Improvement of dose calculation program for patient treatment planning in the Neutron Therapy Facility

Felipe Giraldo

August 5, 2014

Abstract

In order to properly determine the dose given to a patient in the Neutron Therapy Facility at Fermi National Laboratory, a computer program that simulates the dose is used. This program currently uses a fitting of the dose that has areas of poor fit to the data. A new algorithm that more accurately models the dose has been achieved. This new algorithm is able to determine the dose more precisely in all areas.

1 Introduction

The use of particle beams to treat cancer was proposed by Dr. Robert Rathburn Wilson, first director of Fermilab, in 1946[1]. Particle accelerators, like Fermilab's Linac, are used to create the beam of high energy particles, which is then directed towards cancerous tissue. The beam of particles is used to kill the cancerous cells by ionizing molecules and creating irreparable damage. Protons, photons and electrons, which are popularly used in radiation therapy, undergo Coulombic interactions with the electrons of molecules and atoms, ionizing them by breaking chemical bonds. Neutrons, which are electrically neutral, interact directly with nuclei and can change the actual atom they affect. Cells have natural mechanisms to repair broken chemical bonds, but if the chemical itself changes the damage is irreparable. Also, neutrons are referred to as indirectly ionizing radiation since the interactions they have with atoms causes the production of other types of radiation, which then cause a large amount of subsequent ionization[2]. In general, neutrons are better at fighting radioresistant tumors than other types of radiation due to the nature of their interaction with atoms[3]. Because of this, neutron therapy is an attractive alternative to the more popular therapies.

In radiation therapy it is necessary to be able to control the position and size of a beam in order to effectively treat the affected area of a patient, while minimizing the damage on healthy tissue. In the case of protons and electrons this can be easily achieved due to the electrical charge of the particles being used. Magnetic fields can be used to precisely direct the beam of particles in order to attack the cancerous cells. This, however, cannot be done with neutrons since they're neutral. Instead, a collimator has to be used in order to control the region being irradiated. The basic idea of collimators is to put a physical barrier to the neutrons so that the beam comes out of a well-defined area. Collimators are made out of a thick material with a hole of a certain size through which the beam of neutrons comes out. Collimators are made out of different materials like steel and cement that are able to block neutrons in order to narrow the beam. Depending on the size and shape of the affected region of the patient's body a specific collimator (with an aperture of a particular size and shape) is used.

In the Neutron Therapy Facility (NTF) the current configuration uses a series of cylindrical cement blocks with rectangular apertures in the center from where the neutron beam comes out. Various rectangular openings of different sizes can be used to change the beam outline as needed for the individual case. In order to further adjust the beam to match the affected area, triangular and rectangular steel blocks can be placed inside the rectangular opening. The patient is then placed at a certain distance (which can be precisely controlled) away from the collimator with the beam centered on the affected area (see Figure 1.1).

The collimator is able to control the amount of radiation on the plane perpendicular to the trajectory of the beam, but it cannot control the amount of radiation in the direction of the beam trajectory. That is, if a person is placed with his front to the beam, then the collimator is able to control the amount of radiation from side to side (towards or away from his left arm), denoted as the x axis, and up and down, denoted as y axis (towards or away from his head), but it is not able to control the amount of radiation he receives as a function of depth (towards his back), which we denote as z axis (see Figure 1.2). In order to control the z axis, the patient is rotated

Figure 1.1: Patient and therapist in the NTF treatment room showing the configuration of the facility. The cylindrical cement piece in the therapist's hand is one of the collimators, showing how it can be changed depending on the patient's need. Also, below the collimator it is possible to see some of the steel blocks that can be placed in the aperture. Picture courtesy of the Neutron Therapy Facility at Fermi National Laboratory

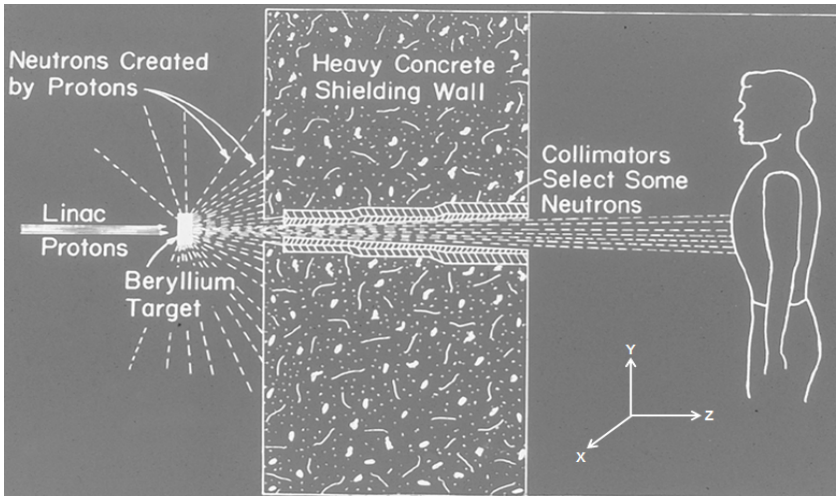

Figure 1.2: Diagram of NTF configuration with axes labeled, where the x axis is the horizontal axis from side to side of the patient's body, the y axis up or down the body, and the z axis is the depth from the chest to the back of the patient. Picture courtesy of the Neutron Therapy Facility at Fermi National Laboratory

at different sessions, with the tumor as the pivot point, in order to maximize the radiation delivered to the tumor while minimizing its effects on the surrounding healthy tissue.

A computer program created at NTF is used to calculate the dose delivered to a patient. This program simulates the dose of all the different collimator configurations. This program was based on previous fits of multiple data sets and was intended to create a universal algorithm that would be able to calculate the dose for all configurations. In the process of expanding this program's capability, it became evident that there were areas where the algorithm is not very representative of the actual data. In order to create a comprehensive treatment plan it is essential for this program to precisely calculate the dose everywhere in order to maximize the effects of the radiation on the cancerous cells while minimizing the detrimental effects on healthy tissue. This is why it seemed necessary to modify the program in order to rectify those particular areas to have a more appropriate fit. The purpose of this work is to create a more appropriate algorithm to approximate the dose in the patient treatment planning at NTF.

2 Notation and Original Algorithm

For clarity, the notation used when talking about a collimator configuration will be explained. Rectangular collimators are denoted by their width and height in centimeters (in that order). So a 6x10 collimator is one that has a rectangular opening that is 6cm wide and 10cm high. The data analyzed in this work had all been previously recorded by earlier investigators by placing an ionization chamber inside a water tank. The water is used to simulate the patient to understand the dose distribution inside of a person. The z value given to characterize the data set is the distance (again in centimeters) from the surface of the water (away from the collimator) at which the data set was taken. So when talking about a collimator data set (whether the actual data or the fitting), for example, of a collimator that is 6cm wide, 10 high and the data set was taken 10cm away from the surface of the water, then the notation will be '6x10 at z=10'.

The data set recorded from a 10x10 at z=10 was initially used. The dose at several points over the z=10 plane had been previously recorded and normalized over a 32x32cm area, going from -16 to 16 cm in both the x and y axis, taking the center of the collimator as the origin. In reality the dosimetry data set was taken over $\frac{1}{8}$ of the area previously discussed and then, using the symmetry of the collimator, it was extrapolated over the entire square. A 3D plot of the dose is shown in Figure 2.1, where the x and y axis are the distances from the center of the collimator and the z is the normalized dose. This data set will be denoted as SD (Square Data). Using a piecewise fit generate by Matlab's CFTOOL[4] a graph of the dose was created, shown in Figure 2.2 and which will be denoted as MFSD (Matlab Fit of Square Data). The CFTOOL in Matlab is an interactive application which generates fittings of 2D and 3D data with a number of predetermined equations or with customized equations which the user can input. The data set from the Original Algorithm (denoted as OA) was generated, this time over a 40x40 cm area (from -20 to 20 cm), and plotted (using the piecewise fit), as shown in Figure 2.3. This fit will be referred to as MFOA (Matlab Fit of Original Algorithm). The main problem with the OA becomes evident with these figures: the corners of the fitting have a much lower value than they should. In order to understand why this phenomenon occurs, the OA fitting mechanism will be explained below. The objective of the present work, then, was to create a new fit that will correct the values in the corners, while still accurately describing the other areas of the graph.

The basic idea of the OA is to create a normalized x and y fit separately, then multiply them, and finally apply an additional algorithm that adds the z dependence and converts the data to dose by multiplying by the appropriate amount. The program creates the x and y fitting, both of which are 2D fits (the horizontal axis being the position using the center of the collimator as the origin and the vertical axis the dose), by using two parameters: the size of the collimator opening and the z axis distance from the source. It has to be noted that the x and y axes in the program are taken to be the two orthogonal axes of the rectangular aperture; these may not always line up with the vertical and horizontal room axes since the collimator can be rotated by an angle with respect to the room reference frame. The program then receives the x and y position of the desired point and calculates the dose in the x and y axis separately, and then multiplies the two values. This is how it determines the dose in the plane perpendicular to the beam trajectory. Finally it applies another algorithm to determine how the dose varies as a function of the z axis, creating the final fit of the dose for the collimator configuration. This final fit can be thought of as a 4D graph, with the 3 spatial dimensions as the independent variables and the dose as the dependent variable.

The z dependence of the dose is accurately described by the OA, the major problem being the perpendicular plane (x and y) dependence. The fit in this plane follows the shape of the actual data closely except for four squares in the corners. The problem there is that the value gets a *'double hit'*. The idea is that in both the x and y 2D fits the dose value is really small. So by multiplying each other the total is made even smaller. That is why instead of having a gradual decay, it becomes small too rapidly. But, apart from the 'double hit' areas, the OA accurately

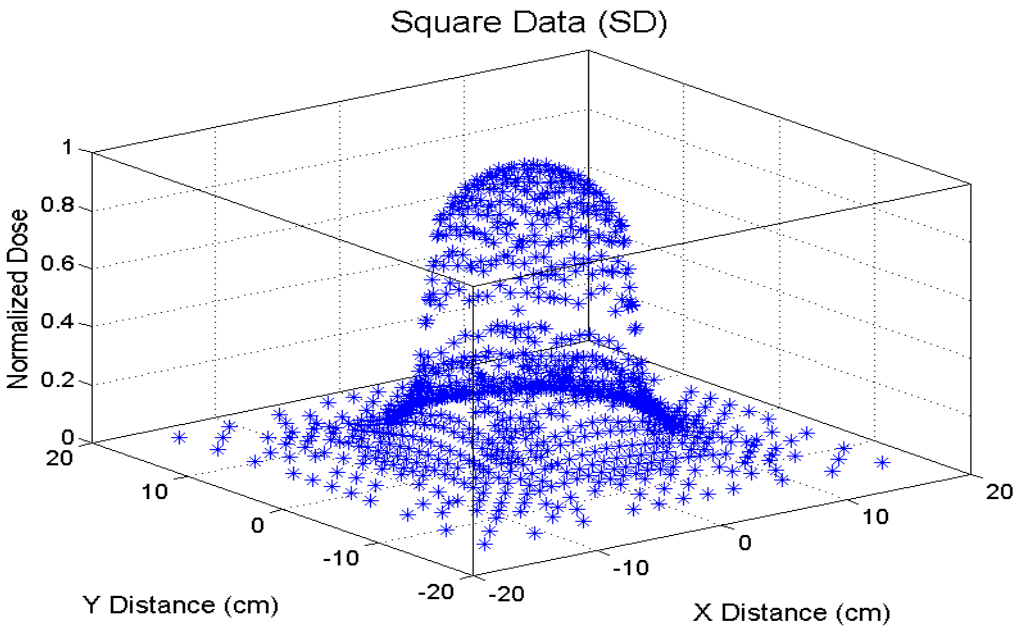

Figure 2.1: 10x10 at z=10 dose data taken over a 32x32cm area used to analyze current and proposed fit (Square Data -SD)

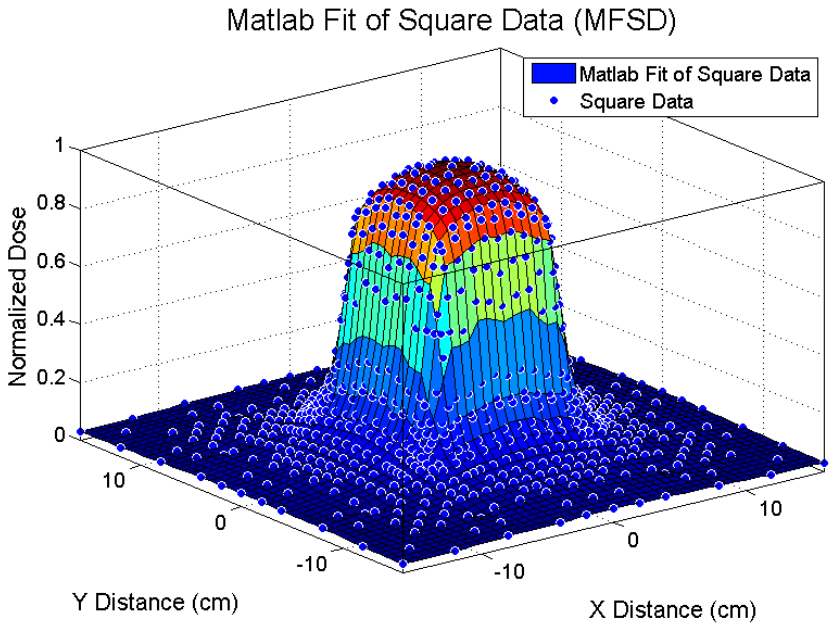

Figure 2.2: Piecewise fit of Square Data (SD) generated by Matlab (MFSD)

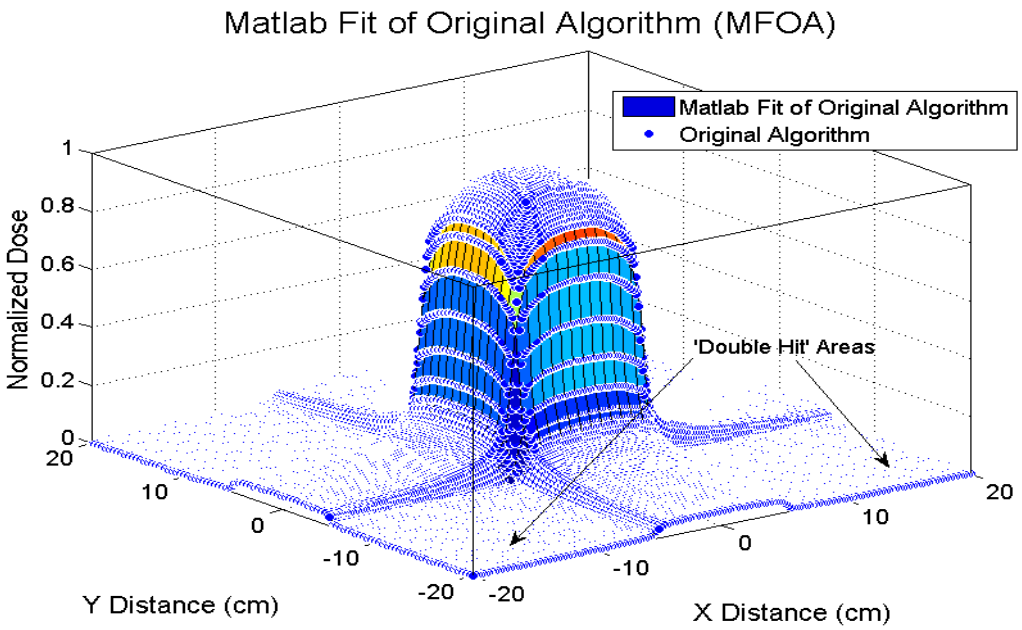

Figure 2.3: Piecewise fit of Original Algorithm generated by Matlab (MFOA), with indicated 'double hit' areas

describes the dose. It is also able to account for different shapes of collimators (like when steel block are placed in the aperture) since the fit in x and y takes into account the width of the collimator at the desired point only, so the calculation is not affected if the width is not a constant over all of the collimator.

3 Methods

Several fitting techniques were contemplated when trying to find a new algorithm for the dose. The techniques were: creating a completely new algorithm (Section 3.1: Top-Hat Beam), modifying the process used in the OA (Section 3.2: Picking Minimum of X and Y) and adding a separate process to the OA that fits the fallout of the data (Section 3.3: Gaussian Fallout and Section 4: Solution: Cauchy Fallout).

3.1 Top-Hat Beam

The first approach taken in order to construct a better algorithm was to analyze the general shape of the SD. The dose has somewhat of a top-hat beam shape[5], since it has a central bulge, a fairly sharp transition and then a fallout decay for large distances from the center. An idealized top-hat as a mathematical construction has a constant maximum value over a finite area and then a zero value everywhere else. A more physical top-hat will transition with a continuous (and differentiable) function between the two values.

In order to analyze the 3D top-hat shape it was easier to analyze it in 2D first and then generalize it to 3D. To study the 2D profile of the dose, a graph of the Central Transverse Profile (CTP) of the SD, that is, the dose as a function of the x axis alone, with the y axis constant at y=0 (the opposite, having x=0 and the dose as a function of y, yields exactly the same results due to symmetry), was plotted, which can be seen in Figure 3.1. The figure suggests a physical top-hat function. A report by Anthony A. Trovar, Ph. D. describes an approach to approximate a top-hat as a sum of an odd number of Gaussians symmetrically placed around the maximum[6]. This seemed ideal due to the fact that low order approximations, up to 3 or 5 Gaussians, have round edges where the maximum starts to decay towards the minimum, just like the dose. More than 5 Gaussians cause the edge to become much sharper, resembling every time more a step function and making it a worse approximation of the data.

This approach of approximating a top-hat also has the advantage that it can be generalized to 3D, by adding 3D Gaussians, which was precisely the next step. Nine and then twenty-five 3D Gaussians were initially used to approximate the SD using the CFTOOL in Matlab (see Figure 3.2). However it was clear that this fit is not much better at approximating the corners and is also significantly worse at representing the central bulge than the OA.

In an attempt to better approximate the fallout, a 3D parabola was added to the sets of Gaussians but it did not significantly improve the fit.

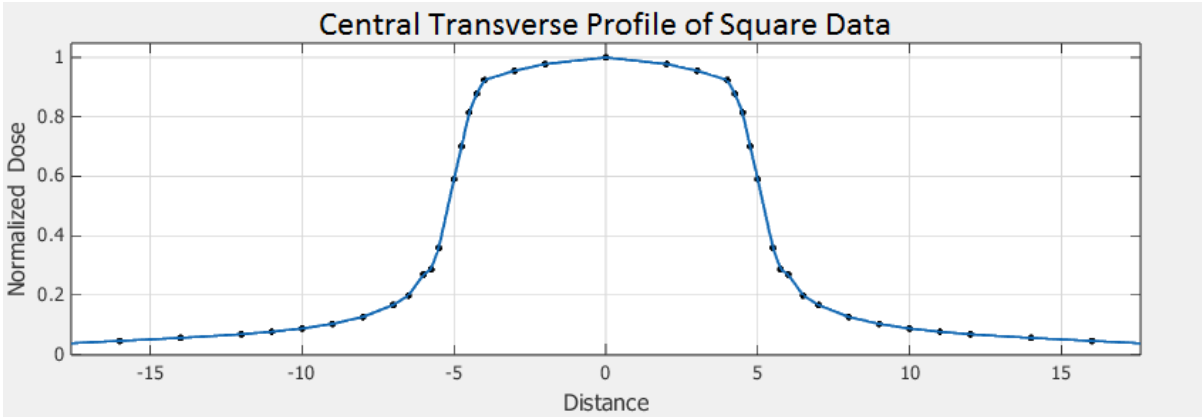

Figure 3.1: Central transverse profile (CTP) of Square Data (SD)

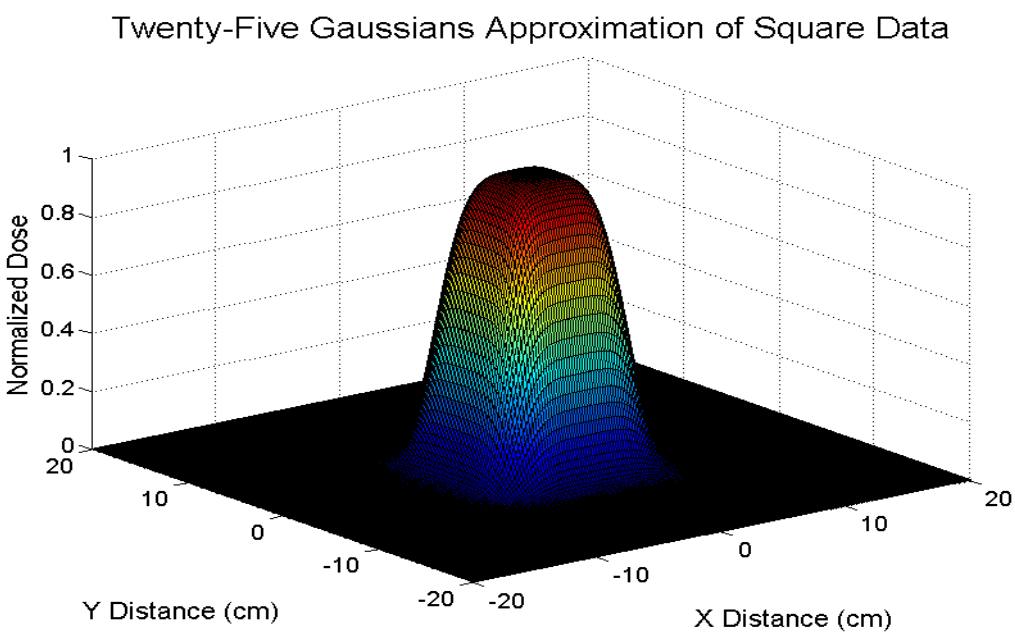

Figure 3.2: Twenty-five, 3D Gaussians added to approximate the Square Data (SD)

3.2 Picking Minimum of X and Y

Another approach that was contemplated was, instead of multiplying the two values of the x and y fits, to choose the minimum of the two values at the desired point. Since the two values do not get multiplied the zones with the 'double hit' would disappear. This approach maintains the versatility of the OA, while eradicating the zones of major discrepancy. It is also straightforward to change the computer program. However, it turns out to be undesirable because it modifies the central bulge and the overall form of the fit by making it sharp, as can be seen in Figure 3.3. Since it chooses between the two values it does not have smooth transitions where the x and y values change relative magnitude, having discontinuities every time it happens. It also makes the values in the central bulge bigger than the OA. It is necessary for a new algorithm to maintain the level of accuracy of the OA in the central bulge since this is where the cancer is located, so it is important to precisely know the dose in this area.

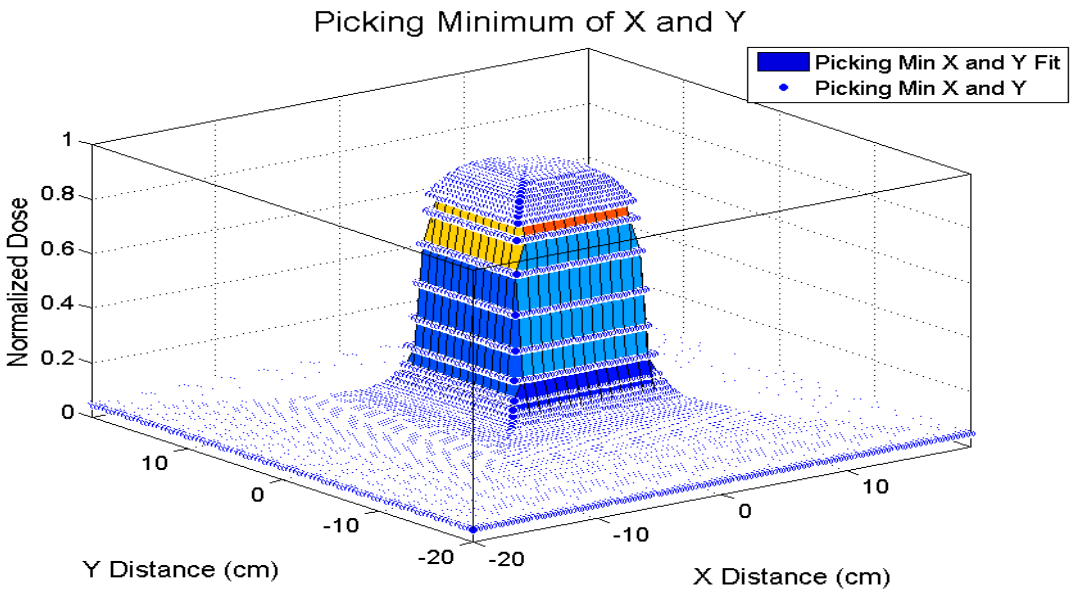

Figure 3.3: Picking minimum of x and y fit of Square Data (SD)

3.3 Gaussian Fallout

The other approach at finding an appropriate fit was to be less radical. It is true that the OA is good at approximating the bulge and other parts of the actual data, so there was no necessity of starting anew. Instead, all that was necessary was to fix the undesired 'double hit' areas, while keeping the positive aspects of the OA. With this in mind, the idea that arose was to create a fit for the fallout alone, while keeping all the other parts of the OA. This idea not only serves as a procedural technique, but can also be rationally justified. It is not entirely unreasonable to think of the dose distribution as composed of a central bulge directly in front of the opening, and a general background that decays as you move farther away from the beam.

Using the CFTOOL in Matlab, a Gaussian was first proposed to be a good candidate for approximating the fallout. Several different variations of Gaussians were used, trying different widths, adding constants so and so on. But, despite all those different attempts, Gaussians are clearly not proper fits. The major reason being the decay characteristic of Gaussians since they reach a minimum value too quickly (see Figure 3.4). To try and fix this problem a 3D parabola was combined with the Gaussians, but this did not improve the fit significantly enough and added the new problem that for large x and y the value becomes negative, implying negative doses.

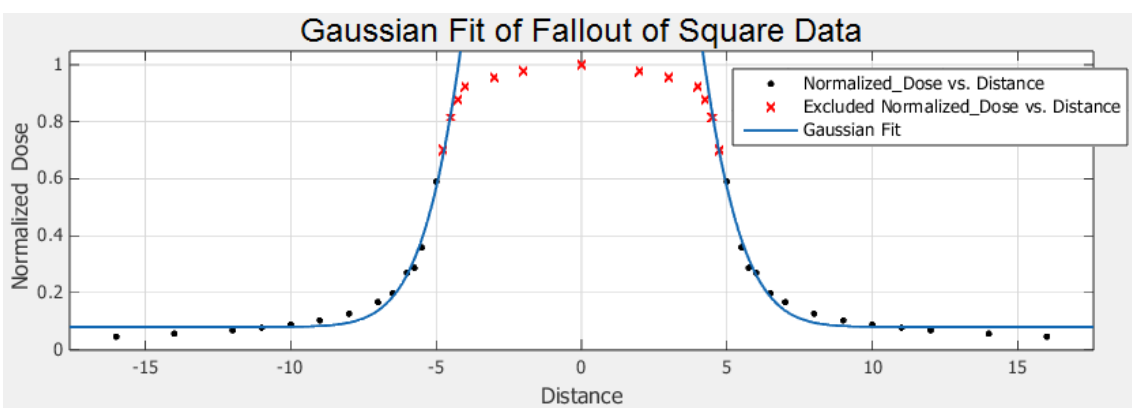

Figure 3.4: Gaussian fit for fallout of Square Data (SD). The central transverse profile (CTP) is plotted, with the solid line showing the Gaussian fit to the fallout, the black dots representing the data of the fallout and the red marks all the other points not considered for the fit

4 Solution: Cauchy Fallout

The goal seemed to be to find a function that would decay similarly to a Gaussian but more gradually. A Cauchy distribution was found to be a good candidate due to its gradual decay and adaptability to 3D. Even with the first fit attempt it was obvious that the Cauchy distribution is much better than the Gaussian. After adjusting different parameters, a 3D Cauchy distribution was clearly the most viable option. The Cauchy distribution is able to accurately describe the fallout (see Figure 4.1), so the next problem was to merge it with the OA.

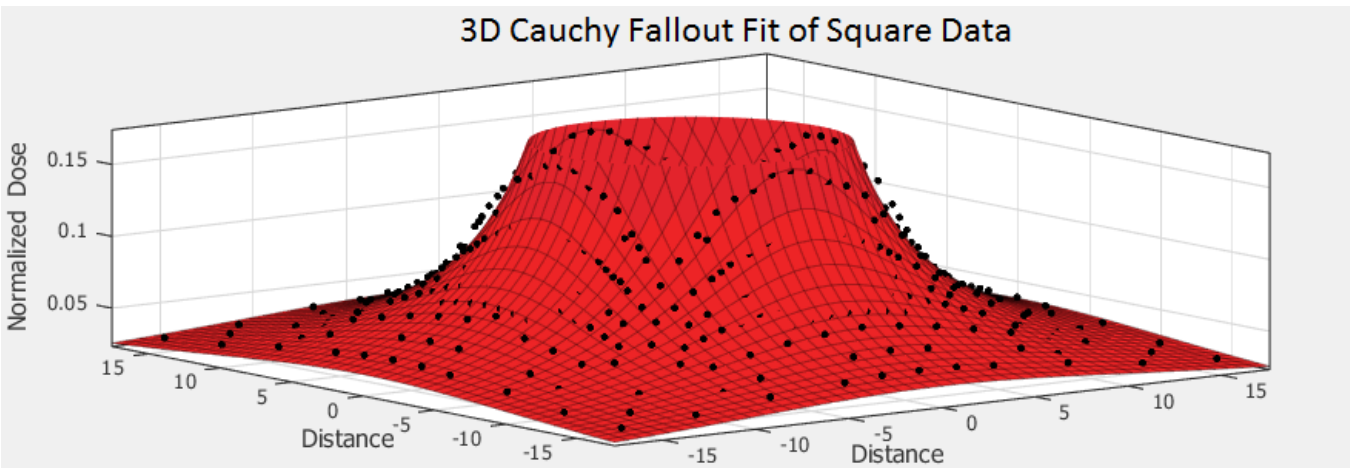

Figure 4.1: 3D Cauchy fallout fit for Square Data (SD), where the black dots are the data and the red surface the 3D Cauchy

The first attempt was to switch between the OA and the Cauchy at a given radius away from the origin, so the Cauchy would start after a circle centered at the origin. However, this causes big disparities in the values and it is difficult to pick a radius that keeps the central bulge while getting rid of all the 'double hit' areas at the same time. Another approach was to start it at a certain radius and also to transition between the OA and the Cauchy with the angle, so that at the corners the fit would be dictated by the Cauchy and at the axis it would keep the values computed by the OA. This does not significantly improve the fit and also adds the problem that if a block is placed in the opening, then the fit will still have areas of 'double hit' since they do not necessarily happen in the corner. Finally the solution was to change from the OA to the Cauchy after the data are 15% away from the edge of the collimator. In order to minimize the discrepancies between the OA and the Cauchy when switching between them, the transition was made continuous. This was achieved by adding an exponential dependence to the transition between the two: the data start as completely coming from the OA calculations and exponentially transition to the Cauchy so that within a small distance from the 15% mark from the collimator edge the Cauchy completely dictates the fit.

The next step was to try and adapt the Cauchy fit to other collimators, and later to other depths. The CTP (Central Transverse Profile) data for different collimators were used to find a scalable adaptation to the Cauchy. There were data sets for 6x6, 14x14, 10x10 and 20x20 collimators at z=10. The idea was to be able to scale the Cauchy with a simple formula which could be easily added to the algorithm. After different attempts a formula was found to scale the Cauchy to properly adjust for the different size collimators.

The next step was to find a way to adapt the Cauchy for different depths. For this purpose CTP data sets of a 10x10 collimator were used; there were data sets at z=2, 5, 10, 15, 20 and 30. In order to adjust for the depth dependence two terms were added to the Cauchy in order to modify it. One was a constant term to multiply the Cauchy, which makes it possible to increase or decrease the width of the Cauchy (how quickly it decays). The other one was adding a constant, which would change the value the Cauchy decays to (so it shifts it up or down). These two parameters were adjusted for each of the z values and plotted. Using Matlab's CFTOOL two equations were found for these two terms. After applying them to the fit, it was necessary to check that all collimators were accurately described by this depth dependence. In order to verify this the data from 4x4, 6x6, 14x14, 20x20 and 24x24 collimators, at the same depths as previously discussed for the 10x10, were used. Indeed it was found that the fit follows the general trend of the data and is accurate at describing both the depth and collimator size dependence of the data. The final Revised Algorithm (RA) fit for the 10x10 at z=10 can be seen in Figure 4.2.

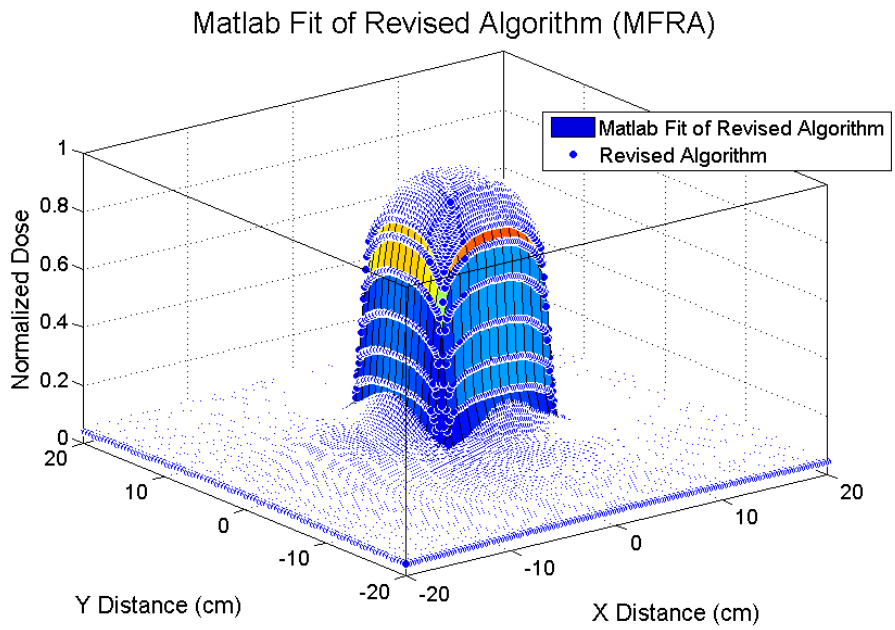

Figure 4.2: Piecewise fit of Revised Algorithm (RA) for Square Data (SD) generated by Matlab (MFRA)

The idea of the Equivalent Square (ESQEF) has to be explained in order to understand the implemented Cauchy in the new fitting algorithm. The ESQEF in the computer program is the size of a side of a square that has the same area as the collimator opening. In the special cases where the collimator aperture is a square then the ESQEF is nothing but the length of the side. However, for other shapes (such as rectangles or irregular shapes created by placing the steel blocks), the ESQEF is not as straightforward and the program has to calculate the area of this new collimator configuration in order to yield the ESQEF. With this in mind, the Cauchy in the proposed algorithm has the following form:

$$Cauchy = \frac{6.658 * \frac{10}{ESQEF}}{1 + (\frac{10}{ESQEF} * 0.8426 * (X^2 + Y^2))} + 0.01279 \tag{4.1}$$

In this equation $Cauchy$ is the basic Cauchy distribution function used to describe the fallout of the x-y plane of the data, the X and Y are the distances from the center of the collimator and the $ESQEF$ is the Equivalent Square previously described. The ESQEF in the equation serves as the scaling factor for collimators different from 10x10. The rationale of using the ESQEF for scaling purposes is that the scattered radiation (which causes the fallout in the graph) is caused by the area of the collimator aperture, more than the specific shape.

The algorithm to scale the Cauchy for different depths is the following:

$$Dose\ Cauchy = Cauchy * (0.1921 * (Z + 6.29)^{0.9} * e^{-0.06373*(Z+6.29)}) + (9.082 \times 10^{-6} * Z^2 + 2.243 \times 10^{-4} * Z + 0.01051) \tag{4.2}$$

In this equation $Dose\ Cauchy$ denotes the Cauchy algorithm with the z dependence included, $Cauchy$ again the basic Cauchy distribution function for x-y plane of the data, and Z denotes the distance from the surface of the water.

An equation of the following type is used transition between the OA and the Cauchy:

$$Dose = (Dose\ OA) * e^{-10\sqrt{(X-HW_x*1.15)^2+(Y-HW_y*1.15)^2}} + (Dose\ Cauchy) * (1 - e^{-10\sqrt{(X-HW_x*1.15)^2+(Y-HW_y*1.15)^2}}) \tag{4.3}$$

In this equation $Dose\ OA$ denotes the dose value computed from the Original Algorithm (OA), $Dose\ Cauchy$ again the Cauchy algorithm (with z dependence included), X and Y the distances from the center of the collimator,

HW denotes the Half Width (HW), which is the distance from the center to the edge of the collimator and can be different for both the X and Y axis (therefore the subscripts), and *Dose* denotes the final algorithm that dictates the dose values. This equation is only applied once either the x or the y distances are bigger than 1.15 times the HW (in the respective axis).

5 Results

In order to properly understand the results of the new fitting algorithm, which we will denote as RA (Revised Algorithm), an explanation of the data sets used to get the results is necessary. The OA and the RA were both compared to 10x10 at z=10 data. There were three different sets of data for this configuration: two CTPs (denoted as Data 1 and Data 2) as well as the SD previously described. The SD is significant since it is the only 3D data set, which makes it possible to directly compare the 'double hit' areas. Because of this, the SD is necessary to interpret the shape of the fittings. However, in order to determine the exact dose there is a displacement factor that has to be corrected in recorded data[7]. This correction was not included in the SD so the two CTPs prevail when trying to determine the actual values of the data (since they do have the proper correction). Due to this inconsistency when trying to determine the appropriateness of the fits, several different procedures were used in order to interpret the results.

5.1 Qualitative

Different graphs were used to qualitatively determine the correctness of the RA. For the 3D aspects of the graph the following procedure was used. Using Matlab's CFTOOL a piecewise function is generated for the RA, denoted as MFRA (Matlab Fit of Revised Algorithm). Then using this MFRA, the MFOA (Matlab Fit of Original Algorithm) and the MFSD (Matlab Fit of Square Data) it is possible to plot the difference between the MFRA and MFSD, as well as that of the MFOA and the MFSD, and also the absolute value of these differences. The plot of the absolute difference between the MFOA and the MFSD can be seen in Figure 5.1, and the plot of the absolute difference between the MFRA and the MFSD can be seen in Figure 5.2.

From these graphs some of the significant aspects of the RA can be seen. As the graphs show, the RA leaves the central bulge unaffected. As for the fallout, it can be noted that the RA maintains a pretty constant difference with the SD, while the OA has the four corners of 'double hit' where it largely differs from the data and cannot be said to accurately describe it. These graphs offer a visual representation of how the RA does a better job at following the general shape of the data.

Another graph used to understand the accuracy of the RA is that of the Half Central Transverse Profiles (or HCTPs). The 'half' denotes the fact that only half of the CTP is plotted: only the positive distances are taken into account (since the data sets are symmetric this is sufficient). In a single graph the HCTPs of Data 1, Data 2, SD and both the OA and RA are plotted. In this graph the y axis is the normalized dose and the x axis is the half-width distance from the center of the collimator, that is, the distance from the center divided by half of the collimator width (so divided by the HW), in this case 5cm (see Figure 5.3). This graph is fundamental to understand the quantitative results that are going to be explained, as well as other benefits of the RA.

An important benefit of the RA can be seen with these plots. Paying close attention to the OA and the data, it can be seen that between 1 and 1.75 half-widths away from the center the OA has a significantly lower value than any of the data. This part is very important because in practice it would be an area of healthy tissue adjacent to the cancerous tumor. This may be a patient's vital organ, so it is important to precisely know the amount of dose given to this area when planning a treatment. In this part of the graph the RA does a better job at approximating the dose of Data 1 and Data 2.

As for larger distances, the RA is also arguably better than the OA. Since Data 1 and Data 2 are to be given priority in the actual value, a fit that does a better job at approximating these is better. The RA is not far from either Data 1 or Data 2, making it much better since it is able to find a balance between the two. In practice the actual value of the dose fluctuates from recording to recording, Data 1 and Data 2 being examples of this fluctuation. It is therefore much better to have a fit that is between the two different sets of data.

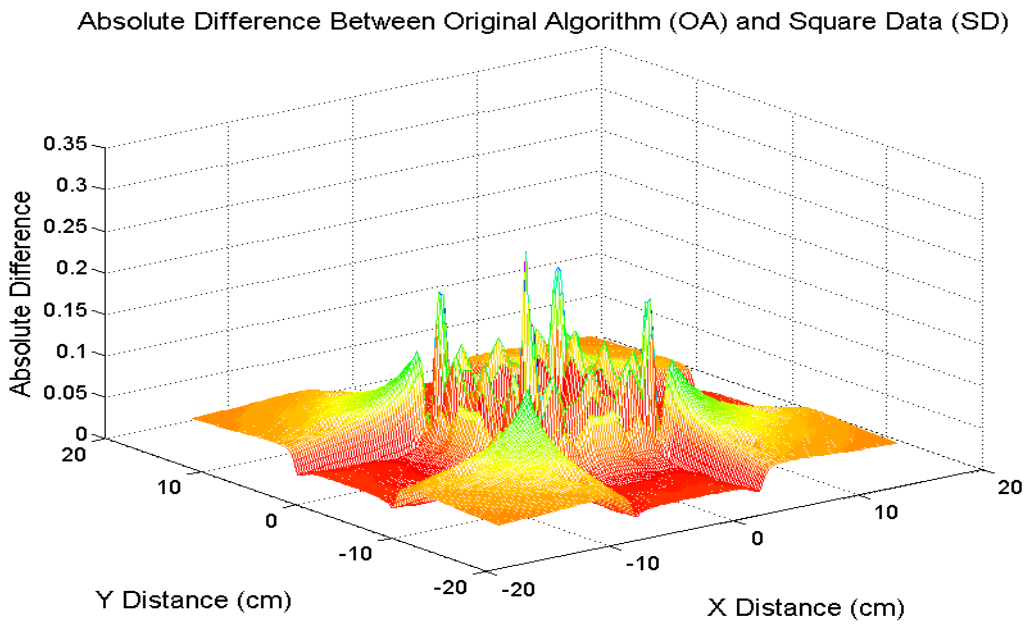

Figure 5.1: Plot of the absolute difference between the Original Algorithm (OA) and Square Data (SD), generated from the piecewise fits (MFOA and MFSD)

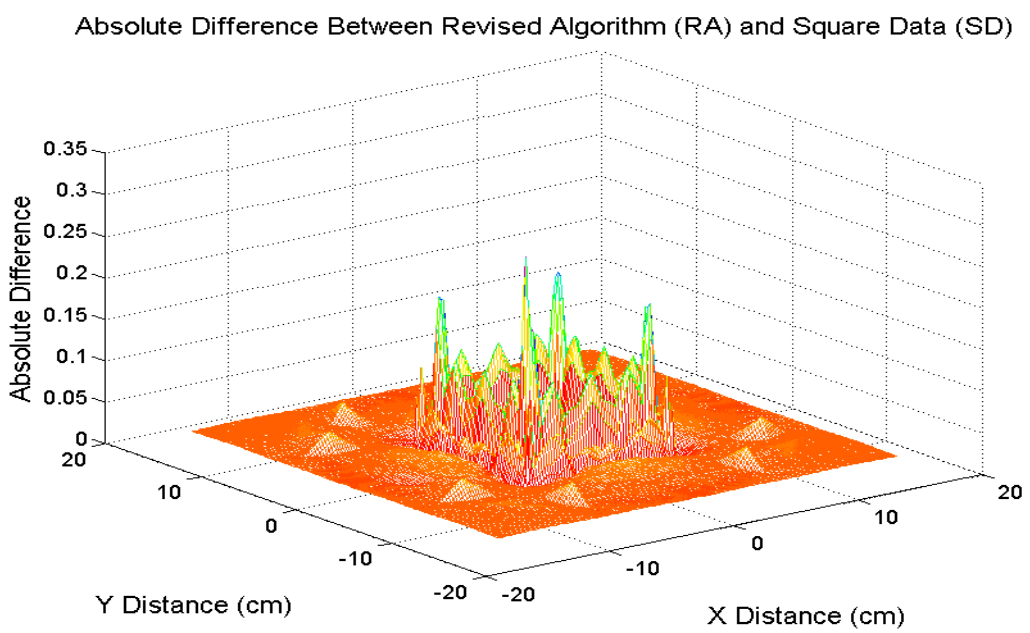

Figure 5.2: Plot of the absolute difference between the Revised Algorithm (RA) and Square Data (SD), generated from the piecewise fits (MFRA and MFSD)

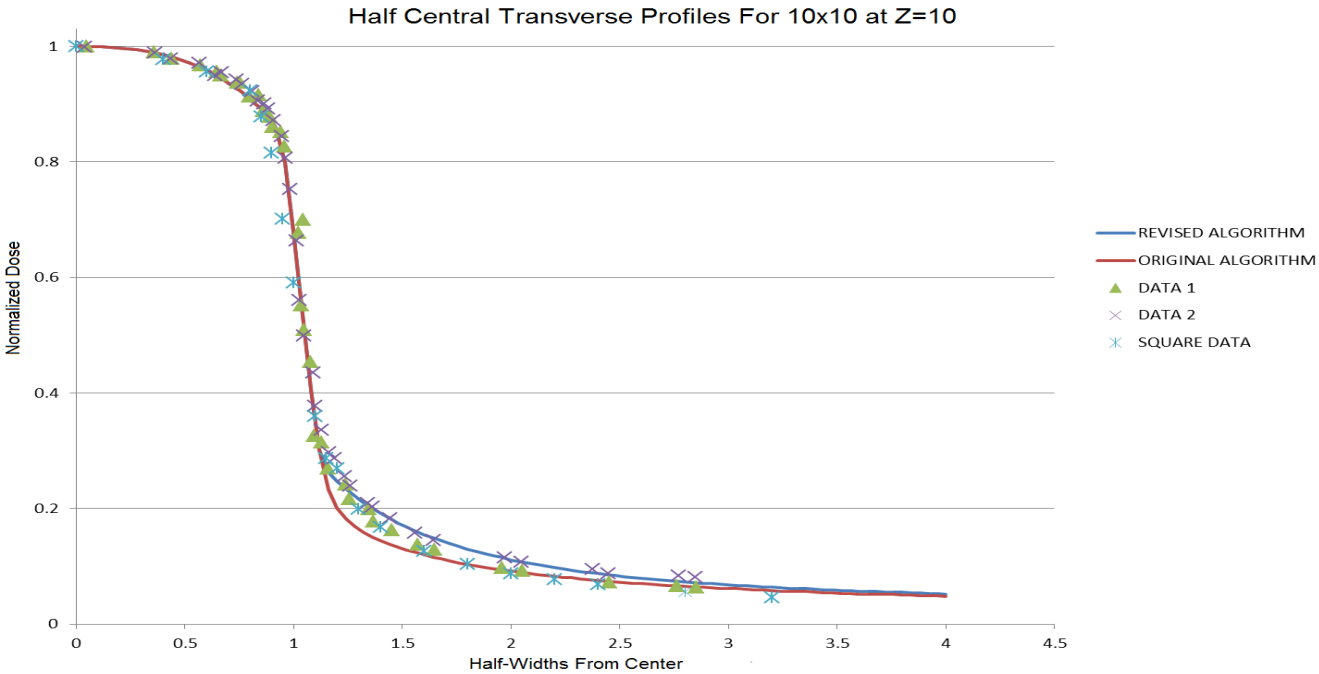

Figure 5.3: Half center transverse profiles (HCTPs) of 10x10 at z=10. The y axis represents the nomalized dose, while the x axis represents the number of half-widths (half the width of the colimator) away from the center

5.2 Quantitative

The SD has a lower value in both the transition and fallout than the other recorded data. This makes the process of determining a quantitative result much harder, since the SD offers information of the shape, but does not offer properly corrected values. Because of this, getting a quantitative result out of the SD alone would not reflect the full extent of the improvements, and neither would a quantitative result using Data 1 or Data 2 since they do not offer any information of the 'double hit' areas or general 3D aspects of the measurement.

To get some sort of quantitative comparison between the OA and RA, Riemann sum approximations to 3D integrals were used. The idea is that from the functions generated by CFTOOL in Matlab it is possible to get approximate values for the data and the fits, as well as their differences. By using a grid of 0.2 cm intervals it is possible to approximate the integral of each of the graphs over a 32x32 cm area. Table 1 shows the values of different integrals. The error was computed by calculating the gradient on each of the axes for the given graph, then adding the magnitude of the x and y gradients (since this gives an estimate of the magnitude of how much the value can vary inside the 0.2x0.2 grid), and then calculating the Riemann sum of this absolute value (this can be thought of as the next term in a Taylor expansion, which is a good estimate of the error assuming the higher terms are small).

Graph	Integral
Square Data (SD) (Figure 2.2)	164.8 ± 8.9
Original Algorithm (OA) (Figure 2.3)	147.3 ± 9.2
Revised Algorithm (RA) (Figure 4.2)	183.5 ± 8.7
Absolute Difference Between Square Data (SD) and Original Algorithm (OA) (Figure 5.1)	27.3 ± 4.4
Absolute Difference Between Square Data (SD) and Revised Algorithm (RA) (Figure 5.2)	21.1 ± 3.4

Table 1: Riemann sum approximations to integrals to quantitatively assess the difference between the Revised Algorithm (RA) and the Original Algorithm (OA) compared to the Square Data (SD), over a 32x32 cm area

Table 1 shows that when compared to the SD, the RA is better than the OA. The value of the absolute difference between the RA and SD is slightly smaller than that of the OA and SD, which reflects the fact that in general the RA is better at approximating the dose since it doesn't have the 'double hit' areas. Also the integral of the RA gives a higher value than that of the SD, while the OA gives a lower value. It is better to have a calculation that

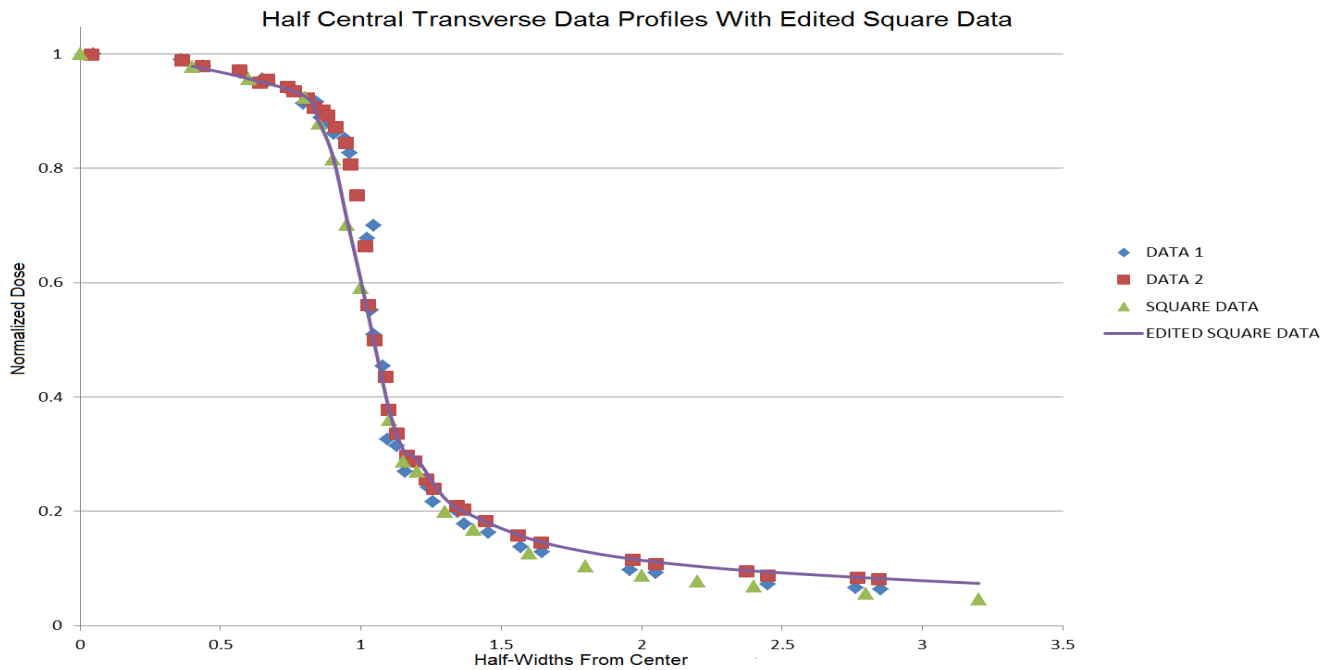

Figure 5.4: Half center data transverse profiles (HCTPs) of 10x10 at z=10 with edited Square Data to match Data 2. The y axis represents the nomalized dose, while the x axis represents the number of half-widths (half the width of the colimator) away from the center

overestimates the dose value than one that underestimates it since this prevents radiating healthy tissue more than expected.

Also, the covariances between both the OA and RA with the SD were computed in order to have a quantitative measure of shape relation between them (to see how well the fits follow the general trend of the data). The covariance between the OA and the SD is of 0.9943 ± 0.00005 and that between the RA and the SD is of 0.9977 ± 0.00005. Even though both values are really high, this shows that the RA is indeed better than the OA.

To get a quantitative difference of the OA and RA with values that are closer to those of Data 2 (which would show the highest difference of any of the recorded data) and that account for the displacement factor, the following was done. First, using the HCTPs, the SD was changed to see if it could be modified to have values closer to those of Data 2. By adding a small amount and renormalizing it in order to account for the displacement factor, the graph of the SD was found to closely follow Data 2 (see Figure 5.4), especially after 1 half-width, which is ideal since the Cauchy starts after 1.15 half-widths. Using this knowledge, the entire SD was modified. Then using this edited SD (denoted as ESD), a similar table as the previous one was created (see Table 2).

Graph	Integral
Edited Square Data (ESD)	190.2 ± 8.7
Original Algorithm (OA)	147.3 ± 9.2
Revised Algorithm (RA)	183.5 ± 8.7
Absolute Difference Between Edited Square Data (ESD) and Original Algorithm (OA)	47.9 ± 4.4
Absolute Difference Between Edited Square Data (ESD) and Revised Algorithm (RA)	11.8 ± 3.4

Table 2: Riemann sum approximations to integrals to quantitatively assess the difference between the Revised Algorithm (RA) and the Original Algorithm (OA) compared to the Edited Square Data (ESD), over a 32x32 cm area

From Table 2 it can be seen that the RA is much better than the OA. The absolute difference between the RA and the ESD is significantly smaller than that between the OA and the ESD. Also the RA integral is much closer to that of the ESD, showing how it is a more appropriate approximation.

6 Conclusion

Both the quantitative and qualitative results mentioned above reveal the Revised Algorithm (RA) as a better fitting scheme than the Original Algorithm (OA). It is able to more closely approximate the fallout of the dose, improving not only the 'double hit' areas, but also other fundamental flaws of the OA. This RA will enable a more precise approximation to the dose, which in turn makes it possible to create a better treatment plan for patients.

.....

Special thanks to Dr. Thomas Kroc for his constant support and patient supervision. Also to the SIST committee and collaborators, and Fermi National Laboratory for allowing this wonderful opportunity.

References

[1] Huan Giap; Bosco Giap. Historical perspective and evolution of charged particle beam therapy. *Translational Cancer Research*, 1(3), October 2012. URL `http://www.thetcr.org/article/view/595/html`.

[2] Keith E. Holber. Radiation effects and damage. Technical report, Arizona State University, 2010. URL `http://holbert.faculty.asu.edu/eee560/RadiationEffectsDamage.pdf`.

[3] L. Cohen; F.R. Hendrikson. Neutron therapy for nonresectable radioresistant tumors. *Arch.Surg.*, 124:294–300, 1989.

[4] MA The MathWorks Inc., Natick. Matlab 2013a.

[5] O. Homburg; T. Mitra. Gaussian-to-top-hat beam shaping: an overview of parameters, methods, and applications. *Proceedings SPIE 8236, Laser Resonators, Microresonators, and Beam Control XIV*, 8236A, February 9 2012.

[6] Anthony A. Trovar. Improvement of laser efficiency using mode selection techniques. Technical report, Eastern Oregon University, 1999. URL `http://physics.eou.edu/~atovar/research/ss/ss99.html`.

[7] Miguel Awschalom; Ivan Rosenberg; Randall K. Ten Haken. A new look at displacement factor and point of measurement corrections in ionization chamber dosimetry. *Med. Phys.*, 10(3):307–313, May/Jun 1983.