Increasing efficiency of BEAMnrc-simulated Co-60 beams using directional source biasing

Walters, B. R. B.

Publisher's version / Version de l'éditeur:
https://doi.org/10.1118/1.4930060
Medical Physics, 42, 10, pp. 5817-5827, 2015-09-15

Increasing efficiency of BEAMnrc-simulated Co-60 beams using directional source biasing

B.R.B. Walters*

Ionizing Radiation Standards, National Research Council of Canada, Ottawa K1A 0R6, Canada

(Dated: September 29, 2014)

Abstract

Purpose: This study describes the implementation of a directional source biasing (DSB) scheme for efficiently simulating Cobalt-60 treatment heads using the BEAMnrc Monte Carlo code. Previous simulation of Co-60 beams with BEAMnrc were impractical because of the time required to track photons not directed into the treatment field as well as secondary charged particles (> 100 hours on 15 1.8 GHz CPU's to obtain < 0.1% uncertainty on doses in 0.5 cm × 0.5 cm × 0.5 cm voxels in water).

Methods: In DSB the Co-60 source preferentially generates low-weight, primary photons directed into a splitting field that encompasses the treatment field. Photons are also split and radially-redistributed immediately below the radially-symmetric portion of the treatment head, meaning that fewer photons need to be tracked above this splitting plane. Transport in the treatment head makes use of the directional bremsstrahlung splitting scheme (with some modifications) already in place for efficient photon beam simulations in BEAMnrc.

Results: The DSB scheme in BEAMnrc increases the photon fluence efficiency in a 10x10 cm^2 Co-60 beam by a factor of 1,800 with a concurrent increase in contaminant electron fluence efficiency by a factor of 1,200. Implementation of DSB in beampp, a C++ code for accelerator simulations based on EGSnrc and the C++ class library, egspp, increases photon fluence efficiency by a factor of 2,800 and contaminant electron fluence efficiency by a factor of 1,600. Optimum splitting numbers are in the range 20,000–40,0000. For dose calculations in a water phantom (0.5×0.5×0.5 cm^3 voxels) this translates into an increased dose calculation efficiency (all doses>0.5×D_{max}) of a factor of ~40. An example calculation of dose to water/dose to chamber (the basis of the beam quality correction factor) to within 0.2% in a realistic chamber using a full simulation of a Co-60 treatment head as a source indicates the practicality of Co-60 simulations with DSB.

Conclusions: The Co-60 beam calculation which required 100 hours now requires only 2.5 hours, making Monte Carlo commissioning of Co-60 beams and calculation of beam quality and correction factors feasible.

* blakerwalters@gmail.com

I. INTRODUCTION

Ongoing interest in the Monte Carlo modeling of Cobalt-60 treatment beams stems from the possibility of accurately calculating correction factors, commissioning Co-60 beams for use in IMRT and using a Co-60 beam together with magnetic resonance imaging (MRI) in image guided radiotherapy (IGRT).

Co-60 is used as the reference source to convert ion chamber response in a megavoltage photon beam of quality, Q, to dose to water[1] by means of the beam quality conversion factor, k_Q. TG51[1] gives k_Q values for many common ion chambers in use at the time of its publication. A recent addendum to TG-51[2] gives Monte Carlo calculated k_Q values for all ion chambers currently available commercially. Continuing development of ion chambers for both commercial and custom use motivates increasing the efficiency of Monte Carlo simulations used to calculate k_Q values.

Despite the relative safety, reliability and simplicity of Co-60 as a source, for many years it was considered unsuitable for IMRT protocols because of its low penetration, large penumbra and low achievable dose rate when compared to megavoltage photon beams. Recently, however, there have been reevaluations of Co-60 for IMRT. Joshi et al[3] presented a design, based on BEAMnrc Monte Carlo simulations, for an efficient Co-60 head for use in tomotherapy. Analysing measured dose volume histograms for IMRT treatment plans, Fox et al[4] concluded that Co-60 could provide similar target coverage and critical organ sparing to a 6 MV source. Measurements by Schreiner et al[5] also indicate the suitability of Co-60 as a source for tomotherapy.

Unlike megavoltage photon beams, Co-60 sources do not require magnetic beam directing systems. Thus, with its establishment as a viable source for IMRT, it was not long before Co-60 was proposed as a source in image guided radiotherapy (IGRT) using magnetic resonance imaging (MRI)[4–7]. MRI offers considerable improvements in soft tissue contrast over other imaging modalities. A clinical MRI-guided, Co-60 based IGRT system has recently been developed by ViewRay (Oakwood Village, OH, USA) and has been used to treat patients since early 2014 at Washington University and, more recently, at the University of Wisconsin. The system has also recently been acquired by cancer centres at the University of Miami and UCLA. Should such units achieve widespread clinical use, an increased interest in Monte Carlo commissioning of Co-60 beams is anticipated.

BEAMnrc[8], a powerful tool for simulating medical accelerators based on the EGSnrc Monte Carlo package[9], has been used by Mora et al[10] to accurately characterize Co-60 beams, however simulation times to achieve precise results are too long to be of practical use. For example, using Mora et al's simulation parameters for a 10×10 cm^2 beam incident on a 21 cm $\times$ 21 cm $\times$ 30 cm water phantom (0.5 cm $\times$ 0.5 cm $\times$ 0.5 cm voxels) requires over 100 hrs on 15 1.8 GHz Opteron 244 CPU's to obtain 0.1% uncertainty on doses $> 0.5\times D_{max}$.

The inefficiency of Co-60 simulations is due to the fact that much of the calculation time is spent tracking photons from the Co-60 source not directed into the treatment field and secondary electrons. This work describes a new directional source biasing (DSB) algorithm which addresses these issues by: 1) preferentially generating only those primary photons emitted by the isotropically-radiating Co-60 source that are directed into a user-defined splitting field (encompassing the treatment field), 2) using the radial symmetry of the Co-60 treatment head above the primary collimator to reduce the number of primary photons tracked above the collimator, subjecting them to further splitting and radial redistribution upon entering the collimator, and 3) reducing the number of secondary electrons tracked by subjecting them to russian roulette. If the user is interested in secondary electron effects, then high-weight electrons are split upon crossing a user-defined splitting plane. Directional source biasing works in conjunction with the directional bremsstrahlung splitting (DBS) algorithm already in place in BEAMnrc[11] for efficient photon beam calculations.

A detailed description of the DSB algorithm, including its relationship to directional bremsstrahlung splitting (DBS), is given in Section II of the paper. DSB is benchmarked by performing a voxel-by-voxel comparison of photon and electron fluence with and without source biasing turned on. This accuracy test is described in Section V.A. The optimization of DSB with respect to its adjustable parameters, photon splitting number, `nbrspl`, and the minimum linear distance between radially redistributed photons, Δ, is described in Sections V.B and V.C. A realistic chamber simulation used to demonstrate the viability of performing Monte Carlo calculations of k_Q is described in Section VI. Concluding remarks appear in Section VII.

II. THE DIRECTIONAL SOURCE BIASING (DSB) ALGORITHM

As mentioned above directional source biasing is coupled with directional bremsstrahlung splitting (DBS). It makes use of the following directional bremsstrahlung splitting input parameters: 1) bremsstrahlung splitting number (`nbrspl`), 2) splitting field radius (`fs`), 3) source-to-surface distance at which `fs` is defined (`ssd`).

Directional source biasing splits each primary photon arising from the Co-60 source (modeled using source 3 in BEAMnrc[8]) `nbrspl` times isotropically (*i.e.* uniformly distributed over 4π). Photons directed into the field defined by `fs` and `ssd` are retained, with weight `1/nbrspl` (*i.e.* they are "thin"), and photons directed outside the field are subject to Russian Roulette (survival probability=`1/nbrspl`). If a photon survives Russian Roulette, it has its weight increased to 1 (*i.e.* it is "fat").

In practice, significant CPU time can be saved by not generating all `nbrspl` split photons but only those directed into the field (*i.e.* Russian Roulette does not have to be played). Assuming `fs` << `ssd` then, to first order, the number of photons directed into the field, `nthin`, is given by:

$$\texttt{nthin} = \frac{\texttt{fs}^2}{\left(2\left(\texttt{ssd} - \texttt{zin}\right)\right)^2}\texttt{nbrspl} \tag{1}$$

where `zin` is the Z-position of the decay event. For each of the `nthin` photons, an end-point toward which it is directed is chosen by randomly selecting a point in the splitting field (points evenly-distributed throughout the field). The X, Y and Z direction cosines (`uin`, `vin`, `win`) are calculated based on photon's point of origin in the source (randomly chosen and evenly-distributed throughout the source volume) and the end-point. The photon is rejected if the following criterion is not met:

$$\texttt{rnd} < \texttt{win}^3 \tag{2}$$

where `rnd` is a random number chosen on [0,1]. Note that use of this rejection criterion is necessary for Equation (1) to hold. In addition, DSB generates a single fat photon directed outside the splitting field to represent any particles that would have survived Russian Roulette.

Further time in DSB is saved by considering the splitting field divided into radial bins as shown in Figure 1. Taking advantage of cylindrical symmetry in the top part of the Co-60 treatment head, the number of photons directed into a given bin can be reduced

by some integer factor, i, provided that, once such a photon crosses into the region where cylindrical symmetry no longer holds (the DSB splitting plane), it is split into i photons evenly distributed about its radial position. By playing Russian Roulette with photons directed into the bin with survival probability 1/i, DSB achieves a factor of i reduction in the number of photons. Photons which survive the Russian Roulette have their weight increased by a factor of i, and, thus have an incident weight of i/`nbrspl`. Once they have been split/redistributed at the DSB splitting plane, their weight is reduced by a factor of i (*i.e.* it returns to 1/`nbrspl`).

The parameter used by the DSB algorithm to divide the splitting field into bins is the minimum linear distance, Δ, between particles (projected to `ssd`) once they have been split and radially redistributed at the DSB splitting plane. Δ is a user input, and a discussion of efficiency of DSB versus the value of Δ appears in Section V.B below. DSB uses the convention that a photon directed into bin i is split i times. With the restriction that the i split photons must be a linear distance $\geq \Delta$ apart, the minimum radius of bin i, r_i, is given by:

$$r_i = \frac{\Delta}{2sin\left(\frac{\pi}{i}\right)} \ , \ i = 2, 3, ..., \texttt{nbin} \tag{3}$$

Note that r_1=0. The total number of radial bins, `nbin`, is equal to the number of times a photon can be split/radially-redistributed at the outer edge of the splitting field, `fs`, and is given by:

$$\texttt{nbin} = \frac{\pi}{asin\left(\frac{\Delta}{2\texttt{fs}}\right)} - 1 \tag{4}$$

Note also that the number of times a photon is split upon reaching the DSB splitting plane cannot exceed the overall splitting number, `nbrspl`. This results in a restriction on the value of Δ:

$$\Delta \geq 2\texttt{fs}\left[sin\left(\frac{\pi}{\texttt{nbrspl}}\right)\right] \tag{5}$$

In practice, `nbrspl` is very large, and a more realistic restriction on Δ is the maximum number of radial bins that can be considered. An implication of this splitting scheme is that photons directed into bin 1 are not split at all (nor are they subject to Russian Roulette).

A general expression for the linear distance, projected to `ssd`, between split redistributed photons directed towards radius, r, in bin i is $\Delta + \delta_i(r)$, where:

$$\delta_i(r) = \left(\frac{r}{r_i} - 1\right)\Delta \ , \ i = 2, 3, ..., \texttt{nbin} \tag{6}$$

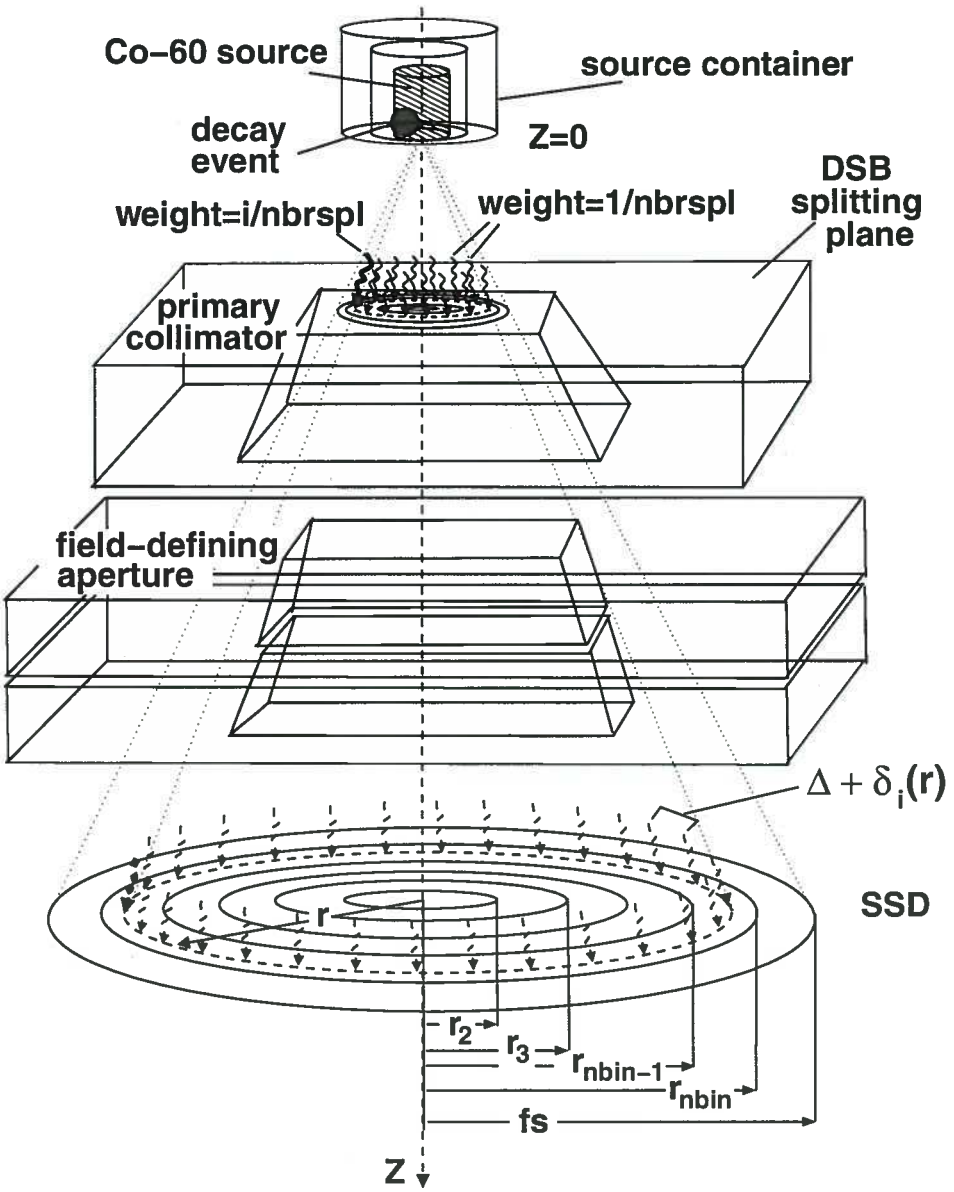

FIG. 1. A schematic of DSB in a Co-60 beam. The splitting field, defined by `fs` and `SSD`, is divided into `nbin` radial bins. A photon directed into bin i, defined by minimum radius, r_i, is split i times and redistributed about its radial position with a minimum linear distance between redistributed photons of Δ. Δ is a user input for DSB and is used to determine `nbin` and the r_i using Equations (4) and (3), respectively. Radial bins and Δ are defined as projected to the SSD, but actual splitting/redistribution of photons occurs at the DSB splitting plane, below which radial symmetry no longer holds. Primary photons directed into bin i have weight i/`nbrspl` (after having survived Russian Roulette with survival probability=1/i). Once they are split/redistributed, they have weight=1/`nbrspl`.

$\delta_i(r)$ is linear with r and attains a maximum as r→r_{i+1} (*i.e.* at the outer edge of the bin). Substituting expressions for r_i and r_{i+1} from Equation (3) into Equation (6):

$$\delta_i(r \rightarrow r_{i+1}) = \left(\frac{sin\left(\frac{\pi}{i}\right)}{sin\left(\frac{\pi}{i+1}\right)} - 1 \right) \Delta \ , \ i = 2, 3, ..., \texttt{nbin} \tag{7}$$

The maximum value of $\delta_i(r \rightarrow r_{i+1})$ is ~0.225Δ for bin 3.

The probability of a photon already directed into the splitting field being directed into bin i, p_i, is given by:

$$p_i = \frac{\left(r_{i+1}^2 - r_i^2\right)}{fs^2} / i \tag{8}$$

This is simply the ratio of the area of bin i to the area of the splitting field reduced by a factor of i to reflect the fact that i photons will be regenerated as soon as photon crosses the DSB splitting plane. Equation 1 for the total number of the original `nbrspl` split photons directed into the splitting field is modified to reflect the probabilities of photons being directed into the individual bins as follows:

$$\texttt{nthin} = \frac{\texttt{fs}^2}{\left(2\left(\texttt{ssd} - \texttt{zin}\right)\right)^2} \left(\sum_{i=1}^{\texttt{nbin}} p_i \right) \texttt{nbrspl} \tag{9}$$

Note that the value of `nthin` calculated using Equation (9) will be less than that calculated using Equation (1) because $\sum_{i=1}^{\texttt{nbin}} p_i$ <1. Again, this reflects the fact that photons will be regenerated once they cross the DSB splitting plane. Equation (9) also implicitly includes Russian Roulette (survival probability=1/i) played with photons directed into each bin i and, thus, eliminates the CPU time required to do this. For each of the `nthin` photons, an end-point in the splitting field toward which it is directed is chosen by first randomly selecting a radial bin index, i (evenly-distributed over the [1,`nbin`]), and then selecting a random point within this bin (evenly-distributed spatially througout the bin). This differs from the selection of random end-points evenly-distributed throughout the entire splitting field, corresponding to the definition of `nthin` in Equation (1) (See above), in that bins closer to the edge of the splitting field, with larger bin index, delimiting radii and areas are relatively undersampled. This is consistent with the fact that photons directed into these bins have higher weight and will be split by a larger number upon reaching the DSB splitting plane. The direction cosines, (`uin`, `vin`, `win`) are calculated based on the photon's point of origin (randomly chosen and evenly-distributed within the source) and the end-point in the splitting field, and the rejection criterion in Equation (2) must be satisfied for the photon to be accepted. Photons directed into bin i are immediately given weight i/`nbrspl`.

With the splitting scheme described above, each primary history generates more than one incident photon. Data for these photons (energy,position, direction) is retained in an array and a new primary history is initiated only when all particles in the array have been used.

Once an incident photon has been generated using the above scheme, transport makes use of the directional bremsstrahlung splitting (DBS) algorithm already in place in BEAMnrc with the following exceptions:

- Above the DSB splitting plane, higher-order photons from bremsstrahlung or other radiative events directed into the splitting field are subject to Russian Roulette with survival probability 1/i, where i is the index of the radial bin that they are directed into. Surviving photons have their weight increased by a factor of i. In standard directional bremsstrahlung splitting, these photons are not subject to Russian Roulette.

- If the electron splitting option in directional bremsstrahlung splitting is being used (necessary if contaminant electrons are to be included), then any bremsstrahlung or other radiative events undergone by non-fat secondary charged particles, with weight i/`nbrspl`, above the DSB splitting plane are split i times. Resultant photons are subject to Russian Roulette as described in the item above. In standard directional bremsstrahlung splitting with electron splitting, such interactions are not split. Note that secondary charged particles with weight i/`nbrspl` can only exist between the Russian Roulette plane (specified with electron splitting) and the DSB splitting plane and, thus, only if the user places the Russian Roulette plane below the DSB splitting plane.

For a complete description of directional bremsstrahlung splitting, see the paper by Kawrakow et al[11].

The introduction of DSB includes a redefinition of the BEAMnrc stack variable, `iphat`. Previously, `iphat` was used with directional bremsstrahlung splitting as a flag to indicate if a particle was fat (`iphat`=1) or thin (`iphat`=0). Now, however, particles above the DSB splitting plane can have varying degrees of "thinness", depending on which radial bin, i, they or their parent particle is directed into. The redefined `iphat` takes on values 1,2,...,`nbin`,`nbrspl` above the DSB splitting plane, where `iphat`=`nbrspl` means the particle is fat. Thus, `iphat` keeps an instantaneous record of which radial bin the particle (or its parent) is directed into

and allows quick computation of Russian Roulette survival probabilities (1/iphat). Note that below the DSB splitting plane iphat will be either 1 (thin) or nbrspl (fat).

III. SIMULATIONS

A. Co-60 beam simulations

In order to study the efficiency of DSB, Co-60 beams are simulated using both BEAMnrc and beampp. The latter uses the EGSnrc C++ class library, egspp[12], to simulate accelerators. It has much of the functionality of BEAMnrc and uses BEAMnrc input files. With both BEAMnrc and beampp, the Co-60 simulations are run as shared-library sources. Shared library sources have been shown[13] to be similar in efficiency to phase space sources, provided that variance reduction techniques are used to optimize the accelerator simulation, and have the advantage that large phase space files need not be stored.

Geometric parameters used for the Co-60 beam simulations are the same as those employed by Mora et al[10] to simulate an Eldorado 6 unit used at the National Research Council of Canada for ion chamber calibrations. The energy spectrum of the Co-60 source is a simple 2-line spectrum, with equiprobable gamma rays at 1.17 and 1.33 MeV.

To reduce their simulation times, Mora et al[10] varied the electron transport cutoff energy (ECUT) throughout the treatment head, with ECUT ranging from 0.6 MeV in the collimator air and air gap to 1.921 MeV in the lead and iron in the source container. Moreover, they employed range rejection – immediate cessation of charged particle transport if the particle cannot make it to the nearest region boundary – on charged particles at all energies (*i.e.* the ESAVE parameter was set $>$ 1.33 MeV). These time saving techniques impact bremsstrahlung yield, and it is found that they result in a $\sim$0.5% difference in peak photon fluence per primary history when compared to simulations where ECUT is set to 0.7 MeV in all regions except the collimator air and air gaps and range rejection is only performed on charged particles with energy $\leq$ 0.7 MeV (ESAVE=0.7 MeV). Since <0.5% precision is desired to assess potential biases in photon fluence introduced by the DSB algorithm, all simulations in this study use ECUT=0.7 MeV everywhere except collimator air and air gaps, where ECUT=0.6 MeV, and range rejection with ESAVE=0.7 MeV. The photon transport cutoff energy (PCUT) is 0.01 MeV.

Co-60 simulations using DSB also make use of a feature in directional bremsstrahlung splitting called augmented range rejection. Charged particles that are not fat (`i.e.` are below the Russian Roulette plane and/or electron splitting plane defined in directional bremsstrahlung splitting[11]) and have energy $\leq$ **ESAVE** are first subject to range rejection. If they are not eliminated in this way, then they are subject to Russian Roulette with survival probability `iphat/nbrspl`. If they survive Russian Roulette, then they have their weight increased by a factor of `iphat` (i.e they become fat). This can save some simulation time over standard range rejection, although in the case of the Co-60 irradiators described here, the savings amount to only a few percent.

The cylindrical symmetry of this particular Co-60 treatment head geometry ends at the top of the primary collimator (which has square symmetry and is simulated using a PYRAMIDS component module in BEAMnrc). Thus, the DSB splitting plane, where photons are split and redistributed evenly about their radial position, as described in Section II above, is placed at the top of the primary collimator geometry (Z=1.5 cm) in all simulations using DSB.

In order to include contaminant electrons in the beam, Co-60 simulations with DSB are run with the electron splitting option in directional bremsstrahlung splitting turned on. Investigations of contaminant electrons undertaken as part of this study show that contaminant electrons reaching the treatment field originate in the source container (ending at Z=0 in the BEAMnrc simulation geometry), the primary collimator (ending at Z=7.7 cm) and the field-defining aperture (ending at Z=27.4 cm). Thus, to include all potential contaminant electrons, the electron splitting plane (where fat electrons are split **nbrspl** times) is placed at the bottom of the field-defining aperture (modeled using a PYRAMIDS geometry in BEAMnrc and beampp) with the Russian Roulette plane (below which Russian Roulette is not played with thin electrons) defined at Z=27 cm. Since there is no cylindrical symmetry at the bottom of the aperture, the option to redistribute the split electrons evenly about their radial position at the electron splitting plane is not used. For more information about electron splitting with directional bremsstrahlung splitting, see the paper by Kawrakow et al[11].

The beampp code offers a more comprehensive electron splitting scheme wherein differing "electron importances" (which are the same as electron splitting numbers) can be assigned to the various regions throughout the accelerator. Since the contaminant electrons originate

throughout the treatment head, it was initially thought that a judicious selection of electron importances would optimize electron fluence efficiency. However, it was found that varying the electron importance has little effect, and a simpler scheme equivalent to placing the electron splitting plane at the bottom of the field-defining aperture (as in the BEAMnrc simulations) is used in the simulations reported.

Fluence calculations described below are performed with and without electron splitting, while the dose and chamber calculations all include contaminant electrons.

B. Fluence calculations

Fluence at the bottom of the accelerator is scored using a custom EGSnrc user code, `fluxyznrc`, which scores planar fluence divided by the Z direction cosine (thus, an estimate of the actual fluence) in rectangular scoring zones. The simulation time for the `fluxyznrc` geometry is negligible compared to that for the Co-60 shared library source, so fluence efficiencies determined using `fluxyznrc` are the same as those of a stand-alone Co-60 simulation.

C. Dose calculations

Doses are calculated in a water phantom with X, Y and Z dimensions of 20.5 cm, 20.5 cm and 30 cm, respectively, using the EGSnrc user code, DOSXYZnrc[14]. Dose scoring voxels are 0.5 cm × 0.5 cm × 0.5 cm. `ECUT` and `PCUT` in the phantom are 0.7 MeV and 0.01 MeV, respectively. In order to increase dose calculation efficiency in the phantom, photons are split 32 times (with weight reduced by 1/32) with interaction sites distributed evenly along the original photon path. For more information about photon splitting in DOSXYZnrc, see the paper by Kawrakow and Walters[13]. In order to prevent higher-weight electrons from compromising statistics in the dose phantoms, contaminant electrons in the Co-60 beam are split 32 times upon entering the phantom. In addition, the "HOWFARLESS" option for increasing the efficiency of dose calculations in a homogeneous phantom by speeding up charged particle transport is also used. For more information on "HOWFARLESS", see the paper by Walters and Kawrakow[15].

In these dose calculations, the air gap between the bottom of the field-defining aperture

and the source-to-surface distance (SSD=100 cm) is included in the DOSXYZnrc simulation instead of the Co-60 simulation as this has been found[13] to be slightly more efficient when photon splitting is used.

D. Chamber calculations

The beam quality correction factor, k_Q, for an ion chamber is given by[1]:

$$k_Q = \frac{[D_w/D_c]^Q}{[D_w/D_c]^{^{60}Co}} \tag{10}$$

where Q denotes the beam quality, D_c is the dose to the chamber, and D_w is the dose to a small volume of water at the same point of measurement. The Co-60 dependent ratio, $[D_w/D_c]^{^{60}Co}$, is measured or calculated at a reference depth of 5 cm.

In this study, $[D_w/D_c]^{^{60}Co}$ is calculated for a simulated Farmer NE-2571 chamber. The NE-2571 is a thimble chamber commonly used for clinical linear accelerator calibrations. The geometry of the simulated NE-2571 chamber is based on that used by Larussa et al[16] and Muir and Rogers[17] and uses radiographic measurements of the chamber made at the National Research Council of Canada. Simulations are performed using the `egs_chamber` user code, which uses the egspp C++ class library for EGSnrc[12] and which has been optimized for efficient calculations of in-phantom ion chamber dose by the introduction of photon cross-section enhancement in the chamber cavity volume and other variance reduction techniques[18]. The electron and photon transport cutoff energies, ECUT and PCUT, used in these simulations are 0.521 MeV and 0.01 MeV, respectively.

The ratio, $[D_w/D_c]^{^{60}Co}$, for a 10 × 10 cm^2 field (SSD=100 cm) calculated using a full, optimized BEAMnrc simulation of the Co-60 beam as a shared library source is compared with the same ratio calculated using a simple photon point source with an energy spectrum derived from phase space data collected at the bottom of the optimized BEAMnrc simulation. Phase space data is collected for 55×10^6 photons, representing 10×10^6 primary histories, and is sufficient to obtain 0.1% uncertainty on spectral peaks with an energy resolution of 0.01 MeV. In the case of the full simulation source, contaminant electrons are included in the chamber calculation.

IV. EFFICIENCY

The efficiency, ϵ, of a simulation is given by:

$$\epsilon = \frac{1}{T\sigma^2} \tag{11}$$

where σ is uncertainty, usually estimated as the standard deviation, on the quantity of interest (*i.e.* dose or fluence), and T is the CPU time required to obtain the value of σ.

In this study, efficiency calculations are performed on 1.8 GHz Opteron 244 CPU's. Every calculation is broken up into 15 separate jobs to allow uncertainties on the efficiencies to be estimated.

V. RESULTS

A. Fluence accuracy

Figure 2 shows the profile in the field X-direction of the fractional difference between fluence calculated with and without DSB for the 10 × 10 cm^2 Co-60 beam at SSD=100 cm. Calculations with DSB use `nbrspl`=2000 and electron splitting at the bottom of the field-defining aperture. The calculation without DSB uses a BEAMnrc simulation source, while calculations with DSB are performed using both BEAMnrc (solid lines) and beampp (dashed lines) simulation sources. Fluence scoring zones in the `fluxyznrc` code are 2 cm × 10 cm (*i.e.* the entire Y-dimension of the field is included). The large scoring zones are used so that high precision can be obtained in a reasonable amount of time. Differences are shown for photon fluence (Figure 2a) and electron fluence (Figure 2b). Without DSB, photon fluences are calculated to within 0.05% uncertainty and electron fluences are calculated to within 1.5% uncertainty (with reasonable simulation times becoming the limiting factor), while with DSB, photon fluence is calculated to within 0.03% uncertainty and electron fluence to within 0.8% uncertainty.

Figure 2a shows agreement between photon fluence estimated with DSB and without DSB to within 2σ across the 10 × 10 cm^2 field. Given this level of agreement, there is a 5% likelihood that the apparent systematic overestimation of photon fluence with DSB may be due to statistics in the calculation without DSB. Given the high precision of these calculations, however, any systematic difference due to DSB will be insignificant in practical

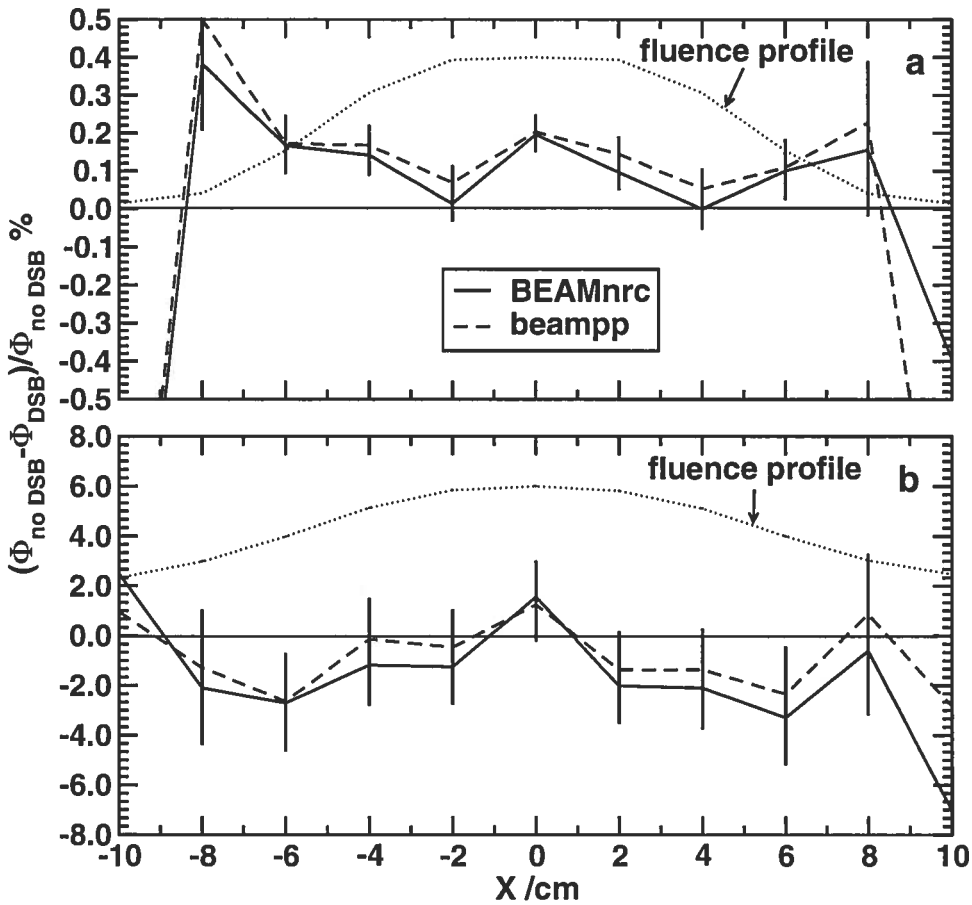

FIG. 2. X-direction profile of the fractional difference between photon (a) and electron (b) fluence calculated without directional source biasing (DSB) and with DSB in the 10 × 10 cm^2 field (SSD=100 cm) of a simulated Co-60 beam. The DSB simulations are performed with `nbrspl`=2000 and electron splitting below the field-defining aperture. Calculations with DSB are done using both BEAMnrc (differences shown as solid lines) and beampp (differences shown as dashed lines) simulation sources, while the base calculation without DSB is performed using a BEAMnrc simulation source. Fluence scoring zones are 2 cm × 10 cm. Normalized photon and electron fluence profiles are also shown.

applications. Increasing the precision of the calculation without DSB to confirm whether or not the difference is statistical is impractical given the length of time already required to achieve 0.05% uncertainty. Figure 2b shows that, within the precision available from the simulation with no DSB, no significant change in electron fluence occurs when DSB is used. Differences similar to those in Figure 2a might occur if electron fluence was calculated

down to similarly high precision. In addition, Figure 2 shows that DSB calculations using BEAMnrc and beampp are in good agreement.

B. Fluence efficiency

Figure 3 shows the total fluence efficiency for photons (a) and contaminant electrons (b) inside the 10 × 10 cm^2 field (SSD=100 cm) of the Co-60 beam as a function of the bremsstrahlung splitting number, `nbrspl`, used with DSB. Efficiencies are all relative to the photon or electron efficiency without DSB. Fluence is scored in 1 × 1 cm^2 zones using the fluxyznrc code with the Co-60 beam as a simulation source. Total fluence efficiency in the field is calculated by summing the squares of the uncertainties in all zones contained within the field and using this as σ^2 in Equation (11). The solid lines show efficiencies using BEAMnrc to simulate the Co-60 beam, and the dashed lines show efficiencies using beampp to simulate the beam. Note, though, that the reference, no-DSB, case is always simulated with BEAMnrc. In the case of photon efficiencies (Figure 3a), results are shown with (thick lines) and without (thin lines) electron splitting at the bottom of the field-defining aperture.

Figure 3a shows that the peak photon fluence efficiency occurs at `nbrspl`$\sim$20,000 for Co-60 simulations using BEAMnrc and at `nbrspl`$\sim$30,000 for simulations using beampp. The efficiency shows little variation around the peak, so the exact choice of `nbrspl` is not critical. Also note that the use of electron splitting does not significantly decrease the photon efficiency (yet is essential to obtain contaminant electrons in the field). This implies that tracking the additional electrons through the airgap below the field-defining aperture does not significantly increase the simulation time.

Peak efficiency of contaminant electron fluence occurs at lower values of `nbrspl`, with peaks at `nbrspl`$\sim$7,500 for the BEAMnrc simulation and `nbrspl`$\sim$15,000 for the beampp simulation. With increasing `nbrspl`, the uncertainty in electron efficiency increases, and it becomes difficult to determine its actual value. This is because contaminant electrons have a relatively low fluence (only $\sim$0.1% of the total fluence), making the arrival of a contaminant electron in the field a low-probability event. This means that high values of `nbrspl`, which is also the electron splitting number, increase the differences in efficiency between those runs which contribute more contaminant electrons and those which contribute fewer (recall that 15 runs are used to determine each efficiency point). Since electron fluence is such a

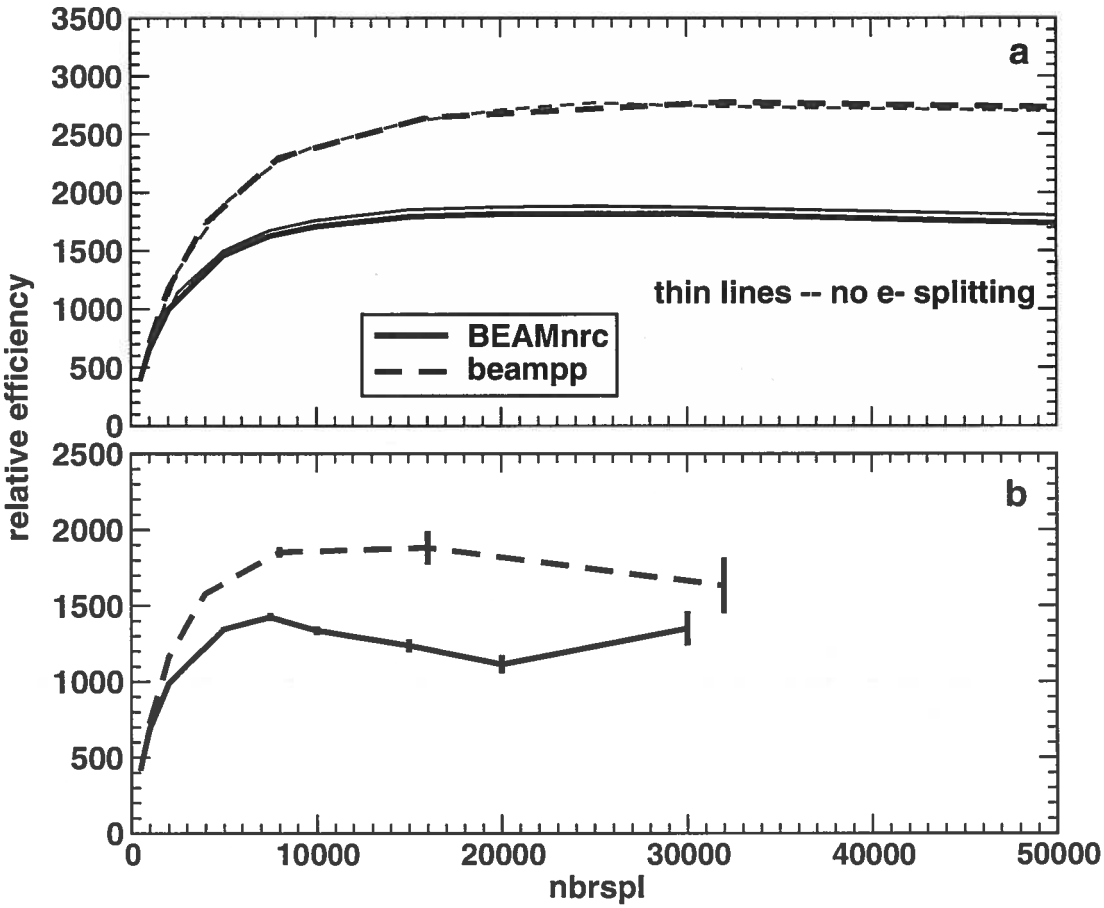

FIG. 3. Total fluence efficiency of photons (a) and electrons (b) inside the 10 × 10 cm^2 field (SSD=100 cm) of Co-60 beams simulated with DSB as a function of the bremsstrahlung splitting number, `nbrspl`. Efficiencies are relative to those without DSB. Fluence is scored in 1 × 1 cm^2 zones and total efficiency is calculated by summing the squares of the uncertainties in each zone contained within the field and using this as σ^2 in Equation (11). Simulations with DSB are performed with BEAMnrc (solid lines) and beampp (dashed lines). Photon fluence efficiencies are also shown for simulations in which electrons are not split below the field defining jaws (thin lines).

small fraction of the total fluence, the optmimum setting of `nbrspl` is that which maximizes photon fluence efficiency.

395 Figure 4 shows photon (a) and electron (b) fluence efficiencies in 1 × 1 cm^2 scoring zones along the X-axis of the 10 × 10 cm^2 field (SSD=100 cm). Efficiencies are relative to the photon or electron efficiencies with no DSB. Again, results are shown for simulations using DSB with both BEAMnrc and beampp. The photon efficiency results without electron

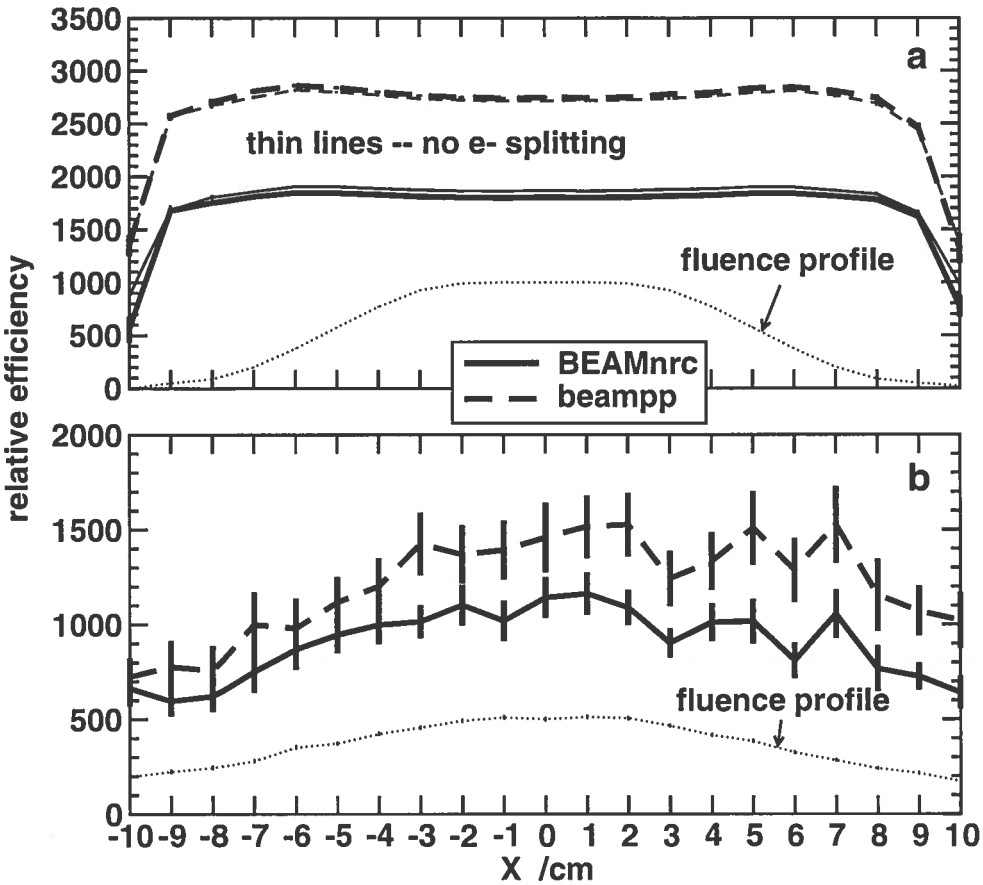

FIG. 4. Fluence efficiency of photons (a) and electrons (b) in 1 × 1 cm^2 scoring zones along the field X-axis in 10 × 10 cm^2 Co-60 beams (SSD=100 cm) simulated with DSB. Efficiencies are relative to those with no DSB. The fluence profiles, themselves, are also shown for reference. The values of `nbrspl`, the splitting number, used are 20,000 for simulations using BEAMnrc (with and without electron splitting below the field-defining aperture), 32,000 for the beampp simulation with electron splitting, and 30,000 for the beampp simulation without electron splitting. These values of `nbrspl` result in peak, or near-peak, photon fluence efficiency.

splitting (thin lines) are also shown in Figure 4a. The values of `nbrspl` for the simulations
400 shown here are chosen to be at or near peak photon fluence efficiency in Figure 3. Thus, the BEAMnrc simulations use `nbrspl`=20,000 and the beampp simulations use `nbrspl`=32,000 (with electron splitting) or 30,000 (without electron splitting).

Figure 4a shows that the photon fluence efficiency gain due to DSB is almost constant across the 10 × 10 cm^2 field. The electron fluence efficiency gains (Figure 4b) have high

uncertainty due to the large values of `nbrspl` necessary to optimize photon fluence (see above for a more detailed explanation). These uncertainties could be reduced somewhat by running more histories to decrease the electron fluence uncertainty.

Figure 5 shows the total fluence efficiency for photons (a) and electrons (b) in the 10 × 10 cm^2 field as a function of the DSB parameter, Δ, the minimum linear distance (as projected to the SSD of 100 cm) between photons when they are split and redistributed about their radial position at the DSB splitting plane. BEAMnrc simulations use `nbrspl`=20,000 and beampp simulations use `nbrspl`=32,000 (*i.e.* the values for optimum photon fluence efficiency).

From Figure 5 it is evident that fluence efficiency does not vary greatly over a large Δ range (1 cm – 2.5 cm). Once again, note the large uncertainty on electron fluence efficiency due to the fact that contaminant electrons at the scoring plane are relatively rare. The value of Δ used for all other simulations in this study is 1.5 cm, which gives near-optimum efficiency gains for photon fluence.

Overall, the scheme of splitting and redistributing photons upon leaving the radially symmetric portion of the treatment head increases the peak total fluence efficiency (obtained with `nbrspl`=20,000 and Δ=1.5 cm) by ~20% for photons and ~15% for contaminant electrons.

C. Dose efficiency

Figure 6 shows the relative efficiency of all doses > 0.5×D_{max} as a function of `nbrspl` for the 10 × 10 cm^2 Co-60 beam incident on a 20.5 cm × 20.5 cm × 30 cm water phantom with 0.5 cm × 0.5 cm × 0.5 cm voxels. Doses are calculated to within 0.5% uncertainty in simulations using DSB and to within 1.5% uncertainty in the calculation without DSB. Recall that in all DOSXYZnrc simulations, photons are split 32 times (necessitating a 32x splitting of contaminant electrons), and that the "HOWFARLESS" technique for increasing efficiency in a homogeneous phantom is also used. Overall, these variance reduction techniques increase the efficiency of the reference (no DSB) calculation by a factor of ~30. The labeled dotted line in Figure 6 shows the efficiency of the calculation without DSB using the time-saving strategies of Mora et al[10] (see Section III.A above): high values of `ECUT` in the source container and range rejection at all energies.

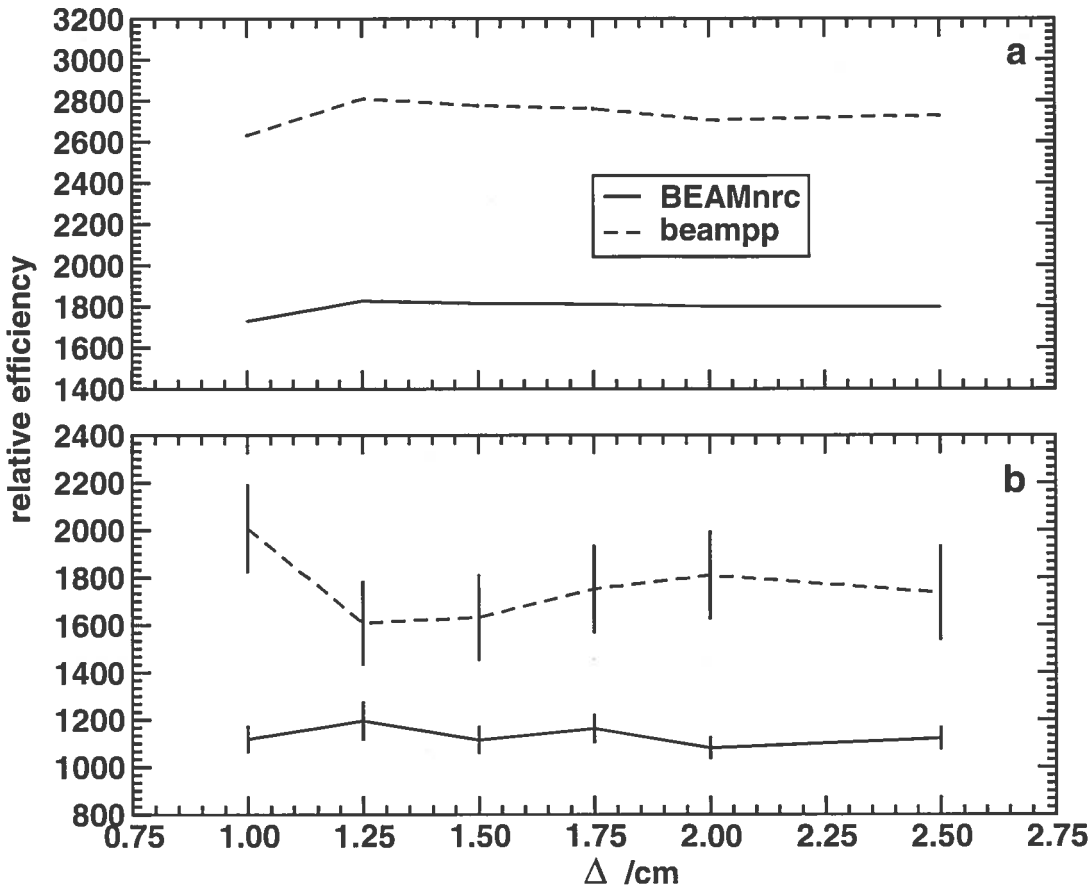

FIG. 5. Relative total fluence efficiency for photons (a) and electrons (b) inside the 10 × 10 cm^2 field (SSD=100 cm) of the Co-60 beam simulations using DSB as a function of Δ, the minimum linear distance (projected to SSD=100 cm) between photons when they are split and redistributed about their radial position at the DSB splitting plane (the top of the primary collimator). For BEAMnrc simulations `nbrspl`=20,000 and for beampp simulations `nbrspl`=32,000, which result in optimum photon fluence efficiency.

Similar to fluence efficiency as a function of `nbrspl` (see Figure 3), the dose efficiency with DSB is optimized at bremsstrahlung splitting numbers in the range 20,000–40,000, with the optimum value of `nbrspl` for BEAMnrc simulations being lower than that for beampp simulations. Also, the efficiency does not vary significantly over a large range of `nbrspl` values around the peak, so the exact setting of `nbrspl` is not critical. Note that the peak efficiency of DSB simulations using beampp, 420 times more efficient than without DSB, is only ~5% higher than the peak efficiency of DSB simulations using BEAMnrc, 400

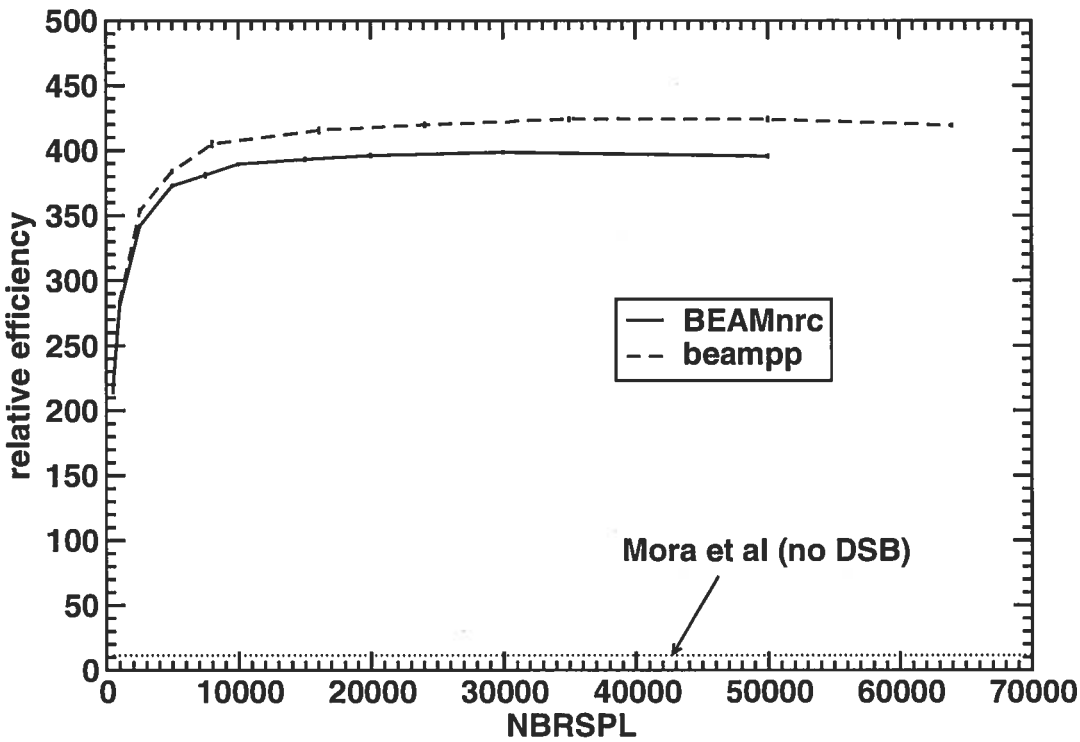

FIG. 6. Efficiency of all doses $> 0.5 \times D_{max}$ as a function of the bremsstrahlung splitting number, `nbrspl`, for the 10 × 10 cm^2 Co-60 beam incident on a water phantom. Efficiencies are relative to the efficiency without DSB. The water phantom is modeled using DOSXYZnrc and has overall X, Y, Z dimensions 20.5 cm × 20.5 cm × 30 cm divided into 0.5 cm × 0.5 cm × 0.5 cm voxels. Results are shown for simulations using DSB with BEAMnrc (solid line) and with beampp (dashed line). The reference calculation uses BEAMnrc to model the Co-60 beam. The dotted line shows the efficiency of the calculation without DSB using the strategy of varying `ECUT` and performing charged particle range rejection at all energies employed by Mora et al[10]. All calculations use 32 × photon splitting and the "HOWFARLESS" option for speeding up charged particle transport in the DOSXYZnrc phantom. Doses include contributions from contaminant electrons.

times more efficient than without DSB. This is in contrast to fluence efficiency, where peak photon fluence efficiency using beampp is ~55% higher than that using BEAMnrc, and shows that, with highly-efficient beam simulations, the time required for the DOSXYZnrc

phantom calculation becomes the dominant factor in the overall efficiency.

The peak efficiency with DSB is ~40 times greater than the efficiency that can be achieved using the time-saving techniques of Mora et al[10] (*i.e.* using high ECUT values in the source container and subjecting all charged particles to range rejection, regardless of energy) which is shown as the labeled, dotted line in Figure 6. Thus, the calculation mentioned in the Introduction (already optimized using photon splitting and "HOWFARLESS") that requires over 100 hours on 15 1.8 GHz Opteron 244 CPU's to obtain $< 0.1\%$ dose uncertainty now only requires ~2.5 hours. Moreover, the use of Mora et al's parameters has been found to result in inaccuracies in fluence/primary history (see Section III.A) which will result in dose inaccuracies.

VI. CHAMBER CALCULATIONS

Based on the fluence efficiency studies in Section V.B above, the BEAMnrc simulation of the Co-60 beam used to calculate $[D_w/D_c]^{^{60}Co}$ is optimized using a DSB splitting number, `nbrspl`, of 20,000 and a minimum linear distance between redistributed split photons, Δ, of 1.5 cm. The value of `nbrspl` optimizes photon fluence efficiency at a slight cost to the electron fluence efficiency.

D_c and D_w are calculated to within 0.1% uncertainty, requiring total cpu times of 6.9 hrs and 2.3 hrs, respectively, for calculations using the full Co-60 simulation source, and 4.9 hrs and 1.3 hrs, respectively, for calculations using the photon point source.

The value of $\left[\frac{D_w}{D_c}\right]^{^{60}Co}$ calculated at a depth of 5 cm for a 10 cm × 10 cm field (SSD=100 cm) using the full Co-60 beam simulation source is 1.109 ± 0.002. The value of $\left[\frac{D_w}{D_c}\right]^{^{60}Co}$ calculated using the photon point source is 1.108 ± 0.002. Agreement at this level has also been observed by Muir and Rogers[17] using a phase space source derived from a BEAMnrc Co-60 treatment head simulation and confirms the negligible effect of secondary electrons at 5 cm depth and the unimportance of the geometric details of the Co-60 treatment head for these calibrations.

VII. CONCLUSION

By selectively splitting only those primary photons directed into a user-defined splitting field, which encompasses the treatment field, directional source biasing (DSB) increases the efficiency of Co-60 beam dose calculations (10×10 cm^2 field) by a factor of almost 40. In addition to preferential photon splitting, the DSB algorithm also saves computation time by taking advantage of the cylindrical symmetry at the top of the Co-60 treatment head to reduce the number of photons tracked there and by playing Russian Roulette with secondary electrons, regenerating the latter if electron contamination is to be included. The algorithm works in conjunction with directional bremsstrahlung splitting. Together with variance reduction techniques already in place in DOSXYZnrc (photon splitting, "HOWFARLESS"), DSB allows calculations that were previously prohibitively long to be done in a reasonable amount of time. The photon splitting numbers (directional bremsstrahlung splitting input, `nbrspl`) at which DSB reaches peak efficiency are in the range 20,000–40,000. The optimum splitting number is likely to vary depending on field size and scoring voxel size, however efficiency does not vary greatly about the optimum value, and a setting of `nbrspl`=20,000 should achieve near-optimum efficiency.

Use of DSB opens up the possibility of regularly performing Monte Carlo Co-60 treatment head simulations which can be used to commission Co-60 beams for use in IMRT and IGRT. Also, since Co-60 energies are used as references in the determination of beam quality factors and correction factors, DSB makes possible the accurate calculations of these in a reasonable time. This is illustrated above by the example calculation of $[D_w/D_c]^{^{60}Co}$ in a realistic ion chamber using a full BEAMnrc simulation of a Co-60 treatment head as a source.

Finally, DSB is not limited to Co-60 sources but will increase the efficiency of other radiating sources, such as Cs-137, making simulation of these sources more practical. The DSB option has been implemented in the current release of BEAMnrc, and documentation will be included in future releases of the BEAMnrc Users Manual.

ACKNOWLEDGMENTS

The author wishes to acknowledge Iwan Kawrakow, currently at ViewRay, who came up with the initial concept for DSB and coded extensive portions of the algorithm during

his tenure at the National Research Council of Canada (NRCC). The author also acknowledges Bryan Muir at the NRCC for providing details of the NE2571 chamber and his MC calculations of k_Q. Malcolm McEwen and Frédéric Tessier at the NRCC provided valuable comments on early drafts of the manuscript. Finally, Ernesto Mainegra-Hing of the NRCC is acknowledged for encouraging the publication of this work.

---

[1] P. R. Almond, P. J. Biggs, B. M. Coursey, W. F. Hanson, M. S. Huq, R. Nath, and D. W. O. Rogers, Med. Phys. **26**, 1847 (1999).

[2] M. McEwen, L. DeWerd, G. Ibbott, D. Followill, D. W. O. Rogers, S. Seltzer, and J. Seuntjens, Med. Phys. **41**, 041501 (2014).

[3] C. P. Joshi, J. Darko, P. B. Vidyasagar, and L. J. Schreiner, Phys Med Biol **53**, 575 (2008).

[4] C. Fox, H. E. Romeijn, B. Lynch, C. Men, D. M. Aleman, and J. F. Dempsey, Phys. Med. Biol. **53**, 3175 (2008).

[5] L. J. Schreiner, C. P. Joshi, J. Darko, A. Kerr, G. Salomons, and S. Dhanesar, J Med Phys **34**, 133 (2009).

[6] J. Dempsey, B. Dionne, J. Fitzsimmons, A. Haghigat, J. Li, D. Low, S. Mutic, J. Palta, H. Romeijn, and G. Sjoden, Med. Phys. (abstract) **33**, 2254 (2006).

[7] T. Kron, D. Eyles, L. J. Schreiner, and J. Battista, J. Med. Phys. **31**, 242 (2006).

[8] D. W. O. Rogers, B. Walters, and I. Kawrakow, NRC Report PIRS 509(a)revI (2005).

[9] 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) (National Research Council of Canada, Ottawa, Canada, 2003).

[10] G. Mora, A. Maio, and D. W. O. Rogers, Med. Phys. **26**, 2494 (1999).

[11] I. Kawrakow, D. W. O. Rogers, and B. Walters, Med. Phys. **31**, 2883 (2004).

[12] I. Kawrakow, "egspp: the EGSnrc C++ class library," Technical Report PIRS–899 (National Research Council of Canada, Ottawa, Canada, 2005).

[13] I. Kawrakow and B. R. B. Walters, Med. Phys. **33**, 3046 (2006).

[14] B. R. B. Walters, I. Kawrakow, and D. W. O. Rogers, NRC Report PIRS 794 (rev B) (2005).

[15] B. R. B. Walters and I. Kawrakow, Med. Phys. **34**, 3794 (2007).

[16] D. J. L. Russa, M. McEwen, and D. W. O. Rogers, Med. Phys. **34**, 4690 (2007).

[17] B. R. Muir and D. W. O. Rogers, Med. Phys. **37**, 5939 (2010).

[18] J. Wulff, K. Zink, and I. Kawrakow, Med. Phys. **35**, 1328 (2008).