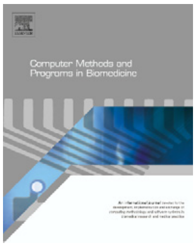


journal homepage: www.intl.elsevierhealth.com/journals/cmpb

Towards real-time radiation therapy: GPU accelerated superposition/convolution

Robert Jacques[a,*], *Russell Taylor*[b], *John Wong*[a], *Todd McNutt*[a]

[a] *School of Medicine, Johns Hopkins University, Baltimore, MD 21231-2410, USA*
[b] *Computer Science Department, Johns Hopkins University, Baltimore, MD 21218, USA*

ARTICLE INFO

Article history:
Received 13 February 2009
Received in revised form
16 April 2009
Accepted 2 July 2009

Keywords:
Convolution/superposition
Radiation therapy planning
Graphics processing unit (GPU)
Inverse planning
Adaptive radiotherapy

ABSTRACT

We demonstrate the use of highly parallel graphics processing units (GPUs) to accelerate the superposition/convolution (S/C) algorithm to interactive rates while reducing the number of approximations. S/C first transports the incident fluence to compute the total energy released per unit mass (TERMA) grid. Dose is then calculated by superimposing the dose deposition kernel at each point in the TERMA grid and summing the contributions to the surrounding voxels. The TERMA algorithm was enhanced with physically correct multi-spectral attenuation and a novel inverse formulation for increased performance, accuracy and simplicity. Dose deposition utilized a tilted poly-energetic inverse cumulative–cumulative kernel, with the novel option of using volumetric mip-maps to approximate solid angle ray casting. Exact radiological path ray casting decreased discretization errors. We achieved a speedup of 34x–98x over a highly optimized CPU implementation.

1. Introduction

Computational performance is the limiting factor in radiation therapy treatment plan quality. Traditionally, improvements in treatment quality have been realized by faster hardware. But, “the free lunch is over” [1]. Instead of doubling in speed every 18 months, computers are doubling the number of processing cores. And as processors have become muti-core, the many-core architectures of graphics processing units (GPUs) have been gaining the flexibility to run general purpose algorithms. In order to realize the promised performance gains of this hardware, traditional serial algorithms must be replaced with parallel ones. We address the conversion of a key algorithm in radiation therapy treatment planning: superposition/convolution [2–4].

Superposition/convolution is a state of the art method of calculating the radiation dose delivered during external beam radiation therapy. Fast, accurate dose computation algorithms are important to radiation therapy planning as they are the only method of determining the dose delivered to a specific patient. Dose computation consists of two parts: a source model and a transport model. The source model provides the incident fluence. The transport model computes the dose that results from the incident fluence and is the performance bottleneck. The three main transport algorithms in order of increasing accuracy/decreasing performance are pencil beam, superposition/convolution and Monte Carlo. Superposition/convolution is the current clinical standard.

In recent years, treatment quality has been increased by the use of intensity modulation. This technique drastically increases the number of beam parameters. Therefore, a dosimetrist specifies an objective function that a treatment planning system optimizes using a dose calculation algorithm. However, the objective function currently does not fully

[*] *Corresponding author*. Tel.: +1 410 955 6980.
E-mail addresses: sandford@jhu.edu, rjacque2@jhmi.edu (R. Jacques), rht@cs.jhu.edu (R. Taylor), jwong35@jhmi.edu (J. Wong), tmcnutt1@jhmi.edu (T. McNutt).

0169-2607/$ – see front matter © 2009 Elsevier Ireland Ltd. All rights reserved.
doi:10.1016/j.cmpb.2009.07.004

capture the desires of the dosimetrist, leading to multiple optimizations with tweaked parameters. Therefore, while a single optimization may take 5 minutes for a set of 5 beams, the entire process may take several hours to produce a clinically acceptable plan. This limits both the quantity and quality of intensity modulated plans a clinic can treat.

This clinical workflow limitation extends to more complex techniques such as volumetric modulated arc therapy [5], intensity modulated arc therapy [6] and adaptive radiation therapy [7]. Furthermore, it prohibits real-time radiation therapy; the ability to scan, re-plan and treat every patient daily.

A deeper review of dose calculation in radiation therapy is available from Ahnesjo and Aspradakis [8].

1.1. *The transport model*

The superposition/convolution algorithm has been shown to produce an accurate calculation of the dose distribution [2–4]. It consists of two stages. First, the incident fluence is transported through the density representation of the patient to compute the total energy released per unit mass (TERMA) at each location. The TERMA, $T_E(r')$, of a particular energy E of at point r' is defined as the fluence of energy E, $\Psi_E(r')$, weighted by the density relative to water, $\rho(r')$, and linear attenuation, $\mu_E(r')$, at point r'.

$$T_E(r') = \frac{\mu_E(r')}{\rho(r')}\Psi_E(r'). \tag{1}$$

The linear attenuation coefficient, $\mu_E(r')$, is also dependent on the atomic material. However at mega-voltage energies Compton scattering dominates, which is dependent on electron density not material. Clinically, there is a piecewise linear relation between standard CT numbers and density for normal human tissues and non-Compton interactions are considered negligible. The fluence, $\Psi_E(r')$, at point r' of energy E is determined by the source focal point s and the incident fluence $\Psi_{E,0}(r')$ of energy E in the direction r'.

$$\Psi_E(r') = \frac{\Psi_{E,0}(r')}{\|r' - s\|^2} e^{\int_s^{r'} -\mu_E(t)dt}. \tag{2}$$

Then, superposition spreads this energy by a dose deposition kernel to determine the final dose at each location. To allow the dose deposition kernel to scale realistically with tissue inhomogeneities the radiological distance between points, $d_\rho(r,r')$, is used. This differentiates superposition from traditional convolution.

$$d_\rho(r, r') = \int_r^{r'} \rho(t)dt. \tag{3}$$

The dose at point r, $D(r)$, is computed from integrated over the TERMA volume weighting by the energy dependent dose deposition kernel, K_E. The standard collapsed cone kernel is indexed by radiological distance and relative angle, ω, between the point and the kernel axis, but lacks the geometric distance squared effect.

$$D(r) = \oiiint \sum_E T_E(r') K_E(d_\rho(r, r'), \omega(r, r')) \frac{1}{\|r - r'\|^2} dr' \tag{4}$$

However, typically the mono-energetic contribution of every voxel, as in (4), is not calculated. Instead, as in (7), a discrete set of ray angles, ω, and directions, ν, are chosen and integrated along using a single poly-energetic kernel. This is justified by the approximately exponential kernel fall-off and the distance squared effect drastically reducing the contribution of any single distant voxel. The distance squared effect is negated by the increase in volume of the solid angle the rays represent. Ray directions are chosen to balance geometric and kernel energy factors.

$$T(r') = \sum_E T_E(r'). \tag{5}$$

$$K(d_r, \omega) = \sum_E K_E(d_r, \omega). \tag{6}$$

$$D(r) \approx \sum_{\omega,\nu} \int T(r + t\nu) K(d_\rho(r, r + t\nu), \omega) dt. \tag{7}$$

Traditionally, TERMA has been calculated by casting a set of rays that deposit the incident fluence to volume according to (1) and (2). For numerical accuracy, approximately 4 rays should pass through each TERMA voxel, attenuation should begin at the patient's surface and the fluence to each voxel should be normalized by the total length of the rays that contributed to that voxel. This normalization removes the normal inverse square fluence drop-off due to a divergence source. It must therefore be reapplied, normally to the TERMA grid. However, a clinically acceptable speed enhancement is to not tilt the dose deposition kernel to align with the ray axis at each voxel. In this case it is more accurate to apply the divergence correction to the dose grid [9].

TERMA is strongly dependent on the beam spectrum. The spectrum is rotationally symmetric about the beam axis and hardens with depth in material. Traditionally, the attenuation is modeled using a lookup table which when combined with the linear attenuation lookup table has axes of depth, density and off-axis angle. This table also requires the use of a fixed step ray casting algorithm, which avoids evaluating the traditionally costly exponential at every step. Though clinically accepted for performance reasons, this lookup table assumes a homogenous media. This is incorrect as heterogeneous tissues preferentially attenuate different spectra. Furthermore, the fixed step size and discretized rays results in numerical artifacts.

The dose deposition kernel also has a dependence of the energy spectrum at each voxel. However, this effect is negligible and a single poly-energetic kernel has been shown to be sufficiently accurate for clinical use and is the current standard of care. Poly-energetic kernels are created by combining a spectrum of mono-energetic kernels, which are generated using Monte Carlo simulations that forces mono-energetic photons to interact at the center of a water sphere and tallying the dose deposited everywhere in that sphere [10,11].

Adapting algorithms to parallel or GPU architectures is not a new idea and several ray cast, ray trace and volumetric visualization algorithms have already been adapted. However, dose calculation is fundamentally interested in the interaction of a line with a volume while visualization algorithms are interested in a property of a line, such as its integral or maximum, making many algorithms inapplicable. Part of the traditional TERMA algorithm is superficially similar to volumetric ray casting though naïvely adapting the previous GPU implementation [12] is impossible and would fail to produce the correct result if possible. Dose deposition deals primarily with electron interactions and is therefore fundamentally different from visualization algorithms. Recently, Nucletron made an announcement [13] regarding GPU acceleration in their treatment planning system, though published details are not yet available.

2. Methods

We have adapted the superposition/convolution algorithm to the GPU using a combination of NVIDA's Compute Unified Device Architecture (CUDA) [14] software development environment and the Digital Mars' D programming language [15] for our implementation.

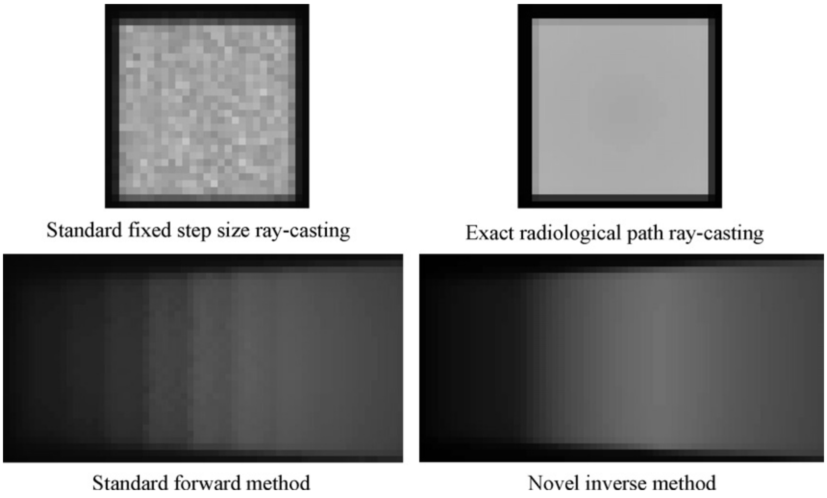
Superposition/convolution is a two-stage algorithm: first the TERMA is calculated and then the dose deposition kernel is superimposed. We replaced the standard fixed step size ray cast algorithm used in both stages with an exact radiological path method, similar to line rasterization [16]. This reduced the per ray per voxel memory accesses to 1 and eliminated related discretization artifacts (see Fig. 1, surface slice).

We enhanced the TERMA calculation with physically correct multi-spectral attenuation [17]. We implemented three methods: the standard forward method, a novel inverse method and a fast, approximate inverse method. The current clinical standard is fixed step fast, approximate forward method which lacks physically correct multi-spectral attenuation. We identify the use of a cached attenuation volume to accelerate the TERMA calculation. The superposition calculation was enhanced with the cumulative–cumulative kernel [18] and kernel tilting [16].

2.1. TERMA calculation

We first implemented the standard forward TERMA algorithm. This method requires overlapping scattered accumulation, which results in read–write conflicts when run in parallel. A read–write conflict occurs when threads simultaneously attempt to read, process and then write to the same memory, resulting in only the last update being recorded. This drastically reduces the effective number of contributing rays to each voxel to below the limit required for numerical accuracy. However, the divergent nature of the source may be utilized to create sets of rays guaranteed to be one voxel apart at all times. While individual sets are large enough to be efficiently run on the GPU, this serialization causes excessive GPU call overhead. The traditional 3D spectral attenuation lookup table exceeded the texture cache size, reducing performance. Performance was improved by individually attenuating separate spectral bins, using shared memory to reduce register usage. This reduced the lookup to a small 2D texture of linear attenuation coefficients parameterized by energy and density. This was enabled by the hardware accelerated exponential on the GPU. Accuracy was also improved as the spectrum was correctly attenuated in heterogeneous media. However, the number of rays required for numerical accuracy had an unclear relation with resolution (see Table 2), discretization effects were still evident (see Fig. 1, penumbra slice) and computation comprised a significant fraction of the total dose calculation time.

We avoided these issues by an inverse TERMA algorithm. While the general strategy of method inversion is well known in the radiation therapy community, we believe TERMA inversion to be novel. The new method is derived by rearranging equations (1), (2) and (5), separating the incidence fluence, $\Psi_{E,0}(r')$, into a spectral weight, w_E, and net fluence factor, $\Psi_0(r')$,



Standard fixed step size ray-casting

Exact radiological path ray-casting

Standard forward method

Novel inverse method

Fig. 1 – Example of TERMA step size (top, surface slice) and ray (bottom, penumbra slice) discretization artifacts of the standard method compared to the novel inverse method.

and defining an attenuation factor,

$$A(r') = \frac{1}{\|r' - s\|^2} \sum_E w_E(r') \frac{\mu_E}{\rho(r')} e^{\int_s^{r'} -\mu_E(t)dt}, \tag{8}$$

which casts a ray from each TERMA voxel back towards the source, gathering the net attenuation along the way. This allows for early ray termination when the patient boundary is reached, increasing both performance and accuracy. Then TERMA is calculated by multiplying the net attenuation for each voxel by the net incident fluence from the source towards that voxel:

$$T(r') = \Psi_0(r')A(r'). \tag{9}$$

While this is an $O(n^4)$ algorithm, as opposed to the $O(n^{\sim 3})$ forward method, each thread only writes to its own voxel, thus avoiding read–write conflicts. The algorithm allows for coalesced read access to texture memory which drastically enhances memory performance. Both algorithms had similar empirical performance over a variety of clinical resolutions (see Table 2) and either may compute the expensive attenuation volume, which provides a multiple order of magnitude performance improvement during interactive use or intensity modulation optimization, when only the fluence field is changed. A key advantage of the inverse method is the elimination of artifacts due to ray discretization. For interactive beam angle changes, a simpler variant without physically correct multi-spectral attenuation was implemented. This is the same approximation used in current clinical systems and resulted in a significant performance improvement.

2.2. *Superposition calculation*

Once TERMA has been calculated, superposition of a dose deposition kernel must be applied. Superposition has two standard formulations. The forward formulation spreads dose from a TERMA voxel to the surrounding dose voxels. This method requires calculating the dose to every patient voxel and suffers from read–write conflicts as multiple TERMA voxels contribute to every dose voxel.

The inverse kernel formulation gathers the contribution to a dose voxel from the surrounding TERMA voxels. This is computationally efficient as only the dose to the volume of interest is calculated. This is possible because the dose deposition kernel is invertible. Strictly speaking, use of kernel tilting in standard superposition breaks the invertible kernel assumption. However, given the distant source and the kernel's rapid fall-off, invertibility is still a reasonable assumption and is in clinical use.

However, the kernel's rapid fall-off also creates numerical sampling issues at typical clinical resolutions. There are two standard alternatives. The cumulative kernel (CK) [19] represents the dose deposition from a ray segment to a point.

$$CK(x, \omega) = \int_0^x K(t, \omega)dt. \tag{10}$$

The cumulative–cumulative kernel (CCK) [18] represents the dose deposition from a ray segment to a ray segment.

$$CCK(x, \omega) = \int_0^x CK(t, \omega)dt. \tag{11}$$

Both are derived from integrating the standard point to point kernel, $K(d_r, \omega)$, for particular radiological depths, d_r, and angles, ω.

$$\int_x^{x+\Delta x} K(v, \omega)dv = CK(x + \Delta x, \omega) - CK(x, \omega). \tag{12}$$

$$\int_0^{\Delta s} CK(x + u, \omega)du = CCK(x + \Delta s, \omega) - CCK(x, \omega). \tag{13}$$

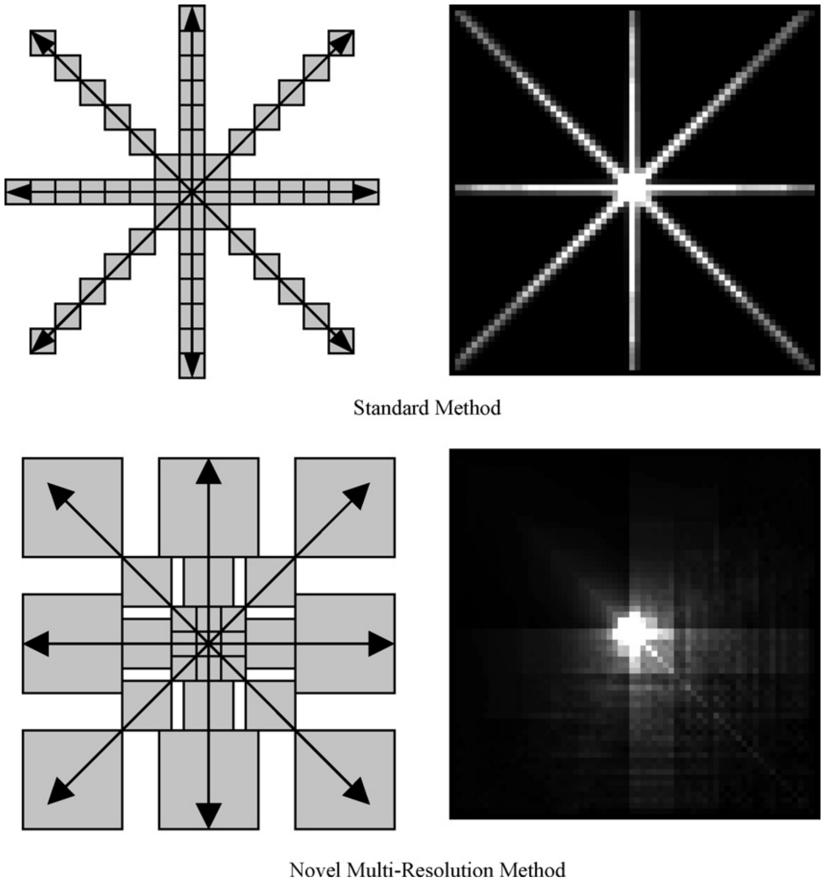
$$\int_0^{\Delta s} \int_{x+v}^{x+\Delta x+v} K(u, \omega)dudv = (CCK(x + \Delta s + \Delta x, \omega) - CCK(x + \Delta x, \omega)) - (CCK(x + \Delta s, \omega) - CCK(x, \omega)). \tag{14}$$

While more accurate, particularly at coarser resolutions, the CCK formulation has traditionally been 50% slower than the CK formulation. However, the GPU's texture unit, which caches memory accesses and provides dedicated linear interpolation hardware, allows the use of the CCK method with a negligible performance decrease.

An advantage of serial CPU implementations is the ability to reuse the ray cast index calculation, by moving all indexes one voxel over. This prevents kernel tilting, resulting in errors at large off-axis angles (see Table 3). Our GPU implementation does not have this optimization allowing both tilting and non-tilting kernels to be implemented. Kernel tilting has traditionally resulted in 300% loss in performance [17], but the GPU performance loss is only 19%.

2.3. *Multi-resolution superposition*

The ability to remain accurate at coarse resolutions, combined with the kernel's rapid fall-off presents the opportunity for a novel superposition algorithm, which uses a multi-resolution superposition to approximate each ray as a true solid angle. Unlike a ray, the width of a solid angle increases with geometric distance. In a discretized volume, a ray's width is proportional to the voxel's width. Therefore, by increasing the voxel width with geometric distance, a ray can approximate a solid angle. This also increases the step size in a logarithmic manner which increases the performance from $O(\omega DT^{1/3})$ to $O(\omega D\log(T^{1/3}))$; where ω is the number of angles, D is the number of dose voxels, and T is the number of TERMA voxels. Compared to the standard method, our proposed method exhibits an interesting accuracy trade-off (see Table 3). Small field artifacts, which occur when an entire beam is missed due to the sparse ray sampling, are reduced. However, larger step sizes decrease the dose deposition kernel accuracy. Artifacts are introduced when neighboring voxels transverse different coarse resolution voxels (see Fig. 2). Our implementation is inherently isotropic,



Standard Method

Novel Multi-Resolution Method

Fig. 2 – Diagrams of the memory access patterns of the standard (top left) and multi-resolution (bottom left) methods. Small field (5 mm) dose deposition slices for the standard (top right) and multi-resolution (bottom right) methods, with identical windows and levels. Note the standard's star pattern and multi-resolution's gridding artifacts.

which decreases the benefit of non-uniform angle sampling.

We implemented the multi-resolution structure using a volumetric mip-map [20], as it is both efficient to calculate and has good cache performance. Resolution changes were limited to a maximum of once per step and were only allowed to occur at voxel boundaries in the coarser resolution. This prevented a TERMA voxel from contributing multiple times to the same dose voxel. Kernel tilting was not incorporated as adding tilting would increase the required registers beyond a performance threshold, dropping GPU occupancy from 25% to 17%.

2.4. *Optimizing CUDA performance*

Several strategies were used to optimize CUDA performance. CUDA's execution model is a 2D grid of 3D blocks of threads which execute a function (called a kernel). Each block represents one parallel work unit and therefore is limited in size. Block thread counts were optimized using NVIDIA's CUDA occupancy calculator. For volume processing we used a 1:1 mapping of threads to voxels in the x and y direction. The z direction was handled by looping over the per voxel function with increasing z indexes. The stride was the z block size, which maintained thread spatial cohesion. Increased thread spatial cohesion, enhanced by cube-like block dimensions, reduced cache misses and increased performance. All input array data was cached, generally in textures, which in the case of the superposition provided a ∼2x improvement in performance. Shared memory was used to cache the array multi-resolution volume structs. Shared memory also offered a performance improvement over registers for the multi-spectral TERMA calculation. A maximum number of 21 energy bins was chosen as being both sufficient for high energy beams and free of bank conflicts.

3. Results

Quantitative analysis of a transport algorithm, such as superposition/convolution, is complicated by a strong dependence on the incidence fluence from the source model [21,22]. The source model in turn is optimized to minimize the error between measured dose and calculated dose. Thus, the source model often masks errors in the transport algorithm. We have used a simple source model for these preliminary experiments. Furthermore, commercial systems also included a separate electron contamination model. Despite these limitations, we achieved similar results using the modeling parameters of a commercial treatment planning system, while providing an order of magnitude speed improvement (see Table 1).

Table 1 – Comparison of dose engine performance on cube water phantoms.

Engine	Kernel type	Rays	64^3			128^3	
			Time (s)	VPS	Speedup	Time (s)	Speedup
GPU	CCK, tilting	72	0.198	5.051	41.8x	2.801	33.7x
GPU	CCK, Non-tilting	80	0.159	6.289	52.0x	2.254	41.9x
GPU	CCK, tilting	32	0.086	11.628	96.1x	1.246	75.8x
GPU	CCK, multi-resolution	80	0.097	10.309	85.2x	0.963	98.1x
GPU	CCK, multi-resolution	32	0.042	23.810	N/A	0.411	N/A
Pinnacle^3	CK, non-tilting	80	8.268	0.121	1.0x	94.508	1.0x

Rates reported in volumes per second (VPS). The tilting 32-ray kernel is more accurate than the non-tilting 80-ray kernel (see Table 3). Adaptive multi-grid methods (not implemented) have a speedup factor of ~2x on clinical data.

Table 2 – Performance of the standard forward, novel inverse and approximate TERMA calculation methods.

Size	Forward method	Inverse method	Approximate attenuation	Intensity modulation
32^3	23 ms	2 ms	1 ms	1 ms
64^3	44 ms	18 ms	2 ms	1 ms
128^3	150 ms	220 ms	26 ms	3 ms
256^3	1869 ms	3317 ms	454 ms	13 ms

Also included is the performance of only updating the incidence fluence field which is useful during intensity modulation.

Clinical references were generated using the Pinnacle^3 (Philips–Madison, WI) treatment planning system commissioned on clinical data from a Varian 6EX linear accelerator. This provided a dose deposition, TERMA, transmission array, spectra, mass attenuation tables and mono-energetic dose deposition kernels. All experiments were run on an AMD Opteron 254 (2 cores, 2.8 GHz). Timing experiments were repeated at least ten times with no other programs active, using the standard superposition/convolution engine.

Timing results for our GPU method were repeated multiple times using the high performance hardware counter of the CPU. Experiments were performed on a 3 GHz Pentium 4 with a single NVIDIA GeForce GTX 280. The 80-ray kernels used 10 zenith by 8 azimuth directions angles. The 32-ray kernels used 4 zenith by 8 azimuth directions. The tilting 72-ray kernel used 6 zenith by 12 azimuth directions, which provided higher accuracy (see Table 3). Total dose engine performance was determined from the sum of the source model, TERMA and superposition performance, excluding reusable calculations such as the flattening filter texture and the TERMA attenuation volume.

All tests were performed on a cube water phantom with a side length of 25.6 cm. Tests were performed on a volume with 64^3 voxels, which is representative of standard clinical workload and an additional high resolution volume with 128^3 voxels. The multi-resolution superposition method, which utilizes volumetric mip-maps, performed 2–3 times faster than traditional superposition and performance scaled better to higher resolutions.

Performance of the novel attenuation volume based TERMA algorithm relative to the traditional ray cast algorithm was compared across a variety of resolutions (see Table 2). Both methods incorporate physically correct multi-spectral attenuation and have been optimized for the GPU. The fast, approximate radiological depth based attenuation algorithm, which makes the standard homogeneous material approximation, provided a speedup of ~8x. The performance of only updating the incident fluence, as this is a common occurrence in treatment planning, is reported. The attenuation method exhibits an empirical performance of $O(n^{3.76})$ which is a slight improvement over its theoretical $O(n^4)$. The ray cast scalability, however, exhibits sweet spots and required parameter tweaking to maintain the prerequisite number of rays transversing each voxel.

The spectrum of a beam is defined along its central axis. Therefore, the central axis TERMA is independent from flattening filter effects. This independence was used to calculate the mono-energetic central axis TERMAs analytically, which were then combined using the beam spectrum to calculate the poly-energetic central axis TERMA. Fig. 3 compares the central axis TERMA values for Pinnacle^3, the inverse method and the analytical formulation. The fixed ray step size used in Pinnacle^3 results in a jagged profile and error in the first voxel.

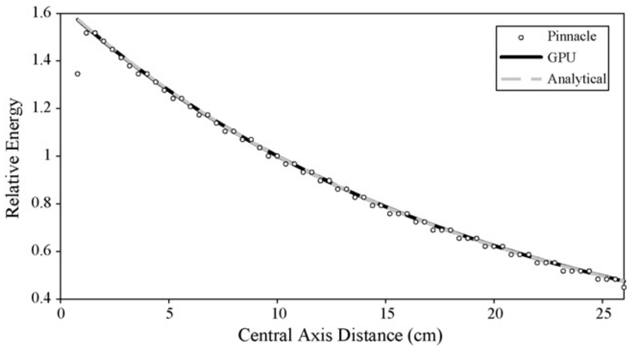


Fig. 3 – Central axis TERMA energies calculated analytically, using Pinnacle^3 and the inverse method, each normalized at a depth of 10 cm. The fixed step length of the Pinnacle^3 ray cast algorithm results in noticeable discretization artifacts, particularly near the surface. The distance squared effect has not been applied.

Table 3 – The average mean deposited dose error relative to $D_{\max}$, the maximum deposited dose, for field sizes from 1 cm to 23 cm.

# of Rays	High dose region			Penumbra region			Low dose region		
	80	72	32	80	72	32	80	72	32
Tilted	0.15%	0.13%	0.34%	0.22%	0.13%	0.30%	0.21%	0.14%	0.25%
Not-tilted	0.37%	0.38%	0.53%	0.79%	0.73%	0.73%	0.43%	0.39%	0.44%
Multi-resolution	0.64%	0.55%	0.76%	0.76%	0.78%	0.81%	0.37%	0.39%	0.38%
Pencil beam	5.92%	5.98%	5.83%	2.77%	2.81%	2.69%	1.68%	1.69%	1.68%

Error is broken down by region with the penumbra region defined as having a dose gradient greater than 0.3 $D_{\max}$ and the low dose region being below 0.2 $D_{\max}$. Fields were square and size was defined at a depth of 10 cm Reference dose deposition was calculated using a tilted kernel sampled with 4608 rays. Pencil beam accuracy was approximated by truncating the superposition kernel at a 3 cm radius [8]. An absolute dosimetry error of 2–5% is clinically acceptable [8].

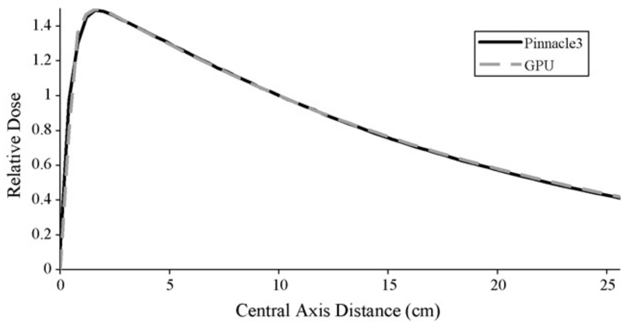


Fig. 4 – Central axis dose depositions for Pinnacle^3 and the GPU implementation normalized at a depth of 10 cm.

The exact radiological path ray casting used in the inverse method avoids discretization, and is therefore very similar to the analytical formulation. All three methods show good agreement, as is to be expected.

Figs. 4 and 5 compare the central axis and 10 cm dose profiles of the GPU implementation to Pinnacle^3 using as similar settings as possible. The implementation shows good agreement with some notable discrepancies. The minor central axis difference is indicative of a slightly harder beam spectrum. This effect may be due to slight differences in spectrum interpolation, or in dose deposition kernel zenith angle interpolation. This can be corrected with a minor change to the modeled beam spectra. The differences in the penumbra of the 10 cm cross sectional dose profile are primarily due to our simplified source modeling.

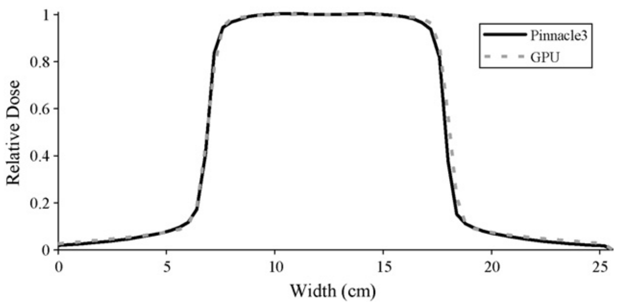


Fig. 5 – Dose deposition profiles for Pinnacle^3 and the GPU implementation at a depth of 10 cm, normalized at the midpoint.

Table 3 compares the accuracy of the multi-resolution method to the standard method across multiple field sizes and kernel ray samplings. The multi-resolution method generally performed better in the penumbra and low dose regions for small field sizes, as less TERMA was geometrically missed by rays. This was at the expense of accuracy in the high dose region as the larger steps caused the beam boundary to be blurred. A variant of the multi-resolution method, using the same step sizes, but not using a multi-resolution data structure exhibited reduced cache performance and increased the mean error by 60% on average.

4. Conclusions

We have implemented a GPU accelerated superposition/convolution based dose engine with real-time performance. The standard TERMA calculation was enhanced with physically correct multi-spectral attenuation and a novel inverse calculation, which is inherently parallel and eliminates ray discretization artifacts. Furthermore, the TERMA attenuation caching strategy improves performance for interactive use and intensity modulation optimization. The tilted, inverse cumulative–cumulative kernel superposition was explored in the standard manner and in a novel way using volumetric mip-maps to approximate solid angle ray casting during dose deposition.

5. Future work

A comprehensive source model which will allow for the quantification of the dose engine accuracy is currently being completed. A number of possible performance enhancements remain to be implemented, including using 16-bit floats and adaptive multi-grid dose calculation. Recent 3D texturing support may allow efficient hardening of the dose deposition kernel.

5.1. *Intensity modulation using superposition/convolution*

We are currently adapting the superposition/convolution algorithm for use during intensity modulation. Superposition/convolution has not been previously used to calculate the optimization derivatives used in intensity modulation. This was due to performance concerns, which are mitigated by our GPU acceleration. Intensity modulation optimizes an objective function (*O*), specified in terms of dose (*D*), with respect to the machine parameters (*P*) that determine of the incidence fluence Ψ_0.

$$\hat{P} = \min_{P} O(D(P)). \tag{15}$$

Several parameter optimization techniques, such as gradient decent, require error, i.e. *dO*/*dD*, to be propagated from the dose grid to parameter space, i.e.

$$\frac{dO}{dP} = \left(\frac{d\Psi_0}{dP}\frac{dT}{d\Psi_0}\frac{dD}{dT}\right)\frac{dO}{dD}. \tag{16}$$

Eq. (16) is determined by the dose transport algorithm and may be calculated using a novel, reverse superposition/convolution. First, *dD*/*dT* is determined by reversing the forward superposition method; instead of spreading dose, the weighted effects of a unit of energy release are gathered. This is identical to using the inverse superposition method with a forward kernel. $dT/d\Psi_0$ is then determined by a ray cast algorithm performing a similar gather weighted by each voxel's attenuation. Continuous methods general postpone the conversion of Ψ_0 to machine parameters to a separate leaf sequencing step. Direct machine parameter optimization and direct aperture optimization benefit from an alternative strategy utilizing a differential fluence with a limited field of influence, similar to current superposition/pencil beam hybrid solutions.

Conflicts of interest statement

None declared.

Acknowledgments

This work was funded in part by the National Science Foundation under grant EEC9731748, and in part by Johns Hopkins University internal funds.

REFERENCES

[1] H. Sutter, The free lunch is over: a fundamental turn toward concurrency in Software, Dr. Dobb's J. 30 (March) (2005).

[2] T.R. Mackie, J.W. Scrimger, J.J. Battista, A convolution method of calculating dose for 15-MV X-rays, Med. Phys. 12 (1985) 188–196.

[3] T.R. Mackie, A. Ahnesjo, P. Dickof, A. Snider, Development of a convolution/superposition method for photon beams, Use Comp. In Rad. Ther. (1987) 107–110.

[4] T.R. Mackie, P.J. Reckwerdt, T.R. McNutt, M. Gehring, C. Sanders, Photon dose computations, in teletherapy, in: J. Palta, T.R. Mackie (Eds.), Proceedings of the 1996 AAPM Summer School, AAPM-College Park, MD, 1996.

[5] K. Otto, Volumetric modulated arc therapy: IMRT in a single gantry arc, Med. Phys. 35 (2008) 310–317.

[6] C.X. Yu, Intensity-modulated arc therapy with dynamic multileaf collimation: an alternative to tomotherapy, Phys. Med. Biol. 40 (1995) 1435–1449.

[7] D. Yan, F. Vicini, J. Wong, A. Martinez, Adaptive radiation therapy, Phys. Med. Biol. 42 (1997) 123–132.

[8] A. Ahnesjo, M. Aspradakis, Dose calculations for external photon beams in radiotherapy, Phys. Med. Biol. 44 (1999) R99–R155.

[9] N. Papanikolaou, T.R. Mackie, C. Meger-Wells, M. Gehring, P. Reckwerdt, Investigation of the convolution method for polyenergetic spectra, Med. Phys. 20 (1993) 1327–1336.

[10] A. Ahnesjo, P. Andreo, A. Brahme, Calculation and application of point spread functions for treatment planning with high energy photon beams, Acta Oncol. 26 (1987) 49–56.

[11] T.R. Mackie, A.F. Bielajew, D.W.O. Rogers, J.J. Battista, Generation of photon energy deposition kernels using the EGS Monte Carlo code, Phys. Med. Biol. 33 (1988) 1–20.

[12] J. Krüger, R. Westermann, Acceleration techniques for GPU-based volume rendering, IEEE Visualiz. (2003).

[13] Nucletron Reports on the North-East Italian Oncentra® MasterPlan User Meeting, Venice, October 2007, www.nucletron.com/content/ContentPage.aspx?app=news&cpID=31&miID=56&id=1250.

[14] NVIDIA CUDA Zone, www.nvidia.com/object/cuda_home.html.

[15] D Programming Language, www.digitalmars.com/d.

[16] J. Amanatide, A. Woo, A fast voxel traversal algorithm for ray tracing, in Eurographics'87, Conference Proceedings (1987).

[17] H.H. Liu, T.R. Mackie, E.C. McCullough, Correcting kernel tilting and hardening in convolution/superposition dose calculations for clinical divergent and polychromatic photon beams, Med. Phys. 24 (1997) 1729–1741.

[18] A. Ahnesjo, Collapsed cone convolution of radiant energy for photon dose calculation in heterogeneous media, Med. Phys. 16 (1989) 577–592.

[19] W. Lu, G.H. Olivera, M. Chen, P.J. Reckwerdt, T.R. Mackie, Accurate convolution/superposition for multi-resolution dose calculation using cumulative tabulated kernels, Phys. Med. Biol. 50 (2005) 655–680.

[20] L. Williams, Pyramidal parametrics, SIGGRAPH Comput. Graph. 17 (3) (1983) 1–11.

[21] R. Mohan, C. Chui, L. Lidofsky, Energy and angular distributions of photons from medical linear accelerators, Med. Phys. 12 (1985) 592–597.

[22] T.R. Mcnutt, dose calculations: collapsed cone convolution superposition and delta pixel beam, Philips White Paper.