Published in final edited form as:
Phys Med Biol. 2010 December 7; 55(23): 7107–7120. doi:10.1088/0031-9155/55/23/S11.

A GPU implementation of a track-repeating algorithm for proton radiotherapy dose calculations

Pablo P Yepes[1,3], **Dragan Mirkovic**[2], and **Phillip J Taddei**[2]

[1] Department of Physics and Astronomy, MS 315, Rice University, 6100 Main Street, Houston, TX 77005, USA

[2] Department of Radiation Physics, Unit 1202, The University of Texas M D Anderson Cancer, 1515 Holcombe Blvd, Houston, TX 77030, USA

Abstract

An essential component in proton radiotherapy is the algorithm to calculate the radiation dose to be delivered to the patient. The most common dose algorithms are fast but they are approximate analytical approaches. However their level of accuracy is not always satisfactory, especially for heterogeneous anatomical areas, like the thorax. Monte Carlo techniques provide superior accuracy; however, they often require large computation resources, which render them impractical for routine clinical use. Track-repeating algorithms, for example the fast dose calculator, have shown promise for achieving the accuracy of Monte Carlo simulations for proton radiotherapy dose calculations in a fraction of the computation time. We report on the implementation of the fast dose calculator for proton radiotherapy on a card equipped with graphics processor units (GPUs) rather than on a central processing unit architecture. This implementation reproduces the full Monte Carlo and CPU-based track-repeating dose calculations within 2%, while achieving a statistical uncertainty of 2% in less than 1 min utilizing one single GPU card, which should allow real-time accurate dose calculations.

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

1. Introduction

Radiation therapy is an important component of the treatment of cancer. Radiation dose absorbed in normal tissues produces acute effects, for example necrosis, and late effects, such as carcinogenesis. An essential component for the quality of a radiotherapy treatment plan is the accuracy of dose calculations (Papanikolaou *et al* 2004). The clinical advantages of more accurate dose calculations—tumor recurrence, local control and normal tissue complications—have not been fully quantified and require further investigation. Nevertheless, evidence exists that dose differences on the order of 7% are clinically detectable (Dutreix 1984). Moreover, several studies have shown that 5% changes in dose can result in 10–20% changes in tumor control probability or up to 20–30% changes in normal tissue complication probabilities (Orton *et al* 1984, Stewart and Jackson 1975, Goitein and Busse 1975).

Dose distributions in proton therapy are typically calculated by commercial treatment planning engines based on analytical pencil-beam algorithms (Petti 1992, Russell *et al* 1995, Hong *et al* 1996, Deasy 1998, Schneider *et al* 1998, Schaffner *et al* 1999, Szymanowski and Oelfke 2002).

[3] Author to whom any correspondence should be addressed. yepes@rice.edu.

Schaffner *et al* (1999) reviewed various analytical proton dose models and concluded that no single pencil-beam model can predict the dose correctly in every situation. They also concluded that the Monte Carlo approach is more accurate than any analytical model and should be used to verify the dose distributions in situations above a certain level of anatomical complexity. Soukup *et al* (2005) derived a pencil-beam method from Monte Carlo simulations; this method works well for simple heterogeneous or slab-structured phantoms; however it does not achieve the accuracy of the Monte Carlo approach for phantoms describing more complex heterogeneous media, for example head and neck and pelvic geometries. Ciangaru *et al* (2005) benchmarked analytical calculations of proton doses in simple heterogeneous phantoms and concluded that the algorithms were reasonably accurate for predicting doses at anatomic sites containing laterally extended inhomogeneities that are comparable in density to one another and located away from the Bragg peak. However, the algorithm had mixed success in calculating proton doses in areas with a combination of high and low density media.

The Monte Carlo approach has been shown to provide higher accuracy (Titt *et al* 2008, Koch *et al* 2008, Newhauser *et al* 2008, Paganetti *et al* 2008, Giebeler *et al* 2010) than pencil-beam algorithms; however, its clinical utilization has been hampered by its high computational requirements. In one study with the Monte Carlo code MCNPX (Pelowitz 2007), for a typical three-beam proton lung cancer, simulating all three proton radiotherapy treatment fields with acceptable statistical precision required up to 5000 h of central processor unit (CPU) time (Taddei *et al* 2009). The high computation times make the use of robust Monte Carlo codes impractical for routine clinical radiotherapy dose calculations, i.e. without the use of large-scale computer clusters.

Various simplified Monte Carlo approaches have also been reported in the literature. Kohno and collaborators (2003) implemented a proton Monte Carlo dose algorithm in which a reduced number of physics processes was taken into account. This approach increased the computational efficiency while largely preserving the accuracy of the calculations (Hotta *et al* 2010). Fippel and Soukup (2004) developed another proton Monte Carlo code based on the concept of the voxel Monte Carlo algorithm that had previously been applied to photons and electrons (Fippel 1999). They reported good agreement between dosimetric predictions from their code and from full Monte Carlo codes. Their calculation times were 35 and 13 times faster than Monte Carlo codes GEANT4 (Agostinelli *et al* 2003, Allison *et al* 2006) and FLUKA (Fasso *et al* 2005), respectively. Tourovsky *et al* (2005) implemented a stochastic proton transport code with simplified physics models and tested it in a variety of clinical cases. Computation times relative to other algorithms were not reported. Li *et al* (2005) reported a track-repeating algorithm for proton therapy that maintained the accuracy of the Monte Carlo technique, while significantly decreasing computation times. However, their validation was limited to simple heterogeneous phantoms. Extending their work, we developed an alternative track-repeating algorithm, the fast dose calculator (FDC), and applied it to clinical proton treatment planning for the pelvis (Yepes *et al* 2009a, 2009b) and the thorax (Yepes *et al* 2010). The FDC reproduced the results of GEANT4 simulations to within 2%, yet required less than 1% of the computation time. While the reduction in computation time with the FDC was significant, calculation times were still on the order of an hour on a single CPU for the dose calculation of a typical radiotherapy treatment. An alternative, more efficient computation device is needed to use the FDC for routine clinical dose verification calculations.

Calculations can be accelerated by performing the computation in a parallel manner on multiple processor systems, like CPU clusters or graphics processing units (GPUs). Dose calculations performed on a large number of distributed CPUs have been reported in the literature (Vadapalli *et al* 2010). GPU clusters are less expensive and easier to maintain than traditional CPU clusters. GPU-based algorithms have been developed for a variety of tasks in radiotherapy (Jia *et al* 2010a, 2010b, Gu *et al* 2009, 2010, Jacques *et al* 2008, Hissoiny *et al* 2009). They have been used for dose calculations by Hissoiny *et al* (2009) and Jacques *et al* (2008), who implemented superposition convolution algorithms for dose calculation on GPUs, and by Gu *et al* (2009), who explored the use of GPUs for a finite size pencil beam model. Moreover, a Monte Carlo code for coupled electron–photon transport was implemented on a GPU architecture by Jia *et al* (2010b). They reported speed gains up to a factor of 6.6. However, to date, to our knowledge no attempt to implement a track-repeating algorithm on a GPU architecture has been reported in the literature.

In this paper, we report our recent development of the FDC track-repeating algorithm for proton therapy on a GPU architecture under the computer unified device architecture (CUDA) platform developed by NVIDIA (2009). The code has been implemented on a single commercially available graphic card with two GPUs. The GPU-based FDC has been benchmarked against the CPU-based FDC code and the full Monte Carlo GEANT4, since the latter has been validated for proton therapy (Aso *et al* 2005, Paganetti *et al* 2008).

2. Methods

2.1. GEANT4

We used the GEANT4 tool kit (version 4.8.3) (Agostinelli *et al* 2003, Allison *et al* 2006) to generate the database of pre-calculated proton histories and to generate the reference dose sample. The physics models in this setup included proton energy loss via the continuous slowing-down approximation for secondary electrons with energy below Tc. Tc is material dependent and calculated from a particle distance range set to 0.1 mm. This translates to 83.6 keV and 250 eV energy cutoffs for electrons in water and air, respectively. In this work we utilized the low-energy parametrized model (Chauvie *et al* 2004), which takes into account atomic and shell effects and is applicable down to 250 eV. It uses the Be the–Bloch formula to calculate hadron ionization down to 2 MeV, and an ICRU 49 parametrization (ICRU 1993) in the range 1 keV to 2 MeV. Below 1 keV the free electron gas approach is utilized. The energy straggling was calculated with a Gaussian distribution with Bohr's variance (ICRU 1993) for distances long enough for the approximation to be accurate. For short distances a simple model of the atom was used (GEANT4 2008).

The multiple Coulomb scattering for protons was estimated with a condensed simulation algorithm, in which the global effects of the collisions are estimated at the end of a track segment. It uses model functions to determine the angular and spatial distributions, which were chosen to reproduce the distributions of the Lewis theory (Lewis 1950).

For elastic hadronic interactions the low-energy parametrized model was implemented. Inelastic interactions were simulated with a pre-equilibrium model in the range of interest of our simulations (0–250 MeV). The model is based on Griffin's semi-classical description of composite nucleus decay (Griffin 1966, Gudima *et al* 1983, Lara and Wellisch 2000). A more detailed description of the models used in proton therapy can be found elsewhere (Jarlskog and Paganetti 2008).

2.2. Fast dose calculator

The FDC algorithm utilizes a pre-generated database of the histories of particles produced by a proton impinging on a water phantom. In our study, we have generated the database using the Monte Carlo code GEANT4. Each particle trajectory is broken into steps, and for each step the direction, length and energy loss is stored. The FDC calculates dose distributions in heterogeneous anatomies, by re-tracing proton tracks. This re-tracing is achieved by scaling the length and scattering angle of each step according to the material in the non-water medium. In this study the database of proton histories was generated by simulating 121 MeV protons impinging on a 510 × 510 × 2500 mm^3 water phantom with a 3 × 3 × 2500 segmentation along the (x, y, z) axes. Further details of the FDC code were described previously (Yepes *et al* 2009a, 2009b).

In this work the dose and deposited energy calculated with GEANT4 and FDC, as described previously (Yepes *et al* 2009a, 2009b, 2010), were used as the basis of comparison with prediction from the GPU-based FDC (GFDC). Results from the CPU version of the FDC reported in this work utilized a database with 10 million pre-calculated proton histories in water, while the GPU-based version utilized a database with only 100 000 proton histories. Such a relative small database was chosen because no undesired correlations were observed with such a reduced number of 100 000 proton histories (see section 3). A small database should make the algorithm easier to install on various platforms.

2.3. CUDA implementation

The GFDC was developed using the CUDA software platform (NVIDIA 2009) on a general-purpose GPU on a single graphics card (GEFORCE GTX 295, NVIDIA, Santa Clara, CA) with 1.79 GB of global memory as the hardware platform. That GPU card holds two GPUs, with each unit holding 240 GPU cores. The software environment calls for the generation of multiple computational threads. The total number of threads is defined by the programmer and can be as high as hundreds of thousands. Threads are divided into blocks, so that the total number of threads is the number of blocks (N_B) multiplied by the number of threads per block (N_T). The number of threads should be a multiple of 32 for optimal performance because of the hardware configuration of the GPUs. Since the track-repeating algorithm is inherently highly parallel, each thread is treated as an independent computational unit. Each unit re-traced one of the proton histories from the database of pre-calculated histories.

A flowchart of the GPU implementation of the FDC is shown in figure 1. The code was split into two sections to be executed on the CPU and on the GPU, with the GPU being called from the CPU code. After the program initialization on the CPU, it first read the material and geometry information from configuration files. That information was then stored in the GPU global memory accessible to the GPU code. The material information corresponds to the scaling parameters of the step length and the scattering angle for all the materials to be used in the calculation. The geometry information, derived from the patient CT scan, consists of a three-dimensional array with a material index for each voxel.

After this initial stage, a loop over a pre-determined number of iterations was initiated. For each iteration, a number histories equal to the number of thread blocks (N_B) was read from the database. In addition an array with phase-space information for $N_T \times N_B$ incident protons to be simulated was generated. This proton phase space can be read from a file or generated randomly by the program within certain controllable ranges in energy and direction. In the results present in this work, the phase space was generated with fixed energy and direction but a spread in the position transverse to the beam (see section 2.4). Once the phase-space array was generated and the proton histories read, both pieces of information were transferred from the CPU to the GPU global memory, where they could be accessed by all the $N_T \times N_B$ GPU threads. Since only N_B pre-calculated proton histories were made available for $N_T \times N_B$ threads, the same proton history was utilized N_T times for various positions of the incident protons. However, since trajectories from a given history traversed different areas of the heterogeneous phantoms, the results from the re-tracing of the same history were expected to be statistically independent. Statistical uncertainties are calculated with two alternative methods, as explained in section 2.3, to test whether different trajectories are statistically independent.

After the operations to initialize an iteration were completed, the GPU code was invoked from the CPU code through a special C-language function termed kernel. The kernel was executed $N = N_T \times N_B$ times on the GPU engine in parallel on independent threads. The GPU code was subdivided into two main tasks: (1) finding the database history to be used and selecting the trajectory and step where the track-repeating algorithm should start and (2) re-tracing the pre-calculated proton history through the heterogeneous phantom.

The deposited energy generated by the different threads was tallied in a large three-dimensional grid with the same number of voxels as the heterogeneous phantom. The tally grid was defined in the GFDC as a one-dimensional array and was placed in the GPU global memory. This approach minimized the amount of memory utilized for tallying, which was significant for large grids. However utilizing a common tally for all threads required a function which blocked access to the memory location while a particular thread updated the information stored in it. Blocking a certain memory location forced other threads, trying to update the same voxel tally, to wait until the operation was completed. Such a mechanism slowed down the algorithm when competing threads attempted to access the same memory location simultaneously. A second tallying array, with the same dimension as the tally for the deposited energy, was utilized to store the sum of the squares of the deposited energy. The second array was necessary to estimate the history-by-history statistical uncertainties (see the next section). At the end of the execution the energy deposited in each voxel was converted to absorbed dose by dividing by the mass of the voxel. The number of blocks (N_B) and threads (N_T) were optimized to minimize the execution time.

The graphics card used in this study housed two GPUs. In order to maximize the performance of the card, two CPU threads were defined in the code, with each thread handling the operation and data transfer to one of the GPUs. Each CPU thread read from the same database; however, they read different pre-calculated proton histories. At the end of the execution of the two CPU threads, the absorbed dose from both threads was combined.

2.4. Statistical uncertainties and dose distribution comparison

Standard methods (Chetty *et al* 2007) were used to calculate statistical uncertainties and to investigate the effects of using the same 100 000 pre-calculated proton histories as many as 250 times. The batch method consists in comparing the results of multiple calculations, or batches, performed with uncorrelated phase space files and random number sequences. For this method, the estimate of uncertainty of the dose, D, is given by

$$\sigma_D = \sqrt{\frac{\sum_{i=1}^{n}(D_i - \overline{D})^2}{n(n-1)}}, \tag{1}$$

where n is the number of independent batches or runs, D_i is the scored dose in batch i and $\bar{D}$ is the mean value of the absorbed dose over all the batches. The sample size is given by the number of batches or independent calculations. In the history-by-history method, where a history dose corresponds to the absorbed dose produced by a single proton impinging on the phantom, the statistical uncertainty is given by

$$\sigma_D = \sqrt{\frac{1}{N-1}\left(\frac{\sum_{i=1}^{n} D_i^2}{N} - \left(\frac{\sum_{i=1}^{n} D_i}{N}\right)^2\right)}, \tag{2}$$

where N is the number of primary histories and D_i is the contribution to the dose by independent history i. Whereas the history-by-history approach may be distorted by hidden correlations, the batch method uncertainties would not because each batch is generated with completely independent databases and random numbers. If the estimate of the uncertainties is biased by hidden correlations, we expect a deviation from the $1/\sqrt{N}$ behavior for the history-by-history uncertainties. Moreover, the estimated uncertainty for the history-by-history approach will be lower than the unbiased correlations. On the other hand, hidden correlations due a small database are not expected to affect the batch estimates. Thus, comparing the results of these two approaches tests the feasibility of using the same proton history multiple times for these types of calculations.

The mean dose uncertainty was defined as the average of σ_D /D over all voxels with a dose larger than half the maximum dose, with σ_D calculated with equations (1) and (2).

To quantify the dosimetric accuracy of the FDC and GFDC dose distributions, we compared them to a distribution from GEANT4 simulations for the same field and voxelized phantom. The figure of merit used to quantify the dosimetric accuracy was the gamma index, Γ (Low *et al* 1998). This method of evaluating the distance to agreement and the dose difference of the sample case versus the reference case is widely used in the comparative analysis of dose distributions in radiotherapy. Two distributions are typically considered to agree well when at least 99% of the voxels, j, have values of Γ_j smaller than unity. GEANT4, which was previously validated for applications in proton therapy (Aso *et al* 2005, Paganetti *et al* 2008), provided reference dose distribution in this study, against which dose distributions from the FDC and the GFDC were compared. The maximum acceptable differences in dose and spatial distance used to calculate the Γ index in this study were 3% and 3 mm, respectively.

2.5. Patient anatomy and radiation field

The geometric model was represented as a voxelized phantom based on the computed tomography (CT) images of the thoracic region of a patient who had been treated for lung cancer at The University of Texas M D Anderson Cancer Center. The phantom contained 6064 305 voxels, each having dimensions of 1 × 1 × 2.5 mm^3. Each voxel was assigned a material composition and a mass density that corresponded to the Hounsfield unit value in the CT scan for that voxel, following the approach described elsewhere (Newhauser *et al* 2008). The thoracic region was selected because the thorax is highly inhomogeneous. It is in such inhomogeneous areas that pencil-beam algorithms are least reliable.

We have simulated a mono-energetic proton field of 121 MeV and a circular cross section of 4 cm radius. The energy of the beam was selected so as to traverse a significant fraction of the lung and, therefore, test the algorithm running on GPUs stringently. The energy and field characteristics were not selected to maximize the dose to the tumor or to minimize the dose to healthy tissue, as this was a generic test field, not a clinically realistic field.

3. Results

Figure 2 shows the distribution of deposited energy versus depth (y-axis) in the heterogeneous phantom, plotted along the beam central axis, (i.e. $x = 0$ and $z = 0$) and 1.5 cm and 3.0 cm lateral to the central beam axis, as predicted by GEANT4, FDC and GFDC. In addition, the percent differences in deposited energies between the track-repeating algorithm (FDC and GFDC) and GEANT4 are plotted along the same axis. Plotting the deposited energy rather than dose was chosen to better show the effects of the inhomogeneity of the anatomy. The GFDC reproduces the results from the CPU version of the FDC code, and the differences can be attributed to statistical fluctuations, since different pre-calculated databases of proton histories were utilized. In addition good agreement was observed between the deposited energies calculated by GEANT4 and GFDC. Figure 3 shows the cross-field profiles in the vertical direction for the isocenter ($x = 0$ and $y = 0$) and 7.5 cm posterior and anterior relative to the isocenter ($x = 0$ and $y = \pm 7.5$ cm). Each profile is calculated along a five voxel thick line. The cross-field profiles are depicted for three penetration depths to illustrate the agreement between the three approaches. Agreement was excellent for both profiles, with the largest discrepancy less than 3%.

The rest of the results comprise comparisons of dose rather than deposited energy. The difference between the GFDC- and GEANT4-calculated doses for each voxel in the anatomic phantom divided by the maximum GEANT4 voxel dose has a RMS value of 0.5%. From that value we conclude that the GFDC reproduces the dosimetric accuracy of GEANT within 1%.

A more comprehensive comparison was obtained by calculating the Γ-index of the FDC and GFDC relative to GEANT4 for each voxel. The Γ index results are presented in figure 4 as the complementary cumulative distribution function such that the ordinate represents the probability that Γ will be greater than the value of the abscissa. Both the GFDC and the FDC have essentially identical distributions, showing that the GPU-based FDC reproduces the results from the CPU-based version. Moreover, less than 0.01% of the voxels have Γ values greater than unity. Thus, the dose distributions from the FDC and the GFDC are in good agreement with the reference dose distribution from GEANT4.

As explained above, the GFDC utilized a database with 100 000 pre-calculated proton histories and re-traces each history multiple times. In order to verify that such re-cycling of proton histories does not produce undesired statistical correlations, the mean dose uncertainties were calculated with the history-by-history approach and with multiple batches. The batch method utilized six independent batches or runs. Results were identical if the number of batches was reduced to three. Figure 5 shows the results of the test of re-cycling the proton history database. In the figure, the lines represent a function $f(N)=C/\sqrt{N}$, where C is adjusted for the function to go through the point with the lowest N for each of the two curves. As can be seen in figure 5, the measured points for both methods follow the*f (N)* function. If correlations were present, we would expect a deviation from $1/\sqrt{N}$. Moreover, the fact that the batch uncertainties, which should not be affected by inter-batch correlations, are smaller than the history-by-history uncertainties demonstrates the absence of undesired statistical correlations introduced by the use of multiple proton histories. As can be seen, uncertainties calculated with the history-by-history method are around 50% higher than those from the batch approach.

The calculation time per proton history was found to depend on the number of blocks (N_B) and the number of threads per block (N_T). The number of blocks and threads per block were varied within the range allowed by the hardware in order to maximize the algorithm performance. The fastest calculation times were obtained for $N_B = 500$ and $N_T = 320$, for which 184 525 proton histories were processed per second utilizing the two GPUs on the graphics card. The CPU-based FDC on one CPU processed 2445 proton histories per second. Therefore the implementation of the FDC algorithm on a GPU card alone achieved a speedup of a factor of 75.5 with respect to the CPU-based implementation.

Storing the error in the tally arrays was found to increase the calculation time by 18%. In the results reported here errors were not stored. The rationale being that in a clinical environment, error is rarely reported for each voxel.

The upper abscissa of figure 5 shows the calculation time on the GPU card as a function of the statistical uncertainties. With the batch method a mean statistical dose uncertainty of 1% was achieved in less than 1 min, while around 2 min were required for the history-by-history approach.

4. Discussion

The good agreement between dose distributions calculated with the GFDC versus the GEANT4 and FDC codes suggests the feasibility of the GFDC to calculate dose distributions in proton radiotherapy as accurately as general-purpose Monte Carlo programs. While preserving accuracy, the GFDC reduced computational times by a factor of 75 with respect to a CPU-based FDC track-repeating algorithm.

Speed gains of 6.6 for a GPU-based Monte Carlo code relative to its CPU-based version were reported in the literature (Jia 2010b). The limited gains of 6.6 for the MC code are thought to be due to the nature of the GPU architecture. On a GPU, the algorithm is executed in multiple threads running in parallel. Threads must run in groups for best performance. Branches in the code do not impact performance provided all threads of a given group follow the same execution path. This may become a significant limitation for any inherently divergent task, like Monte Carlo simulations. It is likely that the speed gains seen for the GFDC were large because of the simpler logic of a track-repeating algorithm, as compared to a full Monte Carlo. A simpler logic should generate threads which follow closer execution paths.

In our current GPU algorithm all the threads in a given group were fed with the same proton history from the database of pre-calculated histories. Even though each history is re-traced in different areas of the phantom, and thus produces statistically independent results, this seems to minimize the logic path divergence for the various threads in a group. When threads in a given group were fed with different proton histories, the execution times increased by about 50%.

The CPU-based version used for the results on pelvis anatomy (Yepes *et al* 2009a, 2009b) was limited to dose calculations in voxelized geometry. Results reported for the thorax anatomy included an aperture and range compensator (Yepes *et al* 2010). For those results the code included a package from ROOT (Brun and Rademakers 1997) to describe arbitrary geometries. In the GFDC, reported in this study, this feature to describe arbitrary geometries has not been implemented yet, due to the difficulties to port the corresponding ROOT classes to GPUs. The GFDC version is restricted to dose calculation in voxelized geometries.

Currently, the large computational requirement for Monte Carlo simulations makes their use difficult for routine clinical proton radiotherapy treatment planning. At present, it is only practical to calculate such treatment plans with large computer clusters, which are unavailable in most clinics. This obstacle also hinders the opportunities for studies in which whole-body dose reconstructions are needed for large numbers of patients, e.g., in clinical trials or radiation epidemiology studies. The results of the present study suggest that it may be feasible to overcome this obstacle with the GFDC approach, although additional development and testing of the codes will still be needed.

The findings of this study indicate that it may be feasible for the GFDC to calculate the dose distribution for an entire proton radiotherapy treatment plan for lung cancer with 1% statistical dosimetric uncertainty on a desktop-size system equipped with 2 GPU cards in about 1 min. Future studies to test this hypothesis should include multiple clinically realistic fields, a plurality of patients and sites. The code should also be extended to make it capable of handling arbitrary geometries to include the patient-dependent beam shaping elements.

5. Conclusions

In conclusion, the GFDC is a promising implementation of a track-repeating code for proton radiotherapy dose calculations using GPU architecture. The dosimetric accuracy of the GFDC algorithm was validated by comparing the results with those generated with GEANT4 Monte Carlo and CPU-based FDC simulations. The GFDC can calculate the dose distribution produced by a 121 MeV proton beam in a thoracic geometry with 1% accuracy in one minute with a graphics card with two GPUs. Only 0.01% of the phantom voxels had a Γ index larger than 1, with the Γ index calculated with a full Monte Carlo as the reference distribution. The implementation of the track-repeating algorithm on a GPU architecture may allow for real-time dose calculations for proton radiotherapy with a desktop-size computer system equipped with multiple GPU cards.

Acknowledgments

This work was supported in part by the John & Ann Doerr Fund for Computational Biomedicine under contract CABC-2006-5-PY; by the Rice Computational Research Cluster funded by the National Science Foundation under Grant CNS-0421109 and a partnership between Rice University, Advanced Micro Devices and Cray Inc. This work was supported in part by the National Cancer Institute (award 1R01CA131463-01A1) and by the Fogarty International Center (award K01TW008409). We are thankful to Wayne Newhauser for his careful reading of the manuscript. We are indebted to Vivek Sarkar and Yonghong Yan for discussions on the technical aspects of the GPU implementation. The content is solely the responsibility of the authors and does not necessarily represent the official views of the sponsors.

References

Agostinelli S, et al. GEANT4—a simulation toolkit. Nucl Instrum Meth A 2003;506:250–303.

Allison J, et al. GEANT4 developments and applications. IEEE Trans Nucl Sci 2006;53:270–8.

Aso T, Kimura A, Tanaka S, Yoshida H, Kanematsu N, Sasaki T, Akagi T. Verification of the dose distributions with GEANT4 simulation for proton therapy. IEEE Trans Nucl Sci 2005;52:896–901.

Brun R, Rademakers F. ROOT—an object oriented data analysis framework. Nucl Instrum Meth A 1997;389:81–6.

Chauvie S, et al. GEANT4 low energy electromagnetic physics. IEEE Nuclear Science Symp Conf Record 2004;3:1881–5.

Chetty JT, et al. Report of the AAPM Task Group No. 105: issues associated with clinical implementation of Monte Carlo-based photon and electron external beam treatment planning. Med Phys 2007;34:4818–53. [PubMed: 18196810]

Ciangaru G, Polf JC, Bues M, Smith AR. Benchmarking analytical calculations of proton doses in heterogeneous matter. Med Phys 2005;32:3511–23. [PubMed: 16475750]

Deasy JO. A proton dose calculation algorithm for conformal therapy simulations based on Moliere theory of lateral deflections. Med Phys 1998;25:476–83. [PubMed: 9571613]

Dutreix A. When and how can we improve precision in radiotherapy? Radiother Oncol 1984;2:275–92. [PubMed: 6522641]

Fasso, A.; Ferrari, A.; Ranft, J.; Sala, PR. FLUKA: a multi-particle transport code. 2005. CERN-2005–10, INFN/50_XT/11

Fippel M. Fast Monte Carlo dose calculation for photon beams based on the VMC electron algorithm. Med Phys 1999;26:1466–75. [PubMed: 10501045]

Fippel M, Soukup M. A Monte Carlo dose calculation algorithm for proton therapy. Med Phys 2004;31:2263–73. [PubMed: 15377093]

GEANT4. Physics Reference Manual. 2008. Version: GEANT4 9.2

Giebeler A, Zhu XR, Titt U, Lee A, Tucker S, Newhauser. Uncertainty in dose per monitor unit estimates for passively scattered proton therapy, Part I: the role of FCSPS in the prostate. Phys Med Biol. 2010 submitted.

Goitein M, Busse J. Immobilization error: Some theoretical considerations. Radiology 1975;117:407–12. [PubMed: 1178875]

Griffin JJ. Semiclassical model of intermediate structure. Phys Rev Lett 1966;17:478–81.

Gu X, Choi D, Men C, Pan H, Majumdar A, Jiang SB. GPU-based ultra fast dose calculation using a finite size pencil beam model. Phys Med Biol 2009;54:6287–97. [PubMed: 19794244]

Gu X, Pan H, Liang Y, Castillo R, Yang D, Choi D, Castillo E, Majumdar A, Guerrero T, Jiang SB. Implementation and evaluation of various demons deformable image registration algorithms on a GPU. Phys Med Biol 2010;55:207–19. [PubMed: 20009197]

Gudima KK, Mashnik SG, Toneev VD. Cascade-exciton model of nuclear-reactions. Nucl Phys A 1983;401:329–61.

Hissoiny S, Ozell B, Despres P. Fast convolution-superposition dose calculation on graphics hardware. Med Phys 2009;36:1998–2005. [PubMed: 19610288]

Hong L, Goitein M, Bucciolini M, Comiskey R, Gottschalk B, Rosenthal S, Serago C, Urie M. A pencil beam algorithm for proton dose calculations. Phys Med Biol 1996;41:1305–30. [PubMed: 8858722]

Hotta K, Kohno R, Takada Y, Hara Y, Tansho R, Himukai T, Kameoka S, Matsuura T, Nishio T, Ogino T. Improved dose-calculation accuracy in proton treatment planning using a simplified Monte Carlo method verified with three-dimensional measurements in an anthropomorphic phantom. Phys Med Biol 2010;55:3545–56. [PubMed: 20508320]

ICRU (International Commission on Radiation Units and Measurements). Report 49. Bethesda, MD: ICRU; 1993. Stopping Powers and Ranges for Protons and Alpha Particles.

Jacques, R.; Taylor, R.; Wong, J.; McNutt, T. Towards real-time radiation therapy: GPU accelerated superposition/convolution. High-Performance Medical Image Computing and Computer Aided Intervention Workshop HP-MICCAI; 2008; 2008.

Jarlskog CZ, Paganetti H. Physics settings for using the Geant4 Toolkit in proton therapy. IEEE Trans Nucl Sci 2008;55:1018–25.

Jia X, Gu X, Sempau J, Choi D, Majumdar A, Jiang SB. Development of a GPU-based Monte Carlo dose calculation code for coupled electron–photon transport. Phys Med Biol 2010b;55:3077–86. [PubMed: 20463376]

Jia X, Lou Y, Li R, Song WY, Jiang SB. GPU-based fast cone beam CT reconstruction from undersampled and noisy projection data via total variation. Med Phys 2010a;37:1757–60. [PubMed: 20443497]

Koch N, Newhauser WD, Titt U, Gombos D, Coombes K, Starkschall G. Monte Carlo calculations and measurements of absorbed dose per monitor unit for the treatment of uveal melanoma with proton therapy. Phys Med Biol 2008;53:1581–94. [PubMed: 18367789]

Kohno R, Takada Y, Sakae T, Terunuma T, Matsumoto K, Nohtomi A, Matsuda H. Experimental evaluation of validity of simplified Monte Carlo method in proton dose calculations. Phys Med Biol 2003;48:1277–88. [PubMed: 12812446]

Lara, V.; Wellisch, JP. Pre-equilibrium and equilibrium decays in Geant4. Proc. Computing in High Energy and Nuclear Physics; Padova, Italy. 2000. p. 52-5.

Lewis HW. Multiple scattering in an infinite medium. Phys Rev 1950;78:526.

Li JS, Shahine B, Fourkal E, Ma CM. A particle track-repeating algorithm for proton beam dose calculation. Phys Med Biol 2005;50:1001–10. [PubMed: 15798272]

Low DA, Harms WB, Mutic S, Purdy JA. A technique for the quantitative evaluation of dose distributions. Med Phys 1998;25:656–61. [PubMed: 9608475]

Newhauser WD, Zheng Y, Taddei PJ, Mirkovic D, Fontenot J, Giebeler A, Zhang R, Titt U, Mohan R. Monte Carlo proton radiation therapy planning calculations. Trans Am Nucl Soc 2008;99:63–4.

NVIDIA. NVIDIA CUDA (Compute Unified Device Architecture) Programming Guide. 2009.

Orton CG, Mondalek PM, Spicka JT, Herron DS, Andres LI. Lung corrections in photon beam treatment planning: are we ready? Int J Radiat Oncol Biol Phys 1984;10:2191–99. [PubMed: 6439697]

Paganetti H, Jiang H, Parodi K, Slopsema R, Engelsman M. Clinical implementation of full Monte Carlo dose calculation in proton beam therapy. Phys Med Biol 2008;53:4825–53. [PubMed: 18701772]

Papanikolaou N, Battista J, Boyer A, Kappas C, Klein E, Mackie T, Sharpe M, Van Dyk J. Tissue inhomogeneity corrections for megavoltage photon beams. AAPM Report No 85. 2004

Pelowitz, DB., editor. MCNPXTM User's Manual Version 2.6.0. Los Alamos, NM: Los Alamos National Laboratory; 2007.

Petti PL. Differential-pencil-beam dose calculations for charged particles. Med Phys 1992;19:137–49. [PubMed: 1320182]

Russell KR, Grusell E, Montelius A. Dose calculations in proton beams: range straggling corrections and energy scaling. Phys Med Biol 1995;40:1031–43. [PubMed: 7659728]

Schaffner B, Pedroni E, Lomax A. Dose calculation models for proton treatment planning using a dynamic beam delivery system: an attempt to include density heterogeneity effects in the analytical dose calculation. Phys Med Biol 1999;44:27–41. [PubMed: 10071873]

Schneider U, Schaffner B, Lomax AJ, Pedroni E, Tourovsky A. A technique for calculating range spectra of charged particle beams distal to thick inhomogeneities. Med Phys 1998;25:457–63. [PubMed: 9571611]

Soukup M, Fippel M, Alber M. A pencil beam algorithm for intensity modulated proton therapy derived from Monte Carlo simulations. Phys Med Biol 2005;50:5089–104. [PubMed: 16237243]

Stewart JG, Jackson AW. The steepness of the dose response curve both for tumor cure and normal tissue injury. Laryngoscope 1975;85:1107–11. [PubMed: 807783]

Szymanowski H, Oelfke U. Two-dimensional pencil-beam scaling: an improved proton dose algorithm for heterogeneous media. Phys Med Biol 2002;47:3313–30. [PubMed: 12375823]

Taddei PJ, Mirkovic D, Fontenot JD, Giebeler A, Zheng Y, Kornguth D, Mohan R, Newhauser WD. Stray radiation dose and second cancer risk for a pediatric patient receiving craniospinal irradiation with proton beams. Phys Med Biol 2009;54:2259–75. [PubMed: 19305045]

Titt U, Sahoo N, Ding X, Zheng Y, Newhauser WD, Zhu XR, Polf JC, Gillin MT, Mohan R. Assessment of the accuracy of an MCNPX-based Monte Carlo simulation model for predicting three-dimensional absorbed dose distributions. Phys Med Biol 2008;53:4455–70. [PubMed: 18670050]

Tourovsky A, Lomax AJ, Schneider U, Pedroni E. Monte Carlo dose calculations for spot scanned proton therapy. Phys Med Biol 2005;50:971–81. [PubMed: 15798269]

Vadapalli R, Yepes P, Newhauser W, Licht R. Grid-enabled treatment planning for proton therapy using Monte Carlo simulations. Nucl Technol. 2010 at press.

Yepes P, Randeniya S, Taddei P, Newhauser W. A track repeating algorithm for fast Monte Carlo dose calculations of proton radiotherapy. Nucl Technol 2009a;168:334–7.

Yepes P, Randeniya S, Taddei P, Newhauser W. Monte Carlo fast dose calculator for proton radiotherapy: application to a voxelized geometry representing a patient with prostate cancer. Phys Med Biol 2009b;54:N21–8. [PubMed: 19075361]

Yepes P, Brannan T, Huang J, Mirkovic D, Newhauser WD, Taddei PJ, Titt U. Application of a fast proton dose calculation algorithm to a thorax geometry. Radiat Meas. 2010 at press. 10.1016/j.radmeas.2010.05.022

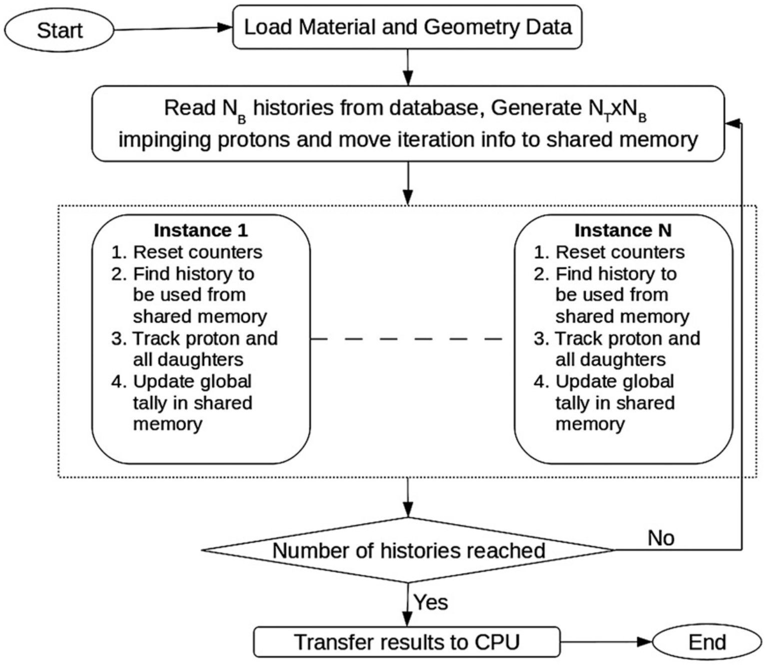
Start
Load Material and Geometry Data
Read N_B histories from database, Generate N_TxN_B impinging protons and move iteration info to shared memory
Instance 1
1. Reset counters
2. Find history to be used from shared memory
3. Track proton and all daughters
4. Update global tally in shared memory
Instance N
1. Reset counters
2. Find history to be used from shared memory
3. Track proton and all daughters
4. Update global tally in shared memory
Number of histories reached
No
Yes
Transfer results to CPU
End

Figure 1.

The flow chart of the GPU-based version of the track-repeating fast dose calculator (GFDC). The kernel, code running on the GPUs, is bounded by the pointed-line rectangle, and is run in parallel by $N = N_T \times N_B$ different CUDA threads, where N_B and N_T are the number of blocks and threads per block, respectively.

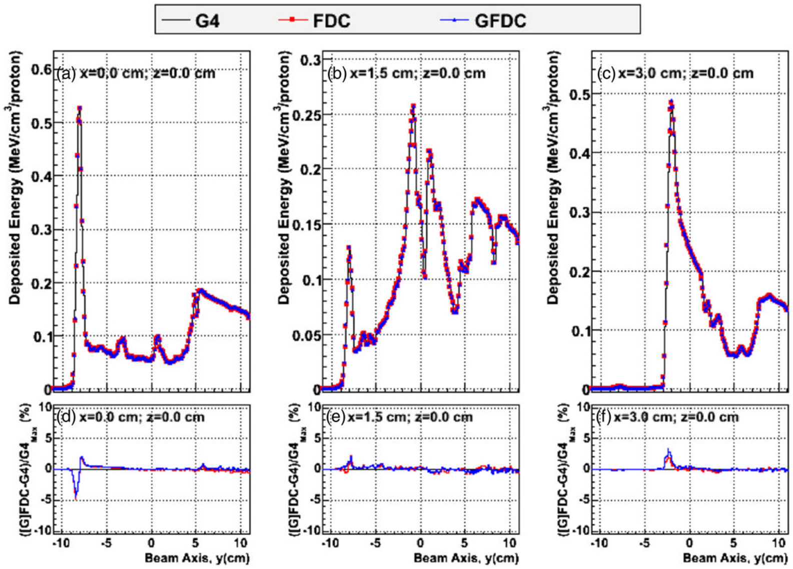

Figure 2.

Deposited energy profiles in voxels along the beam axis, *y*, for *z* = 0 and (a) *x* = 0, (b) *x* = 1.5 cm and (c) *x* = 3.0 cm. The *y*-axis runs from posterior to anterior of the patient. Distributions were calculated with GEANT4 (G4: black line), FDC (red circles) and GFDC (blue triangles). The differences in dose between GEANT4 and FDC (red line) and GFDC (blue line) and GEANT4 divided by the maximum GEANT4 dose are shown in panels (d), (e) and (f) for *x* = 0, *x* = 1.5 cm and *x* = 3.0 cm, respectively.

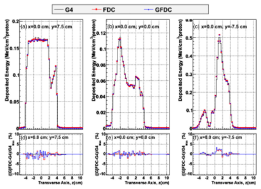

Figure 3.

Cross-field profiles of deposited energy along the patient's vertical axis (z), along the central beam axis (i.e. $x = 0$) for (a) $y = 7.5$ cm, (b) $y = 0$ cm and (c) $y = -7.5$ cm, where the y-axis runs from anterior to posterior of the patient. Distributions were calculated with GEANT4 (G4: black line), FDC (red circles) and GFDC (blue triangles). The GEANT4–FDC (red line) and GEANT4–GFDC (blue line) deposited energy differences divided by the maximum GEANT4 deposited energy is shown in panels (d), (e) and (f) for $y = 7.5$ cm, $y = 0$ cm and $y = -7.5$ cm, respectively.

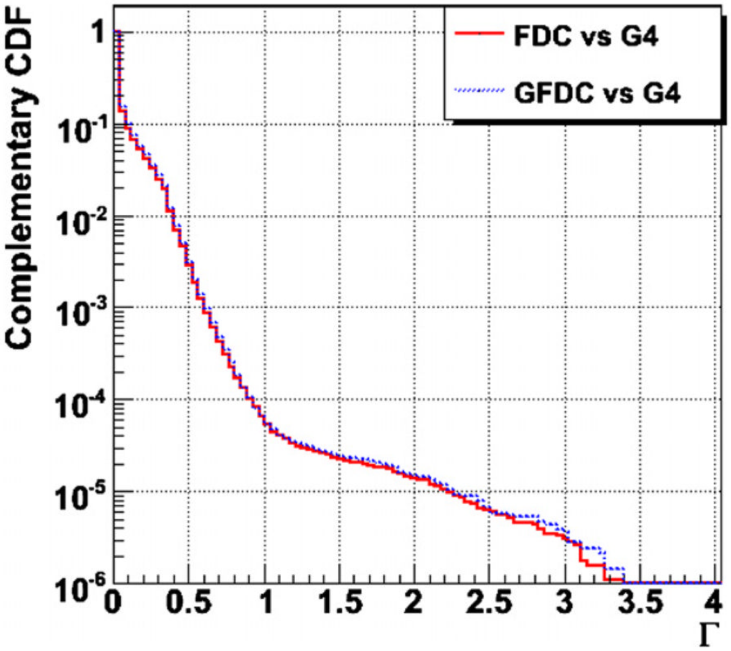

Figure 4.

The complementary cumulative distribution function (CDF) of the Γ index for FDC and GFDC using GEANT4 as the best estimate of the true dose distribution in the heterogeneous phantom representing a thoracic cancer patient. The gamma function was calculated for all non-air voxels in the geometric model.

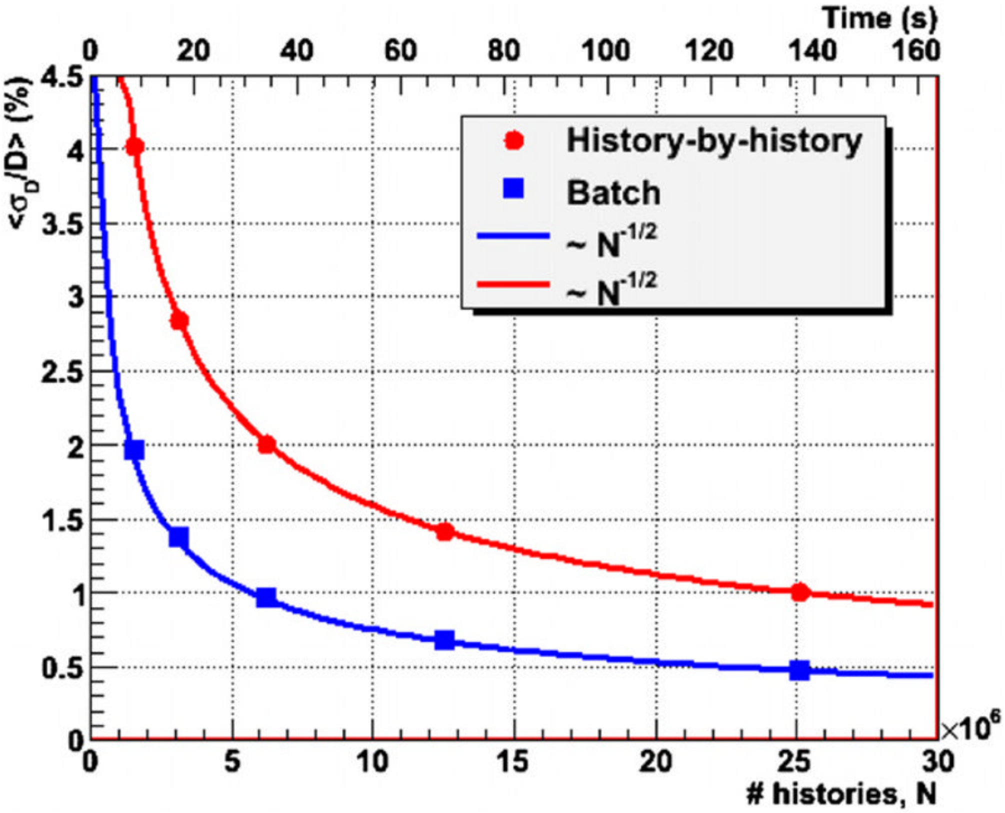

Figure 5.

The mean statistical uncertainties of GFDC dose distributions calculated with the history-by-history and the batch approaches as a function of the number of proton histories (*N*) and calculation times. The lines are functions proportional to $N^{-1/2}$ adjusted to cross the point with the lowest *N* for each of the methods.