effective 3D dose verifi-

cation using EPIs in conjunction with pre-calculated MC doselets. High efficiency calculations were achieved by azimuthally compressing a plan-independent phase-space into beamlets with extremely high particle density. Only a small number of doselets in a spherical phantom were required to perform accurate dose reconstruction. This novel strategy exploited rotational symmetries to mitigate the most computationally intensive part of dose reconstruction - reading, weighting and summing dose matrices.

Chapter 8

Final conclusions

This dissertation presented methods for high speed MC radiotherapy dose calculation. It was found that substantial speed enhancements could be obtained through careful consideration of the bottlenecks in treatment head modeling and dose calculation. Using approximate modeling of the secondary collimators and MLC in combination with a dose calculation code utilizing suitable variance reduction techniques was essential in achieving simulation times on the order of minutes, instead of hours.

For CPU-based calculations, the combination of (1) the phase-space collimation (PhspC) technique for secondary collimator modeling, (2) the vcuDMLCcode for MLC modeling and (3) VMC++ for dose calculation was found to provide the fastest simulation times with accuracy sufficient for most clinical applications. Even with this configuration, calculation speeds on the order of minutes can only be obtained with a powerful computing resource. For example, a typical IMRT plan might require around 64 cores with >1 GB RAM per core.

GPU-based dose calculations are a relatively low cost alternative for high speed MC simulations. However, the standard algorithms developed for CPU-based MC codes are not necessarily efficient on the GPU. Careful design of the source modeling, transport mechanics and variance reduction techniques plays a substantial role in the overall efficiency. It was found that efficient generation of source particles was particularly important. Using a single GPU device, simulation times on the order of minutes for typical IMRT plans were achieved.

The advancements of dose calculation algorithms in this work have applications in clinical radiotherapy treatment planning and quality assurance of advanced techniques such as IMRT and arc therapy. With MC dose calculations requiring on the order or minutes or less, their integration into treatment planning systems and qual-

ity assurance programs is feasible. The algorithmic developments presented in this dissertation reduce the dependence of the software on expensive hardware, increasing the ability of clinics to implement MC programs.

8.1 Dissertation summary

The first investigation in this dissertation involved one of the most difficult bottlenecks in MC simulation times of a linear accelerator. This is the modeling of secondary collimators. In order to achieve high particle density in the beam emitted from the linac, it is generally necessary to simulate a very large number of particles which are subsequently absorbed in the collimators. The simulation of the collimation process is very computationally expensive, often requiring more time than the simulation of dose deposition in the patient. Instead of simulating photon and electron interactions in the collimators, the top surface of the collimators were instead assumed to be perfectly absorbing. The other surfaces of the collimators were not modeled. The result is a technique (called the PhspC method) for very quickly processing the particles in a phase-space by simply ray-tracing (ignoring scatter in air) to the collimator surfaces and removing from the simulation all those which intersect a surface. However, this creates a different distribution of particles due to the lack of scatter and different attenuation modeling. Over 9 test cases, the average γ-index agreement with standard MC for 1%/1 mm criteria and above the 10% isodose was 97.8%. The errors observed were systematic and isolated to out-of-field regions, where the PhspC method delivered less dose compared to MC. The speed-up in secondary collimator simulation was a factor of 39, 53 and 100 for 4×4, 10×10 and 30×30 cm^2 open fields, respectively.

With secondary collimator modeling now reduced to a simple and fast process, the next step was to reduce dose calculation times. This was done by switching out the standard MC dose calculation codes (DOSXYZnrc or VMC++, for example), with the GPU-based software, gDPM. However, gDPM was initially in an early stage of development without support for phase-space sources. Once implemented, the direct use of standard phase-space sources resulted in very low performance on the GPU, due to the random ordering of particles causing poor hardware utilization. A novel phase-space source pre-processing technique that sorted the particles by energy, type and position into *phase-space-lets* (PSLs) remedied this. When combined with the PhspC technique for secondary collimator modeling, gDPM attained simulation times on the order of minutes on a single GPU device. For open field cases, the average

γ-index agreement of a PSL source in gDPM with standard MC for 1%/1 mm criteria above the 10% isodose was 96.4%. For three IMRT plans and 2%/2 mm above 10%, the average agreement was 95.3%. The discrepancies exist due to a combination of differences in (1) secondary collimator model, (2) MLC model and (3) transport mechanics.

In gDPM, the PSL source model was found to still comprise a large portion of the total simulation time. To remedy this, a hybrid source model was developed that combined a point source model of focal spot photons with extra-focal contributions from a phase-space-let source. The point source component was modeled by sampling from 2D histograms over radius and energy that were derived from a phase-space previously calculated upstream of the secondary collimators. To extract focal photons from a phase-space that does not contain LATCH records, a ray-tracing method was designed and shown to provide an accurate characterization. The hybrid source was integrated into the GPU-based dose calculation engine, gDPM v3.0, and tested against standard PSL simulations. Source generation efficiency improvement over using only a PSL source was a factor of 4-5 for open fields. Comparisons of open fields with PSL simulations yielded, on average, agreement in 99% of the voxels above the 2% isodose for 1% / 1 mm chi-test criteria, and a RMSD of 0.5%. A 7-field IMRT patient treatment achieved 95% chi-test agreement for 1% / 1 mm criteria above the 10% isodose, 99.8% for 2% / 2 mm, a RMSD of 0.8%, and source generation speed-up factor of 2.5.

The hybrid source model was successful but resulted in a reduction of dose calculation accuracy. As an alternative way to reduce the source modeling time, the photon splitting variance reduction technique was implemented into gDPM. This involved splitting and redistributing photons are they are transported through the calculation volume. Photon splitting resulted in a reduction of source generation time, but increased photon transport time. As a result, only small efficiency improvements could be attained. However, an alternative algorithm was proposed that, in future work, could reduce the photon transport times.

Finally, since powerful computing resources are not only available, a new lightweight algorithm for dose reconstruction was developed for specific applications in pre-treatment dose verification. Utilizing pre-calculated MC calculated doselets in a spherical water phantom and EPID measurements, various symmetries were leveraged to attain fast dose reconstruction. This was called the spherical doselet modulation (SDM) method. For 7 IMRT plans, the average γ-index test result with 2%/2 mm

criteria above the 2% isodose was 98.8%, and the average RMSD was 1.6%. The dose reconstruction time *per EPI* was 96 seconds using a single CPU-core, and varied very little between cases. The SDM method is highly parallelizable, as it is effectively a series of image rotations, and could be efficiently implemented on the GPU in the future.

Bibliography

[1] Aapm report no. 87. diode in vivo dosimetry for patients receiving external beam radiation therapy tg 62. *Medical Physics Publishing, Madison*, 2005.

[2] W Ansbacher. Three-dimensional portal image-based dose reconstruction in a virtual phantom for rapid evaluation of imrt plans. *Med. Phys.*, 33(9):3369–82, 2006.

[3] A Badal and A Badano. Accelerating monte carlo simulations of photon transport in a voxelized geometry using a massively parallel graphics processing unit. *Medical physics*, 36(11):4878–4880, 2009.

[4] A Bakai, M Alber, and F Nusslin. A revision of the gamma-evaluation concept for the comparison of dose distributions. *Physics in Medicine and Biology*, 48(21):3543–3553, 2003.

[5] M J Berger. Monte carlo calculation of the penetration and diffusion of fast charged particles. *Methods Comput. Phys.*, 1:135–215, 1963.

[6] A Berman, R W Townson, K Bush, and S Zavgorodni. Phase space modulation method for epid-based monte carlo dosimetry of imrt and rapidarc plans. In *IC3DDose - 6th International Conference on 3D Radiation Dosimetry*, pages 112–116, 2010.

[7] H A Bethe. Molière's theory of multiple scattering. *Phys. Rev.*, 89:1256–66, 1953.

[8] W Bogdanich. Radiation offers new cures, and ways to do harm, 2010.

[9] T Bortfeld and S Webb. Single-arc imrt? *Phys. Med. Biol.*, 54(1):N9–N20, 2008.

[10] A Boyer and E Mok. A photon dose distribution model employing convolution calculations. *Med. Phys.*, 12(2):169–77, 1985.

[11] L Brualla and W Sauerwein. On the efficiency of azimuthal and rotational splitting for monte carlo simulation of clinical linear accelerators. *Radiation Physics and Chemistry*, 79(9):929–932, 2010.

[12] K Bush, I M Gagne, S Zavgorodni, W Ansbacher, and W Beckham. Dosimetric validation of acuros (r) xb with monte carlo methods for photon dose calculations. *Medical physics*, 38(4):2208–21, 2011.

[13] K Bush, I A Popescu, and S Zavgorodni. A technique for generating phase-space-based monte carlo beamlets in radiotherapy applications. *Physics in Medicine and Biology*, 53(18):N337–N347, 2008.

[14] K Bush, R W Townson, and S Zavgorodni. Monte carlo simulation of rapidarc radiotherapy delivery. *Physics in Medicine and Biology*, 53(19):N359–N370, 2008.

[15] K Bush, S Zavgorodni, and W Beckham. Inference of the optimal pretarget electron beam parameters in a monte carlo virtual linac model through simulated annealing. *Medical physics*, 36(6):2309–19, 2009.

[16] K Bush, S F Zavgorodni, and W A Beckham. Azimuthal particle redistribution for the reduction of latent phase-space variance in monte carlo simulations. *Physics in Medicine and Biology*, 52(14):4345–60, 2007.

[17] R Capote. Iaea nuclear and atomic data for medical applications: Phase-space database for external beam radiotherapy nuclear data for heavy charged-particle radiotherapy. *Radiotherapy and Oncology*, 84:S217–S217, 2007.

[18] G Cranmer-Sargison, W Beckham, and I Popescu. Modelling an extreme water-lung interface using a single pencil beam algorithm and the monte carlo method. *Phys. Med. Biol.*, 49(8):1557–67, 2004.

[19] J Deng, S B Jiang, A Kapur, J S Li, T Pawlicki, and C M Ma. Photon beam characterization and modelling for monte carlo treatment planning. *Phys. Med. Biol.*, 45(2):411–427, 2000.

[20] A O Ezzati and M Sohrabpour. Analysis of latent variance reduction methods in phase space monte carlo calculations for 6, 10 and 18 mv photons by using mcnp code. *Nuclear Instruments and Methods in Physics Research*, A(701):93–8, 2013.

[21] Gregory A. Failla, Todd Wareing, Yves Archambault, and Stephen Thompson. Acuros xb advanced dose calculation for the eclipse^{TM} treatment planning system. *Varian Medical Systems Clinical Perspecrives*, 2014.

[22] V Feygelman, G Zhang, C Stevens, and B E Nelms. Evaluation of a new vmat qa device, or the "x" and "o" array geometries. *Journal of Applied Clinical Medical Physics*, 12(2):146–168, 2011.

[23] M Fippel, F Haryanto, O Dohm, F Nusslin, and S Kriesen. A virtual photon energy fluence model for monte carlo dose calculation. *Med. Phys.*, 30(3):301–311, 2003.

[24] M K Fix, P J Keall, K Dawson, and J V Siebers. Monte carlo source model for photon beam radiotherapy: photon source characteristics. *Med. Phys.*, 31(11):3106–21, 2004.

[25] A Fogliata, G Nicolini, E Vanetti, A Clivio, and L Cozzi. Dosimetric validation of the anisotropic analytical algorithm for photon dose calculation: fundamental characterization in water. *Phys. Med. Biol.*, 51(6):1421–38, 2006.

[26] I M Gagne, W Ansbacher, S Zavgorodni, C Popescu, and W A Beckham. A monte carlo evaluation of rapidarc dose calculations for oropharynx radiotherapy. *Phys. Med. Biol.*, 53(24):7167–85, 2008.

[27] I M Gagne and S Zavgorodni. Evaluation of the analytical anisotropic algorithm in an extreme water–lung interface phantom using monte carlo dose calculations. *J. Appl. Clin. Med. Phys.*, 2007.

[28] J Gardner, J Siebers, and I Kawrakow. Dose calculation validation of vmc++ for photon beams. *Medical physics*, 34(5):1809–1818, 2007.

[29] J Godart, E W Korevaar, R Visser, D J L Wauben, and A A van't Veld. Reconstruction of high-resolution 3d dose from matrix measurements: error detection capability of the compass correction kernel method. *Phys. Med. Biol.*, 56(15):5029–43, 2011.

[30] S A Goudsmit and J L Saunderson. Multiple scattering of electrons. *Phys. Rev.*, 57:24–9, 1940.

[31] S A Goudsmit and J L Saunderson. Multiple scattering of electrons ii. *Phys. Rev.*, 58:36–42, 1940.

[32] Y J Graves, X Jia, and S B Jiang. Effect of statistical fluctuation in monte carlo based photon beam dose calculation on gamma index evaluation. *Phys. Med. Biol.*, 58:1839–53, 2013.

[33] P B Greer. 3d epid based dosimetry for pre-treatment verification of vmat—methods and challenges. *J. Phys.: Conf. Ser.*, 444:012010, 2013.

[34] P B Greer and C C Popescu. Dosimetric properties of an amorphous silicon electronic portal imaging device for verification of dynamic intensity modulated radiation therapy. *Med. Phys.*, 30:1618–27, 2003.

[35] D Grofsmid, M Dirkx, H Marijnissen, E Woudstra, and B Heijmen. Dosimetric validation of a commercial monte carlo based imrt planning system. *Medical Physics*, 37(2):540–9, 2010.

[36] F Hasenbalg, H Neuenschwander, R Mini, and E J Born. Collapsed cone convolution and analytical anisotropic algorithm dose calculations compared to vmc++ monte carlo simulations in clinical cases. *Phys. Med. Biol.*, 52(13):3679–91, 2007.

[37] S Hissoiny, B Ozell, H Bouchard, and P Despres. Gpumcd: A new gpu-oriented monte carlo dose calculation platform. *Medical physics*, 38(2):754–64, 2011.

[38] L Jahnke, J Fleckenstein, F Wenz, and J Hesser. Gmc: a gpu implementation of a monte carlo dose calculation based on geant4. *Physics in Medicine and Biology*, 57(5):1217–1229, 2012.

[39] X Jia, X Gu, Y J Graves, M Folkerts, and S B Jiang. Gpu-based fast monte carlo simulation for radiotherapy dose calculation. *Physics in Medicine and Biology*, 56(22):7017–31, 2011.

[40] X Jia, X Gu, J Sempau, D Choi, A Majumdar, and S B Jiang. Development of a gpu-based monte carlo dose calculation code for coupled electron-photon transport. *Physics in Medicine and Biology*, 55(11):3077–86, 2010.

[41] I Kawrakow. *VMC++, electron and photon Monte Carlo calculations optimized for radiation treatment planning.* SPRINGER-VERLAG BERLIN, BERLIN; HEIDELBERGER PLATZ 3, D-14197 BERLIN, GERMANY, 2001.

[42] I Kawrakow and A F Bielajew. On the condensed history technique for electron transport. *Nucl. Instrum. Methods*, B(142):253–80, 1998.

[43] I Kawrakow and M Fippel. Vmc++, a mc algorithm optimized for electron and photon beam dose calculations for rtp. *Proceedings of the 22nd Annual International Conference of the Ieee Engineering in Medicine and Biology Society, Vols 1-4*, 22:1490–1493, 2000.

[44] I Kawrakow and D W O Rogers. The egsnrc code system: Monte carlo simulation of electron and photon transport. *Technical Report PIRS-701 (4th printing) NRC Canada*, 2003.

[45] I Kawrakow, D W O Rogers, and B R B Walters. Large efficiency improvements in beamnrc using directional bremsstrahlung splitting. *Medical physics*, 31(10):2883–2898, 2004.

[46] I Kawrakow and B R B Walters. Efficient photon beam dose calculations using dosxyznrc with beamnrc. *Medical physics*, 33(8):3046–56, 2006.

[47] P J Keall, A J V Siebers, A M Arnfield, A J O Kim, and A R Mohan. Monte carlo dose calculations for dynamic imrt treatments. *Phys. Med. Biol.*, 46:929, 2001.

[48] T Knoos, A Ahnesjo, P Nilsson, and L Weber. Limitations of a pencil beam approach to photon dose calculations in lung tissue. *Phys. Med. Biol.*, 40(9):1411–20, 1995.

[49] J J Kruse. On the insensitivity of single field planar dosimetry to imrt inaccuracies. *Medical physics*, 37(6):2516–24, 2010.

[50] H W Lewis. Multiple scattering in an infinite medium. *Phys. Rev.*, 78:526–9, 1950.

[51] D W Litzenberg, J M Moran, and B A Fraass. Verification of dynamic and segmental imrt delivery by dynamic log file analysis. *Journal of applied clinical medical physics / American College of Medical Physics*, 3(2):63–72, 2002.

[52] J Llacer, S Li, N Agazaryan, C Promberger, and T D Solberg. Non-coplanar automatic beam orientation selection in cranial imrt: a practical methodology. *Phys. Med. Biol.*, 54:1337–68, 2009.

[53] D Low, W Harms, S Mutic, and J Purdy. A technique for the quantitative evaluation of dose distributions. *Medical Physics*, 25:656–61, 1998.

[54] C M Ma. Characterization of computer simulated radiotherapy beams for monte-carlo treatment planning. *Radiation Physics and Chemistry*, 53(3):329–44, 1998.

[55] T Mackie, T Holmes, S Swerdlo, P Reckwerdt, J Deasy, J Yang, B Paliwal, and T Kinsella. Tomotherapy: a new concept for the delivery of dynamic conformal radiotherapy. *Med Phys*, 20(6):1709–19, 1993.

[56] T Mackie, J Scrimger, and J Battista. A convolution method of calculating dose for 15-mv x rays. *Med. Phys.*, 12(2):188–96, 1985.

[57] A Mans, M Wendling, L N McDermott, J J Sonke, R Tielenburg, R Vijlbrief, B Mijnheer, M van Herk, and J C Stroom. Catching errors with in vivo epid dosimetry. *Medical physics*, 37(6):2638–44, 2010.

[58] R Mohan, C Chui, and L Lidofsky. Differential pencil beam dose computation model for photons. *Med. Phys.*, 13(1):64–73, 1986.

[59] G Z Molière. Theorie der streuung schneller geladener teilchen. ii. mehrfach- und vielfachstreuung z. naturforsch. *Z. Naturforsch.*, 3a:78–97, 1948.

[60] G Nicolini, E Vanetti, A Clivio, A Fogliata, S Korreman, J Bocanek, and L Cozzi. The glaas algorithm for portal dosimetry and quality assurance of rapidarc, an intensity modulated rotational therapy. *Radiot*, 3(24), 2008.

[61] Canadian Cancer Society's Advisory Committee on Cancer Statistics. Canadian cancer statistics 2013. *Toronto: ON: Canadian Cancer Society*, 2013.

[62] K Otto. Volumetric modulated arc therapy: Imrt in a single gantry arc. *Med Phys*, 35(1):310–7, 2008.

[63] V Panet-Raymond, W Ansbacher, S Zavgorodni, B Bendorffe, A Nichol, P T Truong, W Beckham, and M Vlachaki. Coplanar versus noncoplanar intensity-modulated radiation therapy (imrt) and volumetric-modulated arc therapy

(vmat) treatment planning for fronto-temporal high-grade glioma. *J. Appl. Clin. Med. Phys.*, 13:44–53, 2012.

[64] I A Popescu, C P Shaw, S F Zavgorodni, and W A Beckham. Absolute dose calculations for monte carlo simulations of radiotherapy beams. *Physics in Medicine and Biology*, 50(14):3375–92, 2005.

[65] G Pratx and L Xing. Gpu computing in medical physics: A review. *Medical physics*, 38(5):2685–97, 2011.

[66] A Pugachev, J Li, A Boyer, S Hancock Q, Le, S Donaldson, and L Xing. Role of beam orientation optimization in intensity-modulated radiation therapy. *Int. J. Radiat. Oncol. Biol. Phys.*, 50:551–60, 2001.

[67] M A Renaud, D Roberge, and J Seuntjens. Latent uncertainties of the precalculated track monte carlo method. *Med Phys.*, 42(1):479–90, 2015.

[68] M Rodriguez, J Sempau, and L Brualla. A combined approach of variance-reduction techniques for the efficient monte carlo simulation of linacs. *Phys. Med. Biol.*, 57(10):3013–24, 2012.

[69] D W O Rogers, B A Faddegon, G X Ding, C M Ma, J We, and T R Mackie. Beam - a monte-carlo code to simulate radiotherapy treatment units. *Medical physics*, 22(5):503–24, 1995.

[70] E Salvat, J M Fernandez-Varea, and J Sempau. Penelope-2011: A code system for monte carlo simulation of electron and photon transport. *Nuclear Energy Agency*, NEA/NSC/DOC(2011)5, Workshop Proceedings, Barcelona, Spain, July 4-7, 2011.

[71] F Salvat, J M Fernandez-Varea, J Baro, and J Sempau. Penelope, an algorithm and computer code for monte carlo simulation of electron-photon showers. *University of Barcelona preprint*, 1996.

[72] S Saminathan, R Manickam, V Chandraraj, and S S Supe. Dosimetric study of 2d ion chamber array matrix for the modern radiotherapy treatment verification. *Journal of applied clinical medical physics / American College of Medical Physics*, 11(2):3076–76, 2010.

[73] D Schmidhalter, P Manser, D Frei, W Volken, and M K Fix. Comparison of monte carlo collimator transport methods for photon treatment planning in radiotherapy. *Med. Phys.*, 37(2):492–504, 2010.

[74] J Sempau and A F Bielajew. Towards the elimination of monte carlo statistical fluctuations from dose volume histograms for radiotherapy treatment planning. *Phys. Med. Biol.*, 45:131–57, 2000.

[75] J Sempau, S J Wilderman, and A F Bielajew. Dpm, a fast, accurate monte carlo code optimized for photon and electron radiotherapy treatment planning dose calculations. *Phys. Med. Biol.*, 45(8):2263–91, 2000.

[76] S C Sharma, J T Ott, J B Williams, and D Dickow. Clinical implications of adopting monte carlo treatment planning for cyberknife. *Journal of Applied Clinical Medical Physics*, 11(1), 2010.

[77] J V Siebers, P J Keall, J O Kim, and R Mohan. A method for photon beam monte carlo multileaf collimator particle transport. *Phys. Med. Biol.*, 47:3225, 2002.

[78] J V Siebers, P J Keall, B Libby, and R Mohan. Comparison of egs4 and mcnp4b monte carlo codes for generation of photon phase space distributions for a varian 2100c. *Phys. Med. Biol.*, 44(12):3009–26, 1999.

[79] A M Stell, J G Li, O A Zeidan, and J F Dempsey. An extensive log-file analysis of step-and-shoot intensity modulated radiation therapy segment delivery errors. *Medical physics*, 31(6):1593–602, 2004.

[80] T Teke, A M Bergman, W Kwa, B Gill, C Dunzenli, and I A Popescu. Monte carlo based, patient- specific rapidarc qa using linac log files. *Med. Phys.*, 37:116–23, 2010.

[81] D Terribilini, M K Fix, D Frei, W Volken, and P Manser. Vmc++ validation for photon beams in the energy range of 20–1000 kev. *Medical Physics*, 37(10):5218–27, 2010.

[82] D Thain, T Tannenbaum, and M Livny. Distributed computing in practice: the condor experience. *Concurrency and Computation: Practice and Experience*, 17(2-4):323–356, 2005.

[83] J Thebaut and S Zavgorodni. Coordinate transformations for beam/egsnrc monte carlo dose calculations of non-coplanar fields received from a dicom-compliant treatment planning system. *Physics in Medicine and Biology*, 51(23):N441–N449, 2006.

[84] R W Townson, X Jia, Z Tian, Y J Graves, S Zavgorodni, and S B Jiang. Gpu-based monte carlo radiotherapy dose calculation using phase-space sources. *Physics in Medicine and Biology*, 58(12):4341–4356, 2013.

[85] R W Townson and S Zavgorodni. A hybrid phase-space and histogram source model for gpu-based monte carlo radiotherapy dose calculation. *Phys. Med. Biol.*, 59(24):7919, 2014.

[86] W Ulmer, J Pyyry, and W Kaissl. A 3d photon superposition/convolution algorithm and its foundation on results of monte carlo calculations. *Phys. Med. Biol.*, 2005.

[87] W van Elmpt, L McDermott, S Nijsten, M Wendling, P Lambin, and B Mijnheer. A literature review of electronic portal imaging for radiotherapy dosimetry. *Radiotherapy and Oncology*, 88(3):289–309, 2008.

[88] W van Elmpt, S Nijsten, B Mijnheer, A Dekker, and P Lambin. The next step in patient-specific qa: 3d dose verification of conformal and intensity-modulated rt based on epid dosimetry and monte carlo dose calculations. *Radiother. Oncol.*, 86:86–92, 2008.

[89] A van Esch, C Clermont, M Devillers, M Iori, and D P Huyskens. On-line quality assurance of rotational radiotherapy treatment delivery by means of a 2d ion chamber array and the octavius phantom. *Medical physics*, 34(10):3825–37, 2007.

[90] M van Zijtveld, M Dirkx, M Breuers, H de Boer, and B Heijmen. Portal dose image prediction for in vivo treatment verification completely based on epid measurements. *Medical physics*, 36(3):946–52, 2009.

[91] B Vanderstraeten, N Reynaert, L Paelinck, I Madani, C D Wagter, W D Gersem, W D Neve, and H Thierens. Accuracy of patient dose calculation for lung imrt: A comparison of monte carlo, convolution/superposition, and pencil beam computations. *Med Phys*, 33(9):3149–58, 2006.

[92] F Verhaegen, R Symonds-Tayler, H H Liu, and A E Nahum. Backscatter towards the monitor ion chamber in high-energy photon and electron beams: charge integration versus monte carlo simulation. *Phys. Med. Biol.*, 45:3159–70, 2000.

[93] A E S von Wittenau, L J Cox, P M Bergstrom, W P Chandler, C L H Siantar, and R Mohan. Correlated histogram representation of monte carlo derived medical accelerator photon-output phase space. *Medical physics*, 26(7):1196–211, 1999.

[94] B R B Walters and D W O Rogers. Dosxyznrc users manual. *(Ottawa: National Research Council of Canada) PIRS-794*, 2003.

[95] X Wang, X Zhang, L Dong, H Liu, M Gillin, A Ahamad A, and R Moran. Effectiveness of noncoplanar imrt planning using a parallelized multiresolution beam angle optimization method for paranasal sinus carcinoma. *Int. J. Radiat. Oncol. Biol. Phys.*, 63:594–601, 2005.

[96] E Wieslander and T Knoos. A virtual linear accelerator for verification of treatment planning systems. *Phys. Med. Biol.*, 45(10):2887–96, 2000.

[97] L Xing, L Lee, and R Timmerman. Adaptive radiation therapy and clinical perspectives. *Image Guided and Adaptive Radiation Therapy*, pages 16–40, 2009.

[98] I J Yeo, J Won Jung, M Chew, J Oh Kim, B Wang, S DiBiase, Y Zhu, and D Lee. Dose reconstruction for intensity-modulated radiation therapy using a non-iterative method and portal dose image. *Physics in Medicine and Biology*, 54(17):5223–36, 2009.

[99] S Zavgorodni, E Alhakeem, and R Townson. Monitor backscattering factors for the varian 21ex and truebeam linear accelerators: measurements and monte carlo modelling. *Phys. Med. Biol.*, 59:911, 2014.

[100] S Zavgorodni, K Bush, C Locke C, and W Beckham. Vancouver island monte carlo (vimc) system for radiotherapy treatment planning dosimetry and research. *Radiother. Oncol.*, 84:S49, 2007.

[101] T C Zhu, A Ahnesjo, K L Lam, X A Li, C C Ma, J R Palta, and R C Tailor. Report of aapm therapy physics committee task group 74: in-air output ratio, s-c, for megavoltage photon beams. *Med. Phys.*, 36:5261–91, 2009.

[102] J Zimmerman, S Korreman, G Persson, H Cattell, M Svatos, A Sawant, R Venkat, D Carlson, and P Keall. Dmlc motion tracking of moving targets for intensity modulated arc therapy treatment: a feasibility study. *Acta Oncologica*, 48(2):245–50, 2009.

Appendix A

The VIMC streamlined system and WebMC interface

Monte Carlo simulations for dose calculations of radiotherapy involve a plethora of interconnected software packages, each of which may be the product of years, or even decades of development by numerous research groups. As such, the level of expertise required for high-level interaction with the various components of a MC system is quite substantial. At VIC, a streamlined MC framework was developed to provide users with a simplified interface, with default parameters selected for the most common applications. This was called the Vancouver Island Monte Carlo (VIMC) system (Bush *et al.*, 2008 [14] and Zavgorodni *et al.*, 2007 [100]).

The VIMC framework was originally developed using the TCL/TK programming languages, but in 2007 I ported the interface component to PHP and Javascript. I have been a primary developer of VIMC since that time, and took over the system administration role from Karl Bush in 2010. Some of the basic VIMC functionalities include, management of accelerator models, management of patient treatment plans for import/export from VIMC, configuration and submission of MC calculations, and dose visualization:

- Accelerator management
  - Add new accelerator models
  - Copy & edit existing models
  - Configure other software packages independently per accelerator
    - BEAMnrc

    * PhspC
    * vcuDMLCcode
    * DOSXYZnrc
    * VMC++

- Patient management
  - Add new patients by uploading DICOM files from the TPS
  - Review patient data before and after MC simulations
  - Download results
- Streamlined MC simulations
  - Select the plan & type of simulation to perform
  - Select the accelerator model, dose engine, etc.
  - Create a virtual patient phantom
  - Specify uncertainty requirements
  - Select the computing resource
  - Submit the job to a queue & monitor completion status
- Dose visualization
  - Review the MC dose results slice-by-slice
  - Generate dose profile curves

To enable the work in this dissertation, this system was highly utilized and modified. In particular, the phase-space collimation method of chapter 3 and the SDM method of chapter 7 were integrated into VIMC. This chapter provides functional outlines of these technologies, as well as a suggested strategy for future integration of GPU-based MC calculations using gDPM.

A.1 VIMC and the WebMC interface

The VIMC system refers to the entire framework of servers, software and general process for performing MC simulations at VIC. A typical MC calculation begins with

the user exporting the data files for a plan (in the DICOM standard medical imaging format) from the TPS on the Provincial Health Services Authority (PHSA) file-server (figure A.1). The exported data typically includes the treatment plan configuration (DICOM-RP), CT images (DICOM-CT), anatomical structure contours (DICOM-RS) and TPS dose distributions (DICOM-RD). The user uploads these files to the WebMC server using the WebMC online interface, accessible from any computer inside the PHSA network firewall. The WebMC interface is also used to configure the MC calculation (section A.1.1) and submit the job to a computing resource. The 'frontend node' of the computing resource (or computing cluster) automatically handles the workload, queueing jobs to the 'worker nodes' in first-come-first-serve order. When the calculations are complete, results are copied from the individual worker back to the WebMC server. At this stage, the user is able to access the results through the WebMC interface and copy the data back to the PHSA file-server for analysis in the TPS. Alternatively, the user can observe dose distributions and produce dose curves using the WebMC interface.

A.1.1 Performing a MC calculation

The WebMC system supports a number of different types of MC calculations. After uploading the plan data (a simple process that is not illustrated here), the user navigates to the *Submission* tab in WebMC (figure A.2). The most flexible dose calculation option, providing the user with a variety of configuration parameters to select, is an *Advanced MC Submission*. This section will briefly describe the main components of an *Advanced MC Submission* for comparison with the new options developed for this dissertation.

The simulation configuration page (figure A.3) allows for the selection of the accelerator model and dose calculation engine, among other options. Set-up then proceeds to the MC phantom configuration (figure A.4). This settings available on this page depend on the 'Patient Phantom' option that was selected. For 'Patient Phantom' set to 'CT DICOM', a MC phantom is created from the CT DICOM files exported from the TPS. The user can select the voxel dimensions for the phantom, as well as perform material replacement operations inside or outside of contours (as defined by the structures in the RS DICOM file from the TPS).

The next step in the set-up process allows the user to set the number of particles to simulate using built-in uncertainty estimation (figure A.5). The user can also choose

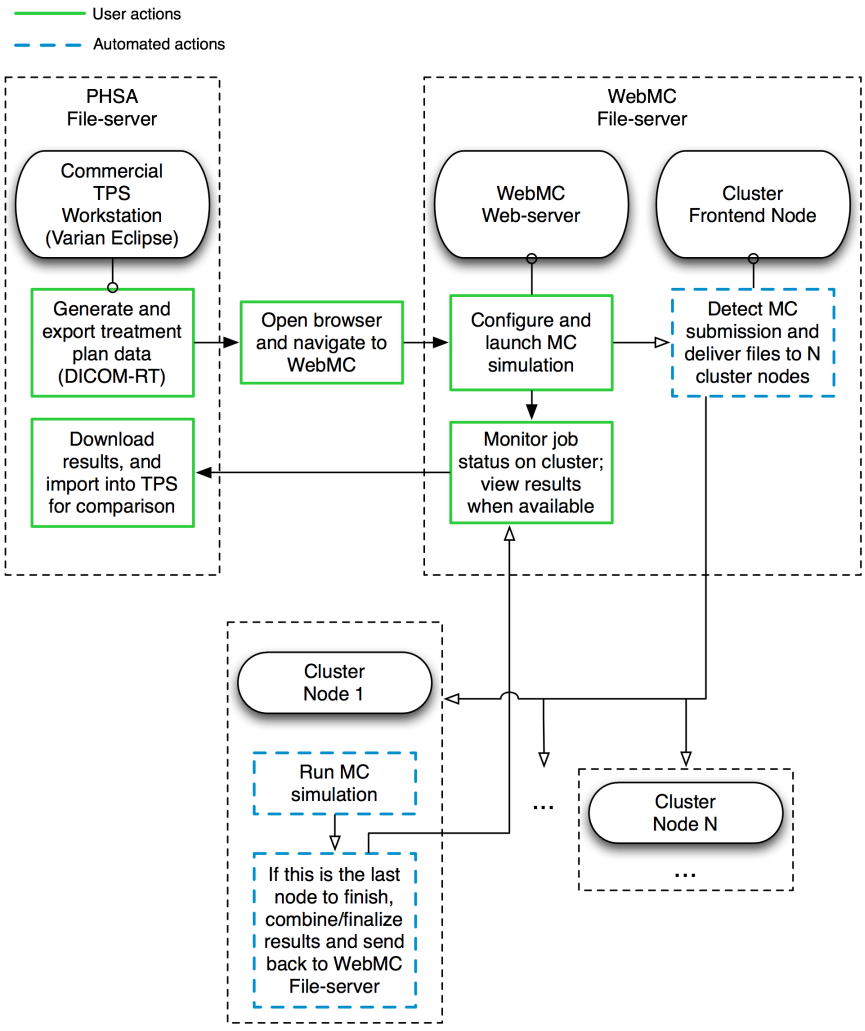

Figure A.1: A flowchart of the VIMC system for performing a typical MC simulation. The user exports a treatment plan from the TPS, uploads it to the WebMC server, configures the simulation and submits the job to the computing resource. The computing servers then distribute the workload and perform calculations, returning the result to the WebMC server when complete. The user then imports the result into the TPS for analysis.

to 'split' fields, which enhances the parallelization of the calculations by creating duplicates of each field. For example, a splitting number of 10 means that each field is split into 10 identical sub-fields, each simulating a tenth of the particles needed to achieve the reqested uncertainty for the field. After simulation but prior to absolute dose conversion, the results of the sub-fields are cumulated.

Finally, the last set-up page (figure A.6) provides a review of the simulation pa-

rameters, and an option to select which computing resource is to be used for the calculations. After submitting the job, the job submission data is placed in a staging directory specific to the selected computing resource. The front-end node of the computing resource monitors this directory, and new submissions are automatically queued to await distribution to the worker nodes for calculations to proceed.

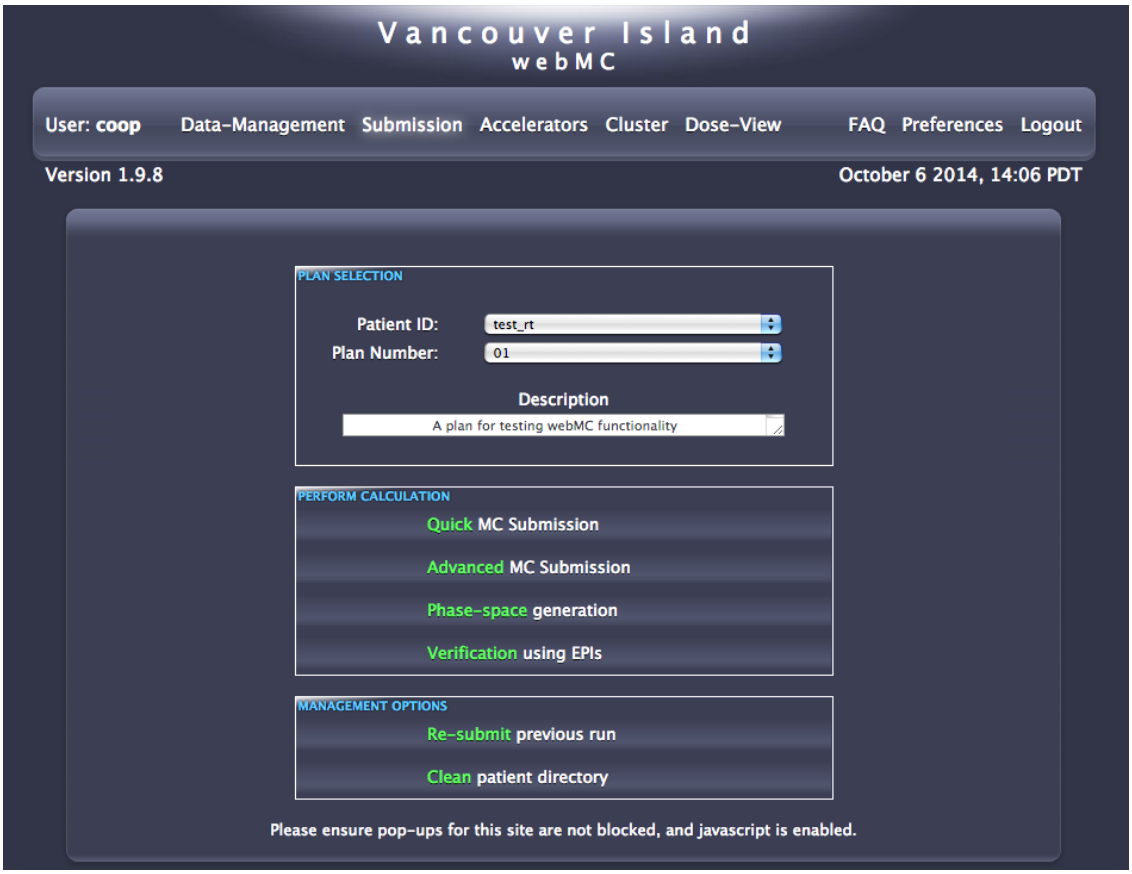

Figure A.2: A screenshot of the main WebMC page where the type of simulation is selected.

A.1.2 Quick Monte Carlo

The *Quick MC Simulation*, or QuickMC option in figure A.2 leads the user to a simplified configuration for MC simulations. In this mode (shown in figure A.7), a series of default options are selected automatically to with the intention of providing fast simulation times with minimal user interaction. All of the set-up is performed on a single page with limited options compared to an *Advanced MC Submission*. The phase-space source (PhspA) is automatically selected from those available for clinical

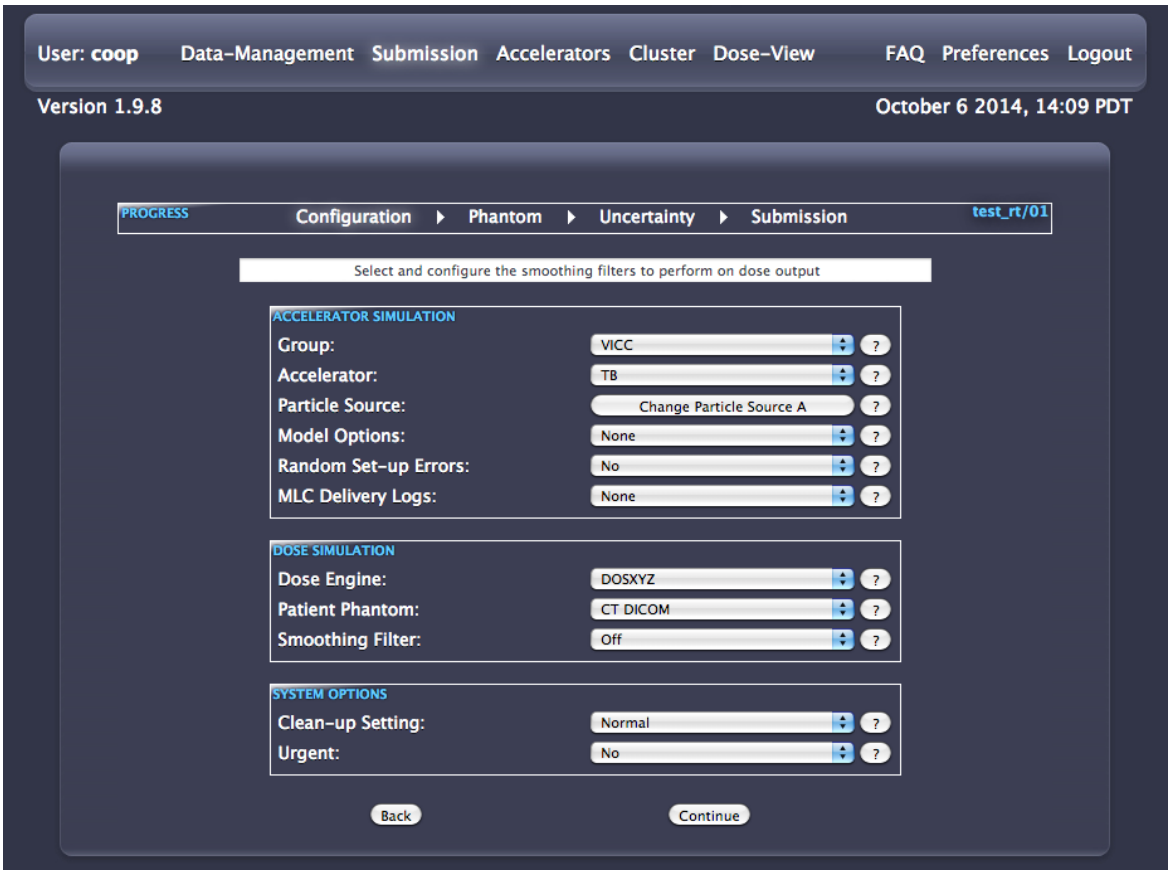

Figure A.3: A screenshot of the Advanced MC Submission configuration page.

use at VICC. The user can select whether to use MLC motions from the RP DICOM plan file or MLC delivery logs (such as dynalog files). The dose engine defaults the fastest MC engine currently available, VMC++. Phantom construction requires that CT DICOM files are available, and the standard options for voxel dimensions and contour fills are available. Finally, field splitting, estimated uncertainty requirements, file clean-up options, calculation priority and computing resource can also be selected.

One of the most important aspects of this mode, that allows it to obtain fast calculation times, is the implementation of the PhspC method from chapter 3. This avoids simulating accelerator head models using BEAMnrc, instead using the simple secondary collimator model of PhspC. The template input format for PhspC is as follows[1]:

```
--FormatEGS
--PhspFile ${PhspFile}
```

[1]For more description of the PhspC input parameters, run PhspC with the argument '–help'.

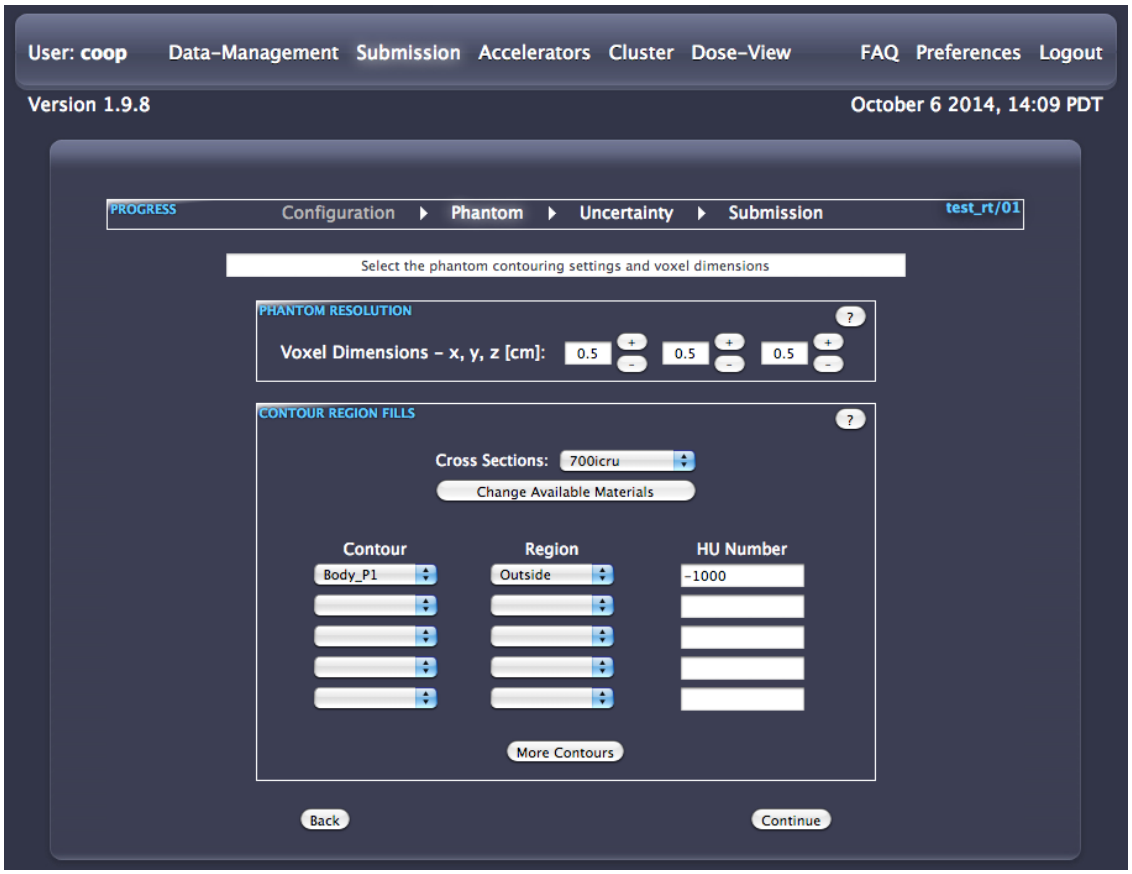

Figure A.4: A screenshot of the Advanced MC Submission phantom contouring page.

```
 --PhspZ ${PhspZ}
 --FinalZ ${FinalZ}
 --YJawZ ${YJawZ}
--XJawZ ${XJawZ}
--X1 ${X1}
--X2 ${X2}
--Y1 ${Y1}
--Y2 ${Y2}
--NumRecycle ${NumRecycle}
--OutputPrefix ${OutputPrefix}
--RequestedNumParticles ${RequestedNumParticles}
--RSeed ${RSeed}
--TempEgslst ${TempEgslst}
--EgslstName ${EgslstName}
```

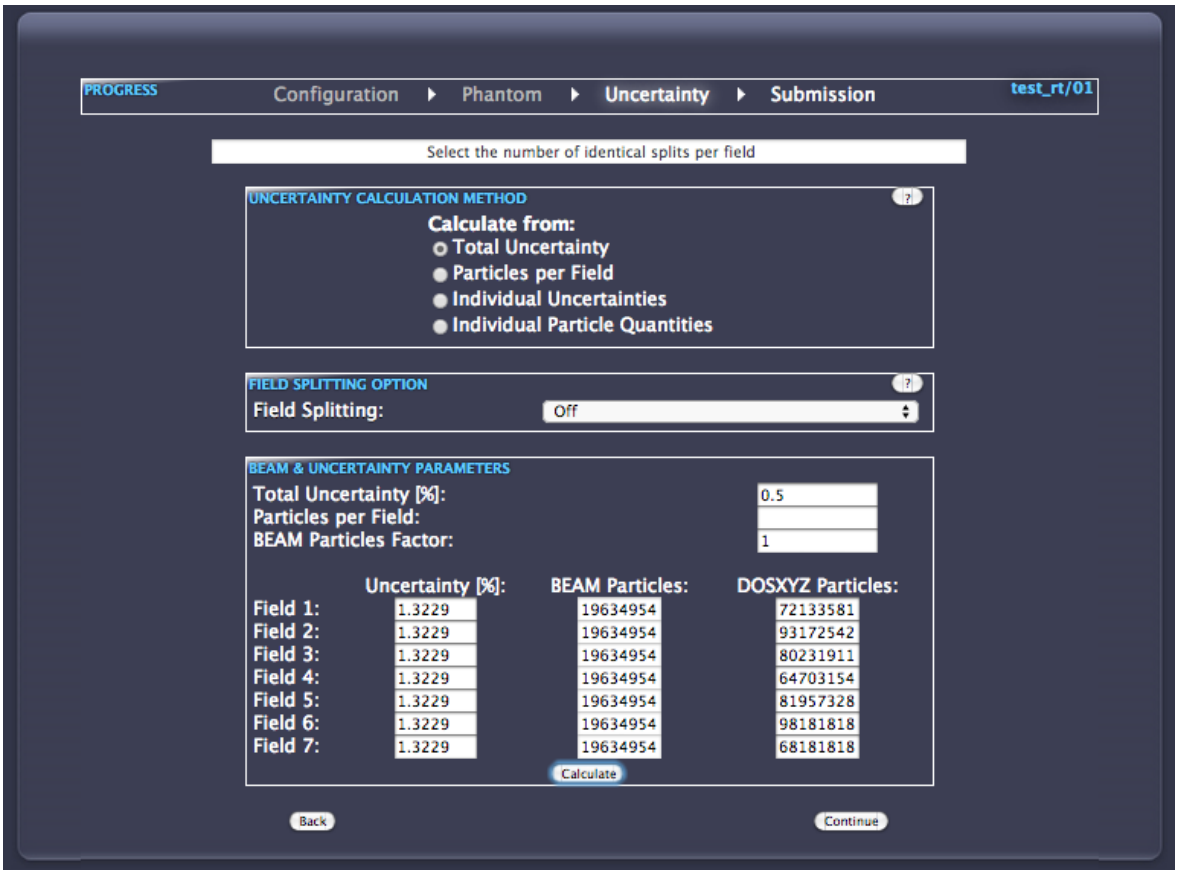

Figure A.5: A screenshot of the Advanced MC Submission uncertainty estimation page.

```
--FileSTT ${FileSTT}
```

A.1.3 Dose verification using the SDM method

The SDM method was integrated into WebMC in the *Verification Using EPIs* section. Currently there is only one other dose verification method supported - the PSM method. As shown in figure A.8, the user selects the portal images, including the flood, calibration and dark fields for the treatment plan. The treatment technique has no effect for the SDM method. In WebMC the SDM method is also referred to as DoseletMC. Since the same set of doselets are reused for all verification calculations, there is no need to create a phantom or perform uncertainty estimation during set-up.

The RP DICOM plan file is used to obtain the treatment angles, MUs, beam energies, etc. Currently a doselet dataset has only been generated for the Varian 21EX 6MV accelerator.

Figure A.6: A screenshot of the Advanced MC Submission cluster submission page.

In the future, the versatility of this implementation could be improve by allowing the user to select between different doselet datasets that have been generated with varying resolution or phantom size. Additionally, the user may wish to independently select the resolution of the final phantom since it can be different from the initial phantom. It may even be beneficial to analyze the field sizes from the plan during set-up in order to recommend an appropriately sized phantom to match the plan requirements (i.e. recommend a larger, perhaps lower density phantom for large field sizes).

A.1.4 Future work: GPU-based dose calculations

Support for GPU-based dose calculations, specifically using gDPM v3.0, have not yet been implemented in WebMC. In this section, an outline how this could be done in

Figure A.7: A screenshot of the Quick MC Submission configuration page.

the future will be presented. This will act as a rough guide for that work, but also helps to illuminate the overall process of how gDPM MC simulations are performed.

Adding gDPM support to an accelerator

Each accelerator used by WebMC has a directory where template input files for the various dose calculation engines are stored. These templates contain the default configurations, as well as variable flags to signify portions of the file that will be replaced for each patient during set-up. Similar, to the other dose engines, gDPM will require a template input file. The format of input for gDPM v3.0 should follow a template similar to the following[2]:

```
${SOURCE}
${FLUENCE_MAP}
```

[2]For more description of the gDPM input parameters, run gDPM with the argument '–help'.

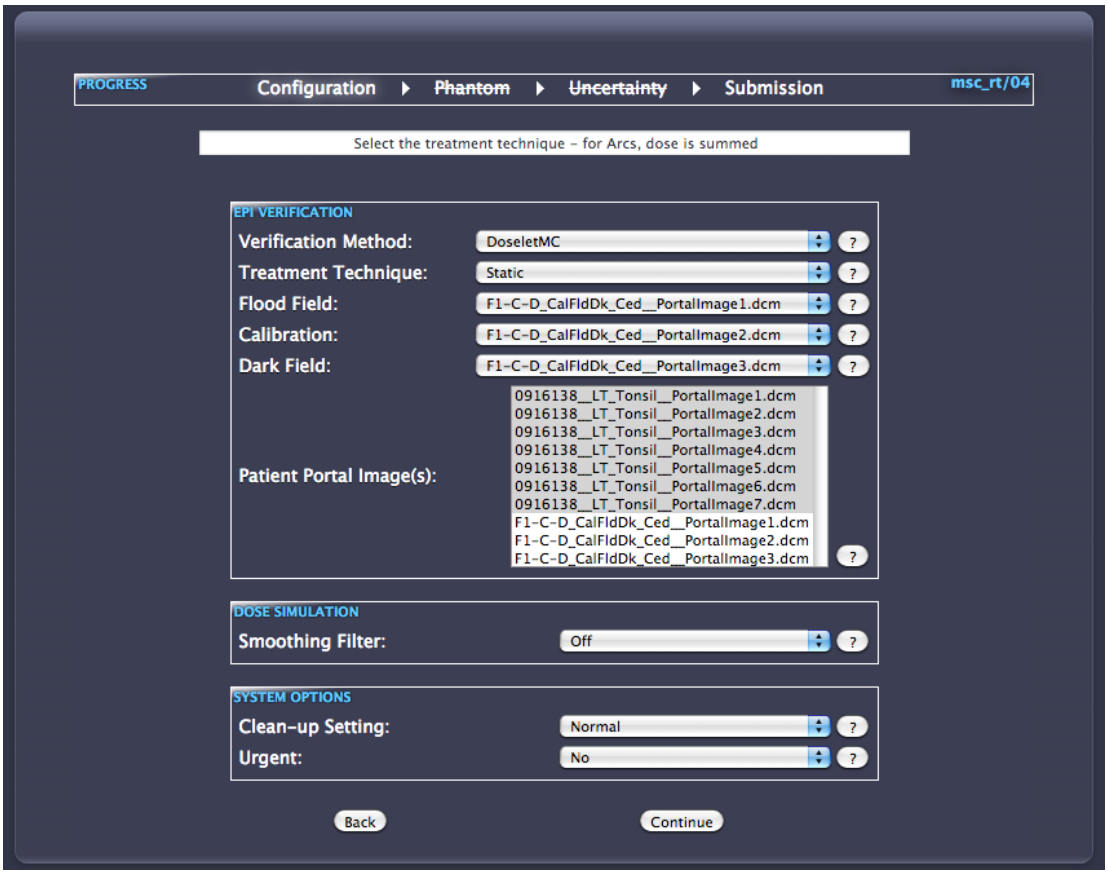

Figure A.8: A screenshot of the Verification Using EPIs configuration page.

```
--NumberOfHistories=${NCASE}
--CalibrationFactor=${CALIBRATION}
--JawsFile=${JAWS_FILE}
--BeamAnglesFile=${ANGLES_FILE}
--EGS4GeometryFile=${PHANTOM}
--CTdimension=${NVOX_X}
--CTdimension=${NVOX_Y}
--CTdimension=${NVOX_Z}
--CTresolution=${DX}
--CTresolution=${DY}
--CTresolution=${DZ}
--CToffset=${MIN_X}
--CToffset=${MIN_Y}
--CToffset=${MIN_Z}
```

```
--Isocenter=${ISOX}
--Isocenter=${ISOY}
--Isocenter=${ISOZ}
--SAD=${SAD}
--ElectronAbsorptionEnergy=200.0e3
--PhotonAbsorptionEnergy=50.0e3
--CompatibleFilePrefix=input/pre4elec
--DoseToWaterConversionFile=input/sfactor.dtw
```

${SOURCE} may be substituted depending on the type of source model that is to be used. For a hybrid source using PSLs for the extra-focal source, replace ${SOURCE} with:

```
--Photons
--SourceEnergy=${ENERGY_EV}
--SourceModelFile=${HISTOGRAM_FILE}
--HybridPhspParticleReduction=1
--PslFilePrefix=${PSL_EXTRAFOCAL_DATABASE}
```

For a PSL source, replace ${SOURCE} with:

```
--PslFilePrefix=${PSL_DATABASE}
```

For a phase-space source, replace ${SOURCE} with:

```
--PhspFile=${PHSP_FILE}
--PhspZ=${PHSP_Z}
```

${FLUENCE_MAP} is an optional parameter to allow for MLC modeling, and when used should be:

```
--FluenceMapFile=${FLUENCE_FILE}
```

Further description of the gDPM arguments and usage is available using the –help argument.

Fluence map generation for gDPM requires an additional input file for the gFluenceMap software. The template for this file should be as follows:

```
##Treatment_modality(IMRTorVMAT)
IMRT
```

```
##input_DICOM
${RP_DICOM_FILE}
##Output_fluencemap_file
${FLUENCE_FILE}
${BIN_FILE}
##Output_angle_file
${ANGLES_FILE}
##Output_parameter_file
${JAWS_FILE}
```

Necessary changes to the set-up process

The option to use gDPM as a dose calculation engine will be added in the *Advanced MC Submission* and *Quick MC Submission* sections, as an alternative to DOSXYZnrc or VMC++. Selecting this option will trigger the following changes to the set-up and execution of the simulation.

It may be useful to have additional configuration options for gDPM appear after it is selected. For example, selection of the source model (hybrid method, PSL method, PhspA method, phase-space collimation method, PhspB method). After continuing past the initial configuration page, a check to see that gDPM is supported for the selected accelerator should be performed. The phantom generation stage of set-up will remain unchanged - gDPM uses the same phantom format as DOSXYZnrc, the *egsphant* format.

The uncertainty configuration stage will require some adjustments for gDPM. The dependency of output dose uncertainty on the number of particles requested for the simulation will be different than the other dose engines. It also depends on the type of source model selected. These dependencies will need to be characterized and included in the uncertainty calculations.

Input files for the MC codes are created as the last step before submission to the computing resource queue. Since MLC modeling in gDPM is normally handled by fluence maps, the (also GPU-based) software for fluence map generation, gFluenceMap, will need to be run before gDPM. The input file for gFluenceMap should be created at this stage, along with the input file for gDPM. Of course, the different source modeling methods will have different input file requirements, so input files may also be needed for BEAMnrc, phase-space collimation, and/or the vcudmlccode if it

is being used for MLC simulation instead of gFluenceMap.

Since gDPM does not include a model of the monitor chamber, the backscatter corrections to the absolute dose of equation 2.22 can not be calculated. Instead, this backscatter correction could be included in the ${CALIBRATION} parameter of the input file after looking up a measured S_b value from a table (section 2.2.5).

For source models where no CPU-based computing resource is required, the job can be submitted directly to the GPU-based system. A queueing system such as Condor should be installed on this resource to handle simultaneous submissions sequentially (if there are multiple GPUs available, as in cavigpumc01, then jobs could be parallelized). If CPU-based computations are required (for BEAMnrc, phase-space collimation and/or vcudmlccode), then these should be performed first, and the queueing and transfer of data to the GPU system can be handled by the last CPU to complete its calculations.

Configuring the GPU-based server

Note that detailed configuration and installation guidelines will be provided in a separate manual. In general, CUDA must be installed along with a number of libraries to support the various software applications.

In order for the WebMC interface to submit a job to the GPU-based server, a job submission file of a particular format is placed in a particular directory on a shared file system. The GPU-based server must continuously monitor this directory new files, using a special daemon (software running as a background process). Once the daemon detects that a job submission has been requested, performs the actual submission to the parallel computing software, Condor (Thain *et al.* 2005 [82]). Since the GPU-based system will likely be a standalone server, Condor must be configured to run both as a submitter node (submits jobs) and a worker node (executes jobs). At a VIC, a similar configuration is used on the server *cavivake01*.

To model new accelerator models using a PSL source model (as described in section 4.1.4), it will be necessary to run the pslGenerator software distributed in the utilities directory of gDPM. To generate a PSL source from a phase-space, run the pslGenerator code with the following arguments[3]:

`--FormatEGS`

[3]For more description of the pslGenerator input parameters, run pslGenerator with the argument '–help'.

```
--PhspFile=${PHSP_FILE}
--NEnergyBins=20
--PhspZ=${PHSP_Z}
--FinalZ=${FINAL_Z}
--X1=10
--X2=10
--Y1=10
--Y2=10
--DX=0.5
--DY=0.5
--OutputPrefix=${PSL_DATABASE}
```

where ${PHSP_FILE} is the path to the EGS format phase-space file, ${PHSP_Z} is the current z-coordinate of all particles in the phase-space (EGS format phase-spaces must be planar), ${FINAL_Z} is the z-coordinate of the plane where the output PSL source should reside (if unsure, use ${FINAL_Z}=${PHSP_Z}), and ${PSL_DATABASE} is the path to the output folder that will contain the PSL data files. For IAEA format input phase-spaces, replace the first two arguments with:

```
--FormatIAEA
--PhspFile=${PHSP_FILE}
--PhspHeaderFile=${PHSPHEADER_FILE}
```

where ${PHSPHEADER_FILE} is the path to the IAEA header file for the phase-space.

To generate a hybrid source (as in chapter 5), use the following input arguments to run the phspManager code[4]. Note that this creates both the histogram file and an output phase-space (for secondary particles), which can then be converted into a PSL database using pslGenerator.

```
--RemovePrimary
--SpotSize=.9
--NumSpectrumBins=20
--NumRadialBins=400
--FormatEGS
```

[4]For more description of the phspManager input parameters, run phspManager with the argument '–help'.

```
--PhspZ=${PHSP_Z}
--DisableAPR
--PhspFile=${PHSP_FILE}
--OutputPrefix=${OUTPUT_PREFIX}
```

where ${OUTPUT_PREFIX} is the path and prefix that will be shared for all output files (the rest of the filename will be automatically appended).

Note that it is possible to use input files to run pslGenerate, phspManager, and gDPM. To do this, simply store the arguments in a file ${ARG_FILE}, separated by spaces or new lines. Then, use the xargs software (installed by default) to run the software ${EXECUTABLE_FILE}. For example:

```
xargs --arg-file=${ARG_FILE} ${EXECUTABLE_FILE}
```