AN ALGORITHM FOR TWO DIMINUTIONS DENSITY RECONSTRUCTION IN PROTON COMPUTED TOMOGRAPHY (PCT)

A Thesis

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

---

by

Jihad Tafas

December 2007

AN ALGORITHM FOR TWO DIMINUTIONS DENSITY RECONSTRUCTION IN PROTON COMPUTED TOMOGRAPHY (PCT)

A Thesis

Presented to the

Faculty of

California State University,

San Bernardino

---

by

Jihad Tafas

December 2007

Approved by:

Dr. Keith Evan Schubert, Advisor, Computer Science	Date
Dr. Arturo Concepción	
Dr. Ernesto Gomez	
Dr. Reinhard W. Schulte	

ABSTRACT

The growth of Proton therapy requires new imaging and treatment planning modalities. X-ray computed tomography (xCT), which is widely available, has been used for the treatment planning for proton therapy, but since the basic interactions of xCT in matter are fundamentally different than those of the protons, the resulting density map from xCT is only an approximation. Accuracy of the electron density map is crucial to successful use of proton therapy, thus requiring proton computed tomography (pCT), which accurately maps the electron density.

The image reconstruction problem for pCT is to obtain the best estimate for the relative electron density map from the measured proton data. The problem is not exactly solvable because of two factors: (1) the statistical fluctuation of the measured energy loss mainly due to energy straggling, and (2) the statistical deviation of the proton from its most likely path (MLP) due to multiple Coulomb scattering.

This thesis develops an optimized and effective iterative reconstruction algorithms taking into account the peculiarities of proton transport through the object, and hardware acceleration methods need to work together synonymously in order to be suitable for clinical applications. Algebraic reconstruction technique (ART) has shown some promise in the literature, but its theoretical basis does not fit with the assumptions of pCT. This thesis explores the assumptions and practical reconstruction of the electron density maps. In particular the performance in terms of reconstruction time, and the parallelizability will be examined.

ACKNOWLEDGEMENTS

Thank you is not enough for those who supported me to reach this level of education.

I would like to acknowledge my committee members. Firstly, Dr. Concepcion, "The Organizer", who taught me planning and scheduling and time management. I would like to thank Dr. Gomez to whom I give credit for all my education about parallel programming and distributed systems. Many thanks for Dr. Schulte who is taking the lead in Proton Computed Tomography.

I would like also all the faculty members at the Department Of Computer Science of California State University, San Bernardino. I would recognize Dr. Richard Botting who taught me my first computer class in USA, Dr. Josephine Mendoza the best graduate coordinator. Not to forget Dr. Owen Murphy, Dr. George Georgiou and Dr. Yasha Karant and all the other Doctors in the Department of Computer Science and other university departments.

I would like to give special thanks to Mrs. Monica Latimer, the best and most supportive department secretary, and Mr. Kwon Soo Han, the best System Administrator, for his IT support though out my BS and my Master years in the department. I would like to thank also the International Extension Program (Jack Linda Chaffee, Joyce Woolsey, and Jorge Razo), the International Admission Office (Cindi Sham), Dean Sandra Kamusikiri, Shelby Reeder from the graduate office, Special Assistant to VP for Advancement Cynthia Pringle, Robert Tiberi EOC Coordinator, Sunny Hyon the best English professor on campus, and the President office for their wonderful support though out my education years at California State University, San Bernardino.

To my tremendous parents, mother and father Tafas, mother in law Kakab, to my sisters Gohaina, Wedad and Reem, to Brothers Yousef Waseem and Jommar, to my nieces Kirstein and Sandra, my nephews Foad and Simon, to all my friends who are my family in America.

Elen, my love, thank you for all the coffee, tea, and the food. Thank you for staying up with me throughout the last two years, to finish this thesis. Thank you for keeping up with me.

Lastly, special thank to Dr. Keith Schubert. Thank you for making advance research more enjoyable to challenge, unknown results more fun to explore, and Math has never been so fun to analyze. Thank you for being there when I needed advice. I hope that my success in my career will pay you back.

To you all, thank you as many as they are “zeros” and “ones” in every computer on the planet.

DEDICATION

To Mary the Soul, the Smile, the Spirit, and the Baby.

Sister Mary “03-18-1974” “09-24-2001”

Baby Mary “08-11-2007”

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. BACKGROUND

1.1 Introduction

Proton computed tomography (pCT) has been explored in the past decades because of its unique imaging characteristics, low radiation dose, and its possible use for treatment planning and on-line target localization in proton therapy [**?**, **?**]. The use of protons for medical imaging was first suggested in the late 1960s by Koehler [**?**]and first experimental work on the concept of a pCT scanner was performed at the Los Alamos National Laboratories in the late 1970's [**?**, **?**] but pCT was never fully developed because of great advances in x-ray CT (xCT) and other imaging modalities. In recent years, pCT has gained relevance because proton treatment centers opened and rotatable proton gantries became available [**?**, **?**].

In his pioneering work, Koehler showed that minute density differences, for example, the addition of a 0.035 g cm^{-2} thick aluminium foil to a stack of aluminum absorbers 18 g cm^{-2} thick, could be discerned by means of 2D-projection proton radiography using radiographic film as the detector. Subsequently, Steward and Koehler (1973a,b 1974) and others (Cookson 1974, Moffett et al 1975, Kramer et al 1979) demonstrated that the high contrast images obtained by proton radiography provided improved imaging of low contrast lesions in human specimens over conventional x-ray techniques at comparable exposure levels. The high contrast obtained by imaging the

energy loss of protons with radiography is a consequence of the sharpness of the well known Bragg peak that occurs near the end of proton range. Even higher contrast can be achieved through the use of heavy ions rather than protons(Benthon et al 1975, Capp et al 1978) [**?**].

Although first suggested by Cormack in 1963 (Cormack 1963, 1964) as a possibility to do tomographic reconstructions with proton imaging, the first author that investigated pCT experimentally was Goitein (1972). He employed projection data with alpha particles measured by Lyman to demonstrate the utility of his least-squares reconstruction algorithm for pCT. Later, in comparisons of heavy charged particles CT (including pCT) with x-ray CT (Crowe et al 1975, Huesman et al 1975, Cormack and Koehler 1976, Hanson 1978) it was shown that charged particles have a clear dose advantage over x-rays [**?**]. This dose advantage might be utilized clinically by providing CT reconstructions with significantly better density resolution than it is possible with x-rays at a given dose level. Furthermore, in charged particle CT, it is the electron density relative to water what is imaged rather than the photon attenuation coefficient relative to water that is used in x-ray CT. This unique imaging characteristics of charged particles may prove also to be beneficial in medical diagnosis [**?**].

X-ray computed tomography (CT) brought slice imaging into wide use for the first time and represented its breakthrough. Today CT is an essential part of radiological diagnostic and can be seen as a mature and clinically accepted procedure. It has supplemented or replaced classical x-ray imaging in many areas [**?**].

A rapid technical development phase in the seventies was followed by an uneventful phase with no essential highlights in the eighties. this was partly caused by the expec-

tation that the importance of CT would decrease successively due to the introduction of magnetic resonance (MR) tomography. Contrary to these expectations, CT is in the phase of rapid technical developments and again broadening its application spectrum. The development of spiral CT and the transition from scanning single slices to the raid scanning of complete value has made CT attractive again and has led to decisive developments in technical and in clinical perspectives. The introduction of multi-row detector systems and scan times in the sub-second range constitute the high point of these developments.

Since x-ray CT is widely available, and proton facilities are still limited to a few centers world wide, current attention is directed to therapeutic applications of pCT [**?**]. Proton therapy is an advanced form of radiation therapy which offers proven advantages over radiation therapy with photons (e.g., high energy x-rays) [**?**]. Treatment planning for proton therapy is currently based on x-ray CT, which has disadvantages because the mapping of Hounsfield CT values to electron density is not unique [**?**]. Furthermore, it would be advantageous to use protons for image guidance in the treatment room for several reasons, including the dose advantage of protons and avoidance of alignment problems between imaging equipment and radiation source.

pCT applied in proton treatment planning would be advantageous because it directly reconstructs the electron density values and uses the same radiation modality that is being used for treatment. Therefore, pCT in combination with proton radiation therapy may lead to ultimate form of image-guided radiotherapy. pCT is generally based on similar underlying principles as other medical imaging modalities but differs in some aspects from x-ray CT. The imaging information of pCT is the

energy loss of individual protons rather than the attenuation of a proton. From this information, the integrated density along the proton path can be estimated. The most important difference with respect to xCT is that protons undergo multiple scattering inside the object and, therefore, follow paths that statistically deviate from straight lines. Because of this different reconstruction algorithms than those relying on straight path assumptions have to be used.

In x-ray CT, data collection is considered mathematically as a Radon transform, i.e., the integration along straight lines of the object source function. In this case, the object data represent the attenuation coefficient map and the projection data the log values of the detected x-ray counts. In pCT, protons with known entry energy are tracked individually on the entry and exit side of the object and their outgoing energy is recorded. This can be achieved with modern particle tracking detectors developed for high-energy physics applications. Because of the random nature of proton scattering, it is not possible to calculate the precise trajectory of the protons within the target, but the entry and exit locations and directions of the protons can be used to estimate their path through the object, and the measured energy loss permits estimating the integrated electron density along the proton path.However, because most of the technological development efforts successfully went into improving the diagnostic xCT in those decades, the interest in developing medical pCT stagnated.

The situation changed with the development of medical proton gantries for delivery of proton beams, first at Loma Linda University Medical Center, and now in several other proton treatment centers, resulting in an increasing number of patients treated with proton therapy. This new technical development and increase of patient number

elevated the need for an accurate prediction of the proton dose distributions and verification of the patient position on the treatment table, and also demanded the development of accurate 3D imaging techniques. This has lead to a renewed interest in proton imaging and the construction of a proton radiography system at the Paul Scherrer Institute in Switzerland [**?**].

The main goal of pCT for proton therapy applications is to determine of the electron density distribution of the object and to use this information for treatment planning and image-guidance in the treatment room. The image reconstruction problem for pCT is then to obtain the best estimate for the relative electron density map from the measured proton data. The problem is not exactly solvable because of two reasons: (1) the statistical fluctuation of the measured energy loss mainly due to energy straggling, and (2) the statistical deviation of the proton from its most likely path (MLP) due to multiple Coulomb scattering. Nevertheless, approximate solutions of the electron density map can be found by iteratively solving the reconstruction problem.

Presently, a pCT system utilizing a proton gantry and fast image reconstruction techniques has not yet been developed. However, a recently published design study has concluded that a pCT scanner should utilize instrumentation developed for high-energy physics such as silicon track detectors and crystal calorimeters equipped with fast readout electronics, allowing one-by-one registration of protons traversing the body during a full revolution of the proton gantry [**?**]. Different from proton beam therapy where the Bragg peak is positioned inside the targeted tumor inside the body, pCT may label each incident proton and detect that proton when it exits from the

body by a high-energy detector where the high-dose Bragg peak will occur inside the detector. The pCT scanner will provide precise information on the protons incident energy, location and direction, as well as its exit energy, location and direction. Another recently published study further concluded that a completely new image reconstruction paradigm is needed for pCT which deals with the proton path of curves, rather than the well-known X-ray path of straight lines in xCT [**?**]. An adequate image reconstruction algorithm shall utilize the pCT scanner measurements to map the energy loss along the proton trajectories through the body.

A major challenge in pCT that requires research and development is to investigate and optimize reconstruction algorithms. Researchers, using the Monte Carlos simulation tool GEANT4 and an elliptical object model have shown that, in principle, an algebraic reconstruction technique (ART) can lead to a satisfactory spatial resolution [**?**]. However, there are many variants of algebraic reconstruction techniques (also called series expansion methods) that should be explored and optimized for the pCT application. These are expected to differ in terms of computer speed, possibility to perform parallel computations, and the accuracy of the reconstruction. The goal of this thesis is to establish the computer science basis for this research allowing future students and researchers to further explore this important field.

1.2 Significance

The number of proton treatment centers has dramatically increased in recent year. For many years, the proton treatment center at Loma Linda University Medical Center, which opened in 1990, was the only hospital based center for proton therapy, but now

there are about 10 such facilities world wide.

With increasing use of protons for the treatment of patients, doctors want to exploit their unique ability to stop the beam immediately in front of a critical structure such as the optic nerve or the spinal cord. This requires very accurate treatment planning and the ability to predict where the proton beam will stop inside the patient. With current x-ray CT based treatment planning the accuracy of proton range predictions is not better than a few millimeters which often causes the radiation oncologist to avoid beams the that would stop in front of critical structures. Such major limitations can be over come by developing pCT.

The knowledge of pCT reconstruction is currently very limited and only a few publications on this subject exist. Therefore, more work in this field is needed and was initiated with this thesis project. Development of time-efficient computer algorithms is the task of computer scientists and this thesis will be significant for further development and optimization of reconstruction in pCT.

A successful implementation of pCT would avoid the ambiguities of mapping xCT Hounsfield units (HU, which is related to the X-ray attenuation coefficients) to electron densities, and would allow actual dose distribution and also verification of patient position in the treatment room. In other words, the availability of pCT in the treatment room will predict very accurately the position of the Bragg peak within the patients body, resulting in a maximum dose delivery to the targeted tumor and successful sparing of the surrounding normal tissues. Furthermore, a successful integration of pCT with proton therapy may lead to the ultimate form of image-guided 3D conformal radiation therapy, which has the potential to deliver the optimal dose to

any point within the patient and provide arbitrarily shaped inhomogeneous dose distributions as desired This is now recognized as the major potential of pCT in medicine. The image formation principles of proton beam are presented below. Hardware configuration and data acquisition for pCT will be discussed. The emphasis will be on the issue of image reconstruction from projected data along proton trajectories through the body.

1.3 Purpose of The Thesis

Preliminary work clearly points to algebraic techniques or series expansion methods for proton CT reconstruction. So far, only an additive algebraic reconstruction technique (ART) has been attempted on a single object. Other algebraic or series expansion techniques may give better results. These techniques have parameters that can be optimized, which has not been done yet. Furthermore, reconstruction has only been done of a 2-D object and with simulated data, this needs to be extended to a 3D objects and to data that have been acquired with an experimental pCT prototype. Lastly, one of the challenges of the algebraic reconstruction or series expansion techniques is their high computation cost (e.g., twelve hours computation time for an additive ART reconstruction on a laptop PC [**?**]). This should be accelerated using parallel computing techniques and/or hardware accelerators but the performance of these algorithms will also depend on the choice of relaxation factors and the particular object type. Clearly, much research is still needed before pCT can be clinically used to the benefit of patients.

The purpose of this thesis is to establish the basis for this research by developing

the software that uses simulated or experimental proton CT raw data as input and calculates most likely paths of protons through the object. It is the plan to use one or two algebraic reconstruction methods to demonstrate that the original object data can be reconstructed with sufficient spatial resolution and without artifacts. Based on the findings of this work recommendations for future directions will be given.

1.4 Assumptions

The following assumptions were made throughout this thesis.

1. The electron density distribution relative to water of the object to be reconstructed using proton CT raw data is exactly known. Differences in the reconstructed object are, therefore, due to imperfections in the raw data, the reconstruction algorithm, or both.

2. The protons incident on the object are monoenergetic and have an energy of 200 MeV.

3. The incident beam consists of protons that are all parallel in their direction and lie in a common plane. The object is reconstructed in this plane.

1.5 Limitations

Note: *This thesis is one of the first in the field of pCT reconstruction. Below I stated only a few known limitations that must be considered when viewing the results. Additional limitations will be stated throughout the remainder of the thesis as they become obvious.*

The spatial and electron density resolution of a pCT scanner are inherently limited by the physical processes of multiple Coulomb scattering (MCS) and energy loss straggling. A clinically meaningful spatial resolution for proton therapy is about 1mm [**?**] and I chose this limit for the voxel size of my pCT reconstructions.

My work focuses on building the first version of a software platform for pCT reconstruction that can be extended to more and more complex algorithms and applications in the future. As such I have limited the first version of the platform to reconstruction of a 2D object scanned with parallel, monoenergetic proton beams and without assuming any technical imperfections in the imaging system.

1.6 Definition of the Terms and Abbreviations

These are standard terms in the field and are included for the convenience of the reader.

2-D Related to a two-dimensional plane.

3-D Related to a three dimensional volume.

Algorithm A set of ordered steps for solving a problem, such as a mathematical formula or the instructions in a program.

ART Algebraic reconstruction technique is used to reconstruct and object from projection data; it is based on matrix algebra. anything.

CERN A large multinational high-energy physics laboratory in Geneva, France.

CT Computed tomography is a technology that allows to reconstruct an object and displays the reconstruction as a stack of 2D images.

DSP Digital Signal Processing.

Electron Density Electron density is the number electrons per unit volume present at a given location of an object.

FPGA Field Programmable Gate Array.

GEANT4 The GEANT4 code developed at CERN is a platform for "the simulation of the passage of particles through matter." It is the most recent in the GEANT series of software toolkits developed by CERN, and the first to use Object oriented programming (in C++).

GPU Graphics Processing Unit.

MCS Multiple Coulomb scattering occurs when a charged particle traverses and interacts with the nuclei of matter.

MLP Most likely path of a proton through a reconstruction volume when only entry and exit position and direction are known.

MRI Magnetic Resonance Imaging, which is diagnostic technique which uses a magnetic field and radio waves to provide computerized images of internal body tissues magnetic resonance imaging. MRIs are used in medicine to help diagnose things that won't show up on an X-ray.

NIST The National Institute of Standards and Technology is a federal technology agency that develops and promotes measurement, standards, and technology.

Particle or atom (nontechnical usage) a tiny piece of anything.

PC Personal Computer.

pCT Proton Computed Tomography is a novel imaging technique that uses protons to perform CT.

PET Positron Emission Tomography, is a diagnostic examination that involves the acquisition of physiologic images based on the detection of radiation from the emission of positrons.

Proton accelerator A proton accelerator is a device that uses electric fields to accelerate protons to a high speed close to the velocity of light and magnetic fields to keep them on a circular path.

Proton Gantry A proton gantry is a medical device that bends the proton beam and allows treating a patient from any direction within a plane of rotation.

Proton Therapy An advanced from of radiation therapy that employs protons to treat tumors and other conditions.

Quadrature A method of numerical integration, often applied to one-dimensional integrals.

Relative Electron Density Electron density relative to that of water. This allows to use unit-less numbers in the vicinity of unity for the reconstruction of patient images.

SPECT Single photon emission computed tomography, is a nuclear medicine tomographic imaging technique using gamma rays.

Tomography A method to generate 2D images of areas inside the body; in computed tomography these images are created by a computer.

x-rays Energetic photons of short wave lengths generated by bombarding electrons onto a dense metallic target. X-rays are used for CT and radiation treatment.

1.7 Organization of the Thesis

In this thesis I will be researching various series expansion reconstruction techniques for pCT. In particular, I will perform the following tasks: **The first step** involves implementation of various pCT reconstruction algorithms in 2-D and testing their performance with GEANT4-simulated pCT data. I will optimize the performance by systematically varying relaxation parameters and iterative refinement steps. This first task is anticipated to take about 4 months. **The second step** is to improve the timing of the selected 2-D algorithm using numerical and parallel processing techniques. This is anticipated to take about 4 months. **The third step** is to extend the optimized 2-D algorithm to three-dimensional objects. This is anticipated to take two months. After conclusion of these steps, I will suggest the path to be followed from that point on. I will look at possible hardware implementation using Field Programmable Gate Array (FPGA), Digital Signal Processing (DSP), and Graphics Processing Unit (GPU) to speed up the process. Some basic tests and analysis will be performed for one month. During the final month, conclusions will then be drawn and recommendations for further work be given.

2. CONVERSION OF ENERGY LOSS TO THE LINE INTEGRAL OF RELATIVE ELECTRON DENSITY

2.1 Background

The history of heavy charged particle radiography began in 1968 with the pioneering work of Koechler (1968). He showed that the addition of an aluminum foil 0.035 gcm^2 thick to a stock of aluminium absorbers, 18 gcm^{-2} thick, could be discerned by means of proton radiography using film as the detector [**?**]. Subsequently, Steward and Koehler (1973a,b, 1974) and others (Cookson 1974, Moffett et al 1975, Kramer et al 1979) did a lot of work to finally publish results that the high contrast images obtained by proton radiography provided improved imaging of low contrast lesions in human specimens over conventional x-ray techniques. The high contrast obtained in this energy-loss form of radiography is a consequence of the sharpness of the well known Bragg peak that occurs near the end of the proton range. Every higher contrast may be achieved through the use of heavy ions [**?**].

Although cited as a possibility by Cormack in 1963, the first to apply charged particles to computed tomography (CT) was Goitein in 1972. He employed projection data measured by Lyman with alpha particles to demonstrate the utility of his least-squares reconstruction algorithm. Later, in the comparison of heavy charged particle CT with x-ray. It was shown that charged particles have a dose advantage over x-

Fig. 2.1: Gantry-1 Proton Treatment Room at Loma Linda Medical Center and University LUMC

rays. This does advantage might be utilized effectively by providing CT reconstruction with significantly better density resolution than it possible with x-rays at a given dose level. Furthermore, in charged particles CT, it is the linear stooping power relative to water that is imaged rather than x-ray attenuation coefficient. The unique imaging characteristics of charged particles may prove to be beneficial in medical diagnosis.

The current proton therapy system at LLUMC is illustrated in figure 2.1 [**?**], where proton gantry delivers proton from any angle around the patient. It delivers wide range of beam intensities, of which protons used for pCT will penetrate the patient. During the gantry rotation, data including, entry and exit location, angle, and energy

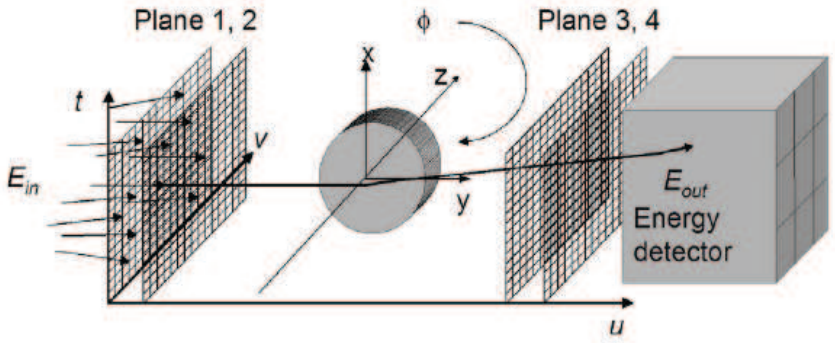

Fig. 2.2: Schematic of the Proposed Approach to Proton Computed Tomography pCT with known entry energy E_{in} [**?**]

loss will be collected.

The main principle of proton computed tomography pCT is based on the determination of the integrated volume electron density, ρ_e, or short, the electron density. This is accomplished by measuring the energy loss of individual protons after traversing the image object. The electron density of a material at a given location is defined as the number of electrons/cm^3.

Schulte, from Loma Linda Medical Center and University, and colleagues from LLUMC and other research centers [**?**], suggested a conceptual design for a proton computer tomography system illustrated in 2.2. The object is traversed by a broad beam of protons with known energy E_{in}. A proton-tracking detector is arranged on both sides of the patient, which records the entrance and exit points and angles of individual protons. Protons are stopped in a scintillator crystal array to measure their outgoing energy.

The figure 2.2 present a schematic of the proposed approach to pCT with known entry energy E_{in} are recorded one by one in the detector reference system (t,u,v) as they traverse the image object from many different projection angle ϕ. The recorded data include entry and exit positions and entry and exit angles as well as exit energy E_{out} in the energy detector. In this figure, planes (1 and 2) register the location and direction of each proton on the entry side, planes (3 and 4) register the location and direction of each proton on the exit side, last we have energy detector measures residual energy of each proton on the exit side

This chapter deals with the development of the analytic formalism allowing an computer-time efficient conversion of the measured energy loss to the integrated electron density along the proton path.

2.2 Interaction of Protons With Matter

When traversing matter, protons lose some of their energy via inelastic collisions with the outer electrons of the target atoms leading to ionizations and excitations. Furthermore, they will be deflected by multiple small-angle scattering events (i.e., multiple Coulomb scattering –MCS) from the atomic nuclei. These two main processes, occurring a great number of times along the macroscopic path of the protons, lead to the macroscopic effects of the interaction of protons with matter: (1) loss of energy and (2) deflection from their original direction. As individual interaction events occur randomly, these two processes result in a statistical distribution of the following two principal quantities observed for proton imaging: (1) the amount of energy lost by each proton after traversing the body, and (2) the lateral and angular

displacement of the proton from its incident position and direction. The amount of energy-loss variation (i.e., energy straggling, which is reflected by the variation of the Bragg peak location of a proton traversing along the same path through the same object) is the principal limitation for the intrinsic image contrast or density resolution of pCT (Satogata et al. 2003; Schulte et al. 2005). The variation of proton trajectory due to the random MCS, resulting in the lateral and angular displacements, is the principal limitation for the intrinsic image spatial resolution of pCT. These two principal limitations are discussed in more detail in [?] [?] [?].

In addition to the two main processes of inelastic collisions with the outer atomic electrons and elastic deflection from the atomic nuclei due to MCS, protons in the energy range (at the MeV level) used for pCT also undergo inelastic nuclear interactions, leading to reduction of proton transmission in a depth-dependent manner. Protons undergoing nuclear interactions mostly deposit their energies locally and hence contribute to the dose within the patient without contributing to the image formation. This is important for developing pCT for clinical use, but the magnitude of this effect is well understood and contribution to patient dose is relatively small (Schulte et al. 2005). For example, the probability of a 200 MeV proton to be transmitted without undergoing a nuclear interaction is 92.2% for a water layer of 10 cm thickness, and 83.6% for a layer of 20 cm thickness. Thus the majority of protons will contribute to the image when using energy loss as the physical quantity to reconstruct the electron density along the path.

2.3 Energy Loss and Electron Density: The Bethe Bloch Equation

As we will see below, the energy loss of protons going through an object is closely related to the path integral of electron density. This relationship can be exploited to reconstruct the object distribution of electron density in 3D.

The protons used for pCT must have sufficient energy to penetrate the body to be imaged. According to the NIST PSTAR data base [1], the path depth or range in a continuous slowing-down approximation (CSDA) of 200 MeV protons in a tissue equivalent plastic is 25.8 cm, which is sufficient to penetrate an adult human skull (nominal width of 20 cm in anterior-posterior direction). For 250 MeV protons the range is 37.7 cm, sufficient to penetrate an adult trunk (nominal width of 34 cm, excluding arms).

The energy-loss method of pCT is based upon the relationship between electron density and energy loss per unit track length. Let us first consider the relationship between energy density and physical density, which is is given by:

$$\rho_e = \rho N_A \left(\frac{Z}{A}\right) \tag{2.1}$$

where ρ is the physical density, N_A is Avogadro's number (6.023×10^{-23}), and Z and A are the (effective) atomic number and atomic weight of the traversed material, respectively. When the object material is comprised of a compound, for example, water(H_2O), or a mixture of elements or compounds, the electron density is given by:

$$\rho_e = \rho N_A \Sigma_i w_i \frac{Z_i}{A_i} \tag{2.2}$$

where w_i is the fraction by weight of element i, and Z_i and A_i are the atomic number

[1] http://www.physics.nist.gov/PhysRefData/Star/Text/PSTAR.html

and atomic weight of the ith element. The elemental and compound data may be obtained on-line from the NIST material composition database [**?**].

Since the ratio Z/A is fairly constant for human tissues, the electron density closely reflects the physical density of the imaged tissue [**?**]. To avoid the large numbers associated with absolute electron density values, which are of the order of 10^{23} electrons/cm^3, it is better to express results in terms of relative volume electron density, which is defined as:

$$\eta_e = \frac{\rho_e}{\rho_{e,water}} \tag{2.3}$$

where $\rho_{e,water} = 3.343 \times 10^{23} electrons/cm^3$ is the electron density of water.

The energy loss per unit track length of a proton, also called the stopping power, S, is described by the Bethe Bloch equation [**?**] as:

$$S(u) = -\frac{dE}{du}(u) \tag{2.4}$$

$$= \eta_e(u) F(I(u)E(u)) \tag{2.5}$$

where u represents the penetration depth of a proton, $E(u)$ is the energy, $\eta_e(u)$ the relative electron density defined above, and $I(u)$ the mean excitation potential of the object material at depth u. The mean excitation potential is a material-dependent constant that may change with penetration depth if the object's atom composition changes with depth. Its value is also given in the NIST material composition data base [**?**]. One should note that the mean excitation potential is similar for most human tissues and, therefore, its value for water of the human body ($I_{water} = 75eV$) may be used as a representative value for human tissues.

The function $F(I,E)^2$ in equation 2.4 is defined as [?]:

$$F(I,E) = K\frac{1}{\beta^2(E)}\left[\ln\left(\frac{2m_ec^2}{I}\frac{\beta^2(E)}{1-\beta^2(E)}\right)-\beta^2(E)\right] \tag{2.6}$$

where $m_ec^2 = 511.011keV$ is the electron rest energy and $\beta(E)$ is the proton velocity relative to the speed of light c. The constant K is defined as:

$$K = 4\pi u_e^2 m_e c^2 \rho_{e,water} = 0.170\frac{MeV}{cm} \tag{2.7}$$

where $r_e = 2.818\times10^{-13}cm$ is the classical electron radius. The relationship between β and E is given by the relativistic relationship [?]:

$$\beta(E) = \sqrt{1-\left(\frac{E_p}{E+E_p}\right)^2} \tag{2.8}$$

where $E_p = 938.29459$ MeV is the proton rest energy.

Note that the Bethe-Bloch equation 2.4 is a non-linear first order differential equation of the function $E(u)$. Since $I(u)$ is usually not exactly known because the object composition is unknown, integration of this equation is only possible under the assumption that $I(u) = const.$ As discussed above, for human tissues encountered in proton CT, the variation of $I(u)$ with penetration depth is not very large, and the function F has only a week logarithmic dependence on $I(u)$. Therefore, the assumption of a constant value of 75.0 eV, the mean excitation potential for water, is reasonable. In this case, F becomes function of E only, and one can separate the terms depending on the variable u and the variable E:

$$\eta_e(u)du = -\frac{dE}{F(E,I_{water})} \tag{2.9}$$

[2] Note that the formula given here is only an approximation of the original Bethe Bloch equation, which contains a term Wmax, the maximum energy transfer in a single collision. The approximation given here is valid if the mass of the incident particle is large relative to the electron mass, which is the case for protons (mp/me . 1800).

Integrating the left side of the equation with respect to the penetration depth u along the proton path and the right side with respect to the energy between initial energy E_{in} at the beginning of the path and E_{out} at the end of the path, we get:

$$\int_{u_{in}}^{u_{out}} \eta_e(u)du = -\int_{E_{in}}^{E_{out}} \frac{dE}{F(E, I_{water})} \tag{2.10}$$

$$= \int_{E_{out}}^{E_{in}} \frac{dE}{F(E, I_{water})} \tag{2.11}$$

It is now obvious that the integral of the relative electron density along the proton path can be calculated based on the knowledge of in-and outgoing proton energy. Due to the complicated energy dependence of $\mathbf{F}$, the integration needs to be performed numerically or modeled as a polynomial equation. Also note that the integrated density along the proton trajectory is nothing else than the water-equivalent length of the proton track through the medium [**?**]. Unlike the inversion in xCT, the proton path is unknown in pCT due to MCS and must be estimated. This uniqueness of pCT renders a challenge for image reconstruction from the projection data along an unknown path. In addition to this challenge and the approximation made for 2.4 from the original Bethe-Bloch equation, it shall be further noted that the integrated density along the proton is approximated by the water-equivalent length of the proton trajectories through the body because of the use of I_{water}.

2.4 Simplified Version of The Bethe Bloch Equation

One goal of my thesis was to mathematically simplify the integral in equation 2.10, in particular the Bethe Bloch equation term $F(E, I)$. The gain will be an increase in performance by reducing the calculational errors and speeding up the reconstruction.

Here is the final version of the simplified formula that was derived starting from equation 2.10:

$$\int_{u_{in}}^{u_{out}} \eta_e(u)dx = \int_{E_{out}}^{E_{in}} \frac{dE}{K\frac{(E+E_p)^2}{E^2+2EE_p}\left[\ln(2m_ec^2) - \ln(I_{water}) + \ln(E+2E_p) + \ln(E) - 2\ln(E_p)\right] - K} \quad (2.12)$$

2.4.1 Proof of the Simplified Version of Bethe-Bloch Equation

The following is a proof of equation 2.12. Starting from the equation 2.10 we write:

$$\begin{aligned}
\int_{u_{in}}^{u_{out}} \eta_e(u)dx &= -\int_{E_{in}}^{E_{out}} \frac{dE}{F(E, I_{water})} \\
&= \int_{E_{out}}^{E_{in}} \frac{dE}{F(E, I_{water})} \\
&= \int_{E_{out}}^{E_{in}} \frac{dE}{K\frac{1}{\beta^2(E)}\left[\ln\left(\frac{2m_ec^2}{I_{water}}\frac{\beta^2(E)}{1-\beta^2(E)}\right) - \beta^2(E)\right]}
\end{aligned}$$

First, let us consider the term $equ_b = \ln\left(\frac{2m_ec^2}{I_{water}}\frac{\beta^2(E)}{1-\beta^2(E)}\right)$. Taking into account that

$$\begin{aligned}
\ln(a \cdot b) &= \ln(a) + \ln(b) \\
\ln(\frac{a}{b}) &= \ln(a) - \ln(b)
\end{aligned}$$

and using the definition of β in equation 2.8, we get

$$\begin{aligned}
equ_b &= \ln\left(\frac{2m_ec^2}{I_{water}}\frac{\beta^2(E)}{1-\beta^2(E)}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\frac{1-\frac{E_p^2}{(E+E_p)^2}}{1-\left(1-\frac{E_p^2}{(E+E_p)^2}\right)}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{1-\frac{E_p^2}{(E+E_p)^2}}{1-\left(1-\frac{E_p^2}{(E+E_p)^2}\right)}\right)
\end{aligned}$$

$$
\begin{aligned}
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{1-\frac{E_p^2}{(E+E_p)^2}}{\frac{E_p^2}{(E+E_p)^2}}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{1}{\frac{E_p^2}{(E+E_p)^2}} - 1\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{(E+E_p)^2}{E_p^2} - 1\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{(E+E_p)^2}{E_p^2} - \frac{E_p^2}{E_p^2}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{E^2+2E_pE+E_p^2-E_p^2}{E_p^2}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{E^2+2E_pE}{E_p^2}\right) \\
&= \ln\left(\frac{2m_ec^2}{I_{water}}\right) + \ln\left(\frac{E(E+2E_p)}{E_p^2}\right) \\
&= \ln(2m_ec) - \ln(I_{water}) + \ln(E+2E_p) + \ln(E) - 2\ln(E_p)
\end{aligned}
$$

Next, we substitute this term and the term for β into 2.10 and obtain:

$$
\begin{aligned}
\int_{u_{in}}^{u_{out}} \eta_e(u)dx &= \int_{E_{out}}^{E_{in}} \frac{dE}{\frac{K}{\beta^2(E)}\left[\ln\left(\frac{2m_ec^2}{I_{water}}\frac{\beta^2(E)}{1-\beta^2(E)}\right) - \beta^2(E)\right]} \\
&= \int_{E_{out}}^{E_{in}} \frac{dE}{K\frac{(E+E_p)^2}{E^2+2EE_p}\left[\ln(2m_ec^2) - \ln(I_{water}) + \ln(E+2E_p) + \ln(E) - 2\ln(E_p)\right] - K}
\end{aligned}
$$

which is the most simplified version of the equation relating energy loss to the path integral of the relative electron density.

2.5 High Speed (HS) Algorithm for Converting Energy Loss to Integrated Relative Electron Density

2.5.1 Overview

The High Speed (HS) Algorithm is been implemented to determine the physical density of a brain tissue using the Bethe-Bloch Equation 2.4. The first challenge faces

this algorithm was to calculate multiple integrations multiple times in one equation. The second challenge was to apply this equation on $180 \times 50 \times 10^3$ protons. This created a lot of challenge in time, performance, memory leak, and error boundaries.

Thanks to all my math instructors who taught me how to best simplify a math equation. My skills resulted the simplified version of the Bethe-Bloh Equation 2.13. My algorithm will output the integrated relative electron density of each proton. The input will be only the Entry Proton Energy and the Exit Proton Energy. To

The tremendous number of protons, on which I will do my calculations has lead us toward using MatLab environment, since it can handle huge number of variables and equations. I input the data file from the experiment data into a MatLab matrixes and feed it to my HS algorithm. Refer to Appendix A for information about matrix, and Appendix B for more information about Numerical Analysis.

To know the Integrated Relative Electron Density for every Proton we must apply a numerical computation including multiple use of quadrature functions object the result of the physical density starting from the energy loss of protons after traversing the image object I developed the following algorithm to embed the equation 2.13, which is:

$$\int_{u_{in}}^{u_{out}} \eta_e(u)dx \;=\; \int_{E_{out}}^{E_{in}} \frac{dE}{K\frac{(E+E_p)^2}{E^2+2EE_p}\left[\ln(\frac{2m_ec}{I(u)}+\ln(E+2E_p)+\ln(E)-2\ln(E_p)\right]-K}$$

2.5.2 Background Numeric Integration and Quadrature Function

In numerical analysis, numerical integration constitutes a broad family of algorithms for calculating the numerical value of a definite *integral*, and by extension, the term is also sometimes used to describe the numerical solution of differential equations. The

term **quadrature** is more or less a synonym for *numerical integration*, especially as applied to one-dimensional integrals [**?**]. Two- and higher-dimensional integration is sometimes described as cubature, although the meaning of quadrature is understood for higher dimensional integration as well [**?**].

The most basic problem to be solved using numerical integration is to compute the approximate solution of a definite integral:
$\int_a^b f(x)dx$. For more information about numeric integration visit Appendix B.

Why Numeric Integration? Why Not?

Several facts and reasons affect scientist to use numeric integrations. For example, the integrand function f, from the previous equation, may be known only at certain points, such as obtained by sampling. Many computer applications and computer embedded systems often use numerical integration for such a reason.

The absente of finding an antiderivative is also another reason of using Numeric Integration. A formula for the integrand may be known, but it may be difficult or impossible to find an antiderivative. example $\exp(t^{-2})$.

2.5.3 Description of the Algorithm

Look at the algorithm code 2.5.3. This algorithm carefully translate the Bethe-Bloch Equation 2.13 blocks into lines of code which run smoothly to successfully meets our goal.

The second challenge, after translating the Bethe-Bloch Equation into lines of code, is to apply numerical integration Matlab functions on this algorithm(or I am going

to call it the HS function). It may be possible to find an antiderivative symbolically, but it may be easier to compute a numerical approximation than to compute the antiderivative. That may be the case if the antiderivative is given as an infinite series or product, or if its evaluation requires a special function which is not available.

Quadrature is a numerical method used to find the area under the graph of a function, that is, to compute a definite integral. In MatLab, there is a build in numerical integration functions that helps do the numerical integration. Those functions are called Quadrature Functions. **Quad** and **Quadl** are MatLab Build-in functions, I used both functions, as I write code for **Gaussian Quad** function [3].

After developing the Algorithm above and translate it into MatLab code, I applied the function on a different numerical integration methods, and compare the results with NIST [**?**] results.

2.6 Comparison of Different Numerical Integration Methods

2.6.1 Overview

As I described above, I had run my experiment with different numerical integration functions, one of which I had to create myself.

I have to check for correctness of my function performance against the other functions. Also I have to check my function and the other MatLab functions correctness against the NIST National Institute of Science and Technology data 2.4.

The first comparison was against time performance between the three functions, the second was for accuracy. I, as each of my committee members, were very sat-

[3] Refer to Appendix B if more information about numeric Integration are needed.

functionoutput=**HS**(E)

$e_p = 938.295; \%MeV$

$K = 0.170; \%MeV.cm^{-2}$

$mec2 = 511.011 * 10^{-}3;$ *%KeV therefore we multiply by three 3*

$I = 75 * 10^{-}6;$ *%eV therefore we multiply by six 6*

$I = 91.90000010^{-}6;$ *% this is for bone.*

$EEP2 = (E + e_p). * (E + e_p);$

$E22EP = E. * E + (2 * e_p). * E;$

$div01 = EEP2./E22EP; div03 = E22EP./EEP2; div02 = 2 * mec2/I;$

$ln_mec2i = log(div02);$

$output = K. * div01. * (ln_mec2i + log(E) + log(E + 2 * e_p) - 2 * log(e_p) - div03);$

$output = 1./output;$

Fig. 2.3: High Speed Algorithm for Converting Energy Loss to Integrated Relative Electron Density

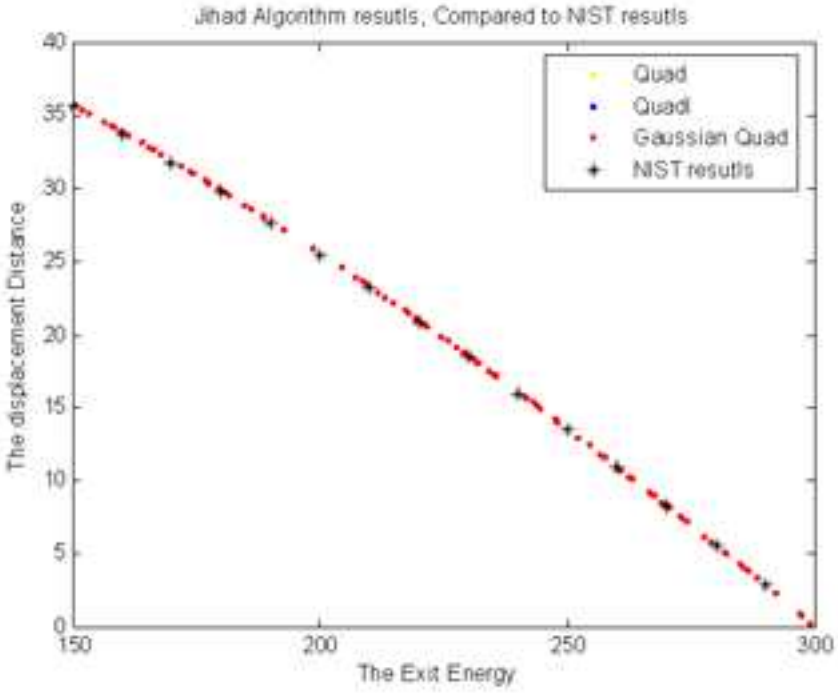

Fig. 2.4: Comparison of Different Numerical Integration Methods with NIST database

isfied with the results of my home-constructed gaussian quad function. I will be going quickly over NIST database [4], and then I will write in details about my performance and accuracy comparison, then I concluded this section with a summary of this chapter.

2.6.2 Comparison of Computer Run Time

One o the first main reasons to do this thesis is to shorten the time needed to render 5^6 protons.

We are running this algorithm on millions of protons for one hundred and eighty time; therefore, every clock cycle we can save will be a plus.

The first run of my HS algorithm shows that the more Protons we render the more time takes us to do the calculation. After weeks of rendering the algorithm for many trials on different numbers of protons, I resulted that time will increase exponintially when applying the algorithm on greater number of Protons. Refer to chart 2.5 for more details.

After developing many versions of my HS algorithm, and implementing an excellent gaussian quad function code. the results highlighted that my home-made gaussian quad algorithm has great performance comparing to the regular build-in math lab functions.

The following chart followed by its table data will support my argument above.

From the charts 2.6, and in more detail the chart 2.7, and by looking at the logarithmic scale of the Gaussian quad, we can successfully adopt this home-made gaussian quad function. because it save us a lot of time especially we are not running

[4] Refer to Appendix C for more information about NIST Database.

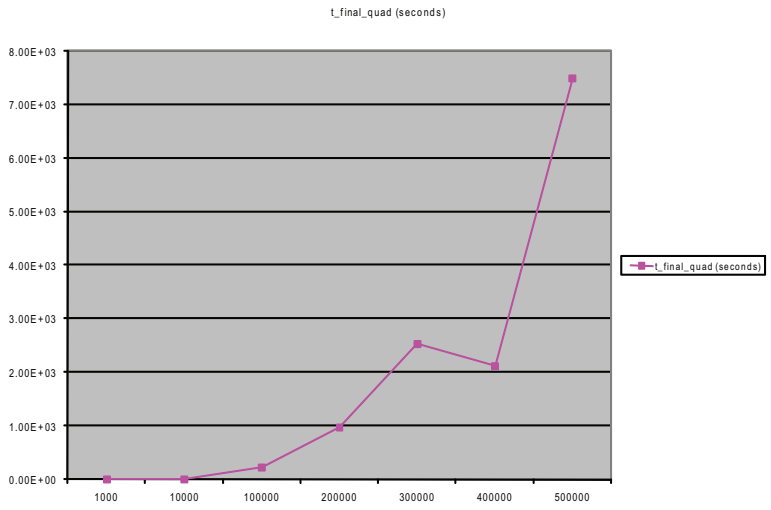

Fig. 2.5: Rendering Up to 5×10^5 Protons Verses Time in Seconds

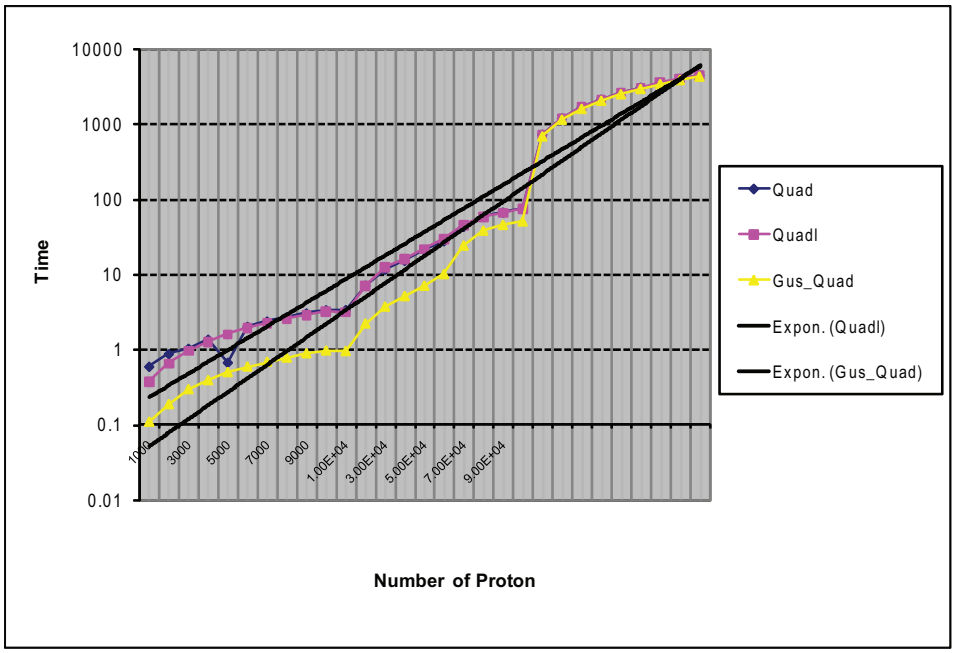

Fig. 2.6: Logarithmic Scale of Number of Proton Verses Time.

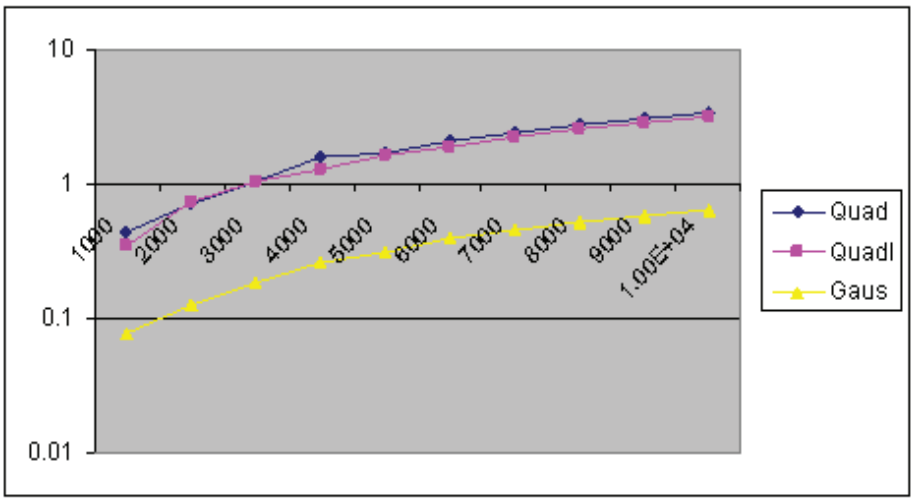

Fig. 2.7: In Detailed Logarithmic Scale of Number of Proton Verses Time

more than thirty five thousand (35000)protons at the same time. For almost 3^{10} protons quad needs 1.031 seconds, while quadl needs 0.96 seconds, and our home-made gaussian quad needs only 0.29 of a second.

Here is a full table of time VS. the three functions.

2.6.3 Comparison of Numerical Accuracy

After running my test for weeks, I resulted that the Numerical Accuracy of my HS algorithm very satisfiable. Comparing the results of the HS algorithm results, applied on MatLab functions (Quad, and Quadl), and my custom-build Gaussian Quad function, with the NIST pStar Database, and by looking at average error, max error, min error, and standard deviation, I found the error not to exceed %0.14. The following is a table with supported example of my comparison.

The first test was to compare HS Algorithm result with the NIST results. I tried

run—E	# E	quad	quadl	gu_el
1	1000	0.5938	0.375	0.1094
2	2000	0.875	0.6563	0.1875
3	3000	1.031	0.9688	0.2969
4	4000	1.359	1.281	0.3906
5	5000	0.672	1.609	0.5000
6	6000	2.016	1.938	0.5938
7	7000	2.375	2.234	0.6875
8	8000	2.703	2.563	0.7813
9	9000	3.047	2.875	0.8906
10	1.00E+04	3.359	3.203	0.9688
11	1.00E+04	3.375	3.156	0.9531
12	2.00E+04	7.125	7.031	2.2190
13	3.00E+04	11.64	12.48	3.7190
14	4.00E+04	15.28	16.33	5.1250
15	5.00E+04	21.11	22	7.0310

Tab. 2.1: Time Needed by Each Function to Render a Number of Protons

run—E	# E	quad	quadl	gu_el
16	6.00E+04	27.83	29.48	10.0500
17	7.00E+04	43.33	45.22	24.0800
18	8.00E+04	58.91	58.56	38.0900
19	9.00E+04	67.03	66.52	45.3800
20	1.00E+05	75.06	74.56	50.4100
21	2.00E+05	729.5	732.5	680.8000
22	3.00E+05	1201	1208	1136.0000
23	4.00E+05	1675	1680	1582.0000
24	5.00E+05	2136	2143	2027.0000
25	6.00E+05	2610	2615	2474.0000
26	7.00E+05	3076	3079	2916.0000
27	8.00E+05	3548	3557	3372.0000
28	9.00E+05	4031	4032	3819.0000
29	1.00E+06	4491	4488	4259.0000

Tab. 2.2: Continued... Time Needed by Each Function to Render a Number of Protons

	Min	Max	Standard Dev.	Average
Gaussian	0.01148834198939	0.13949398745925	0.03940854915190	0.07856807529633
Quad	0.01148834127241	0.13845100255350	0.03916737322183	0.07837578341375
Quadl	0.01148834127241	0.13845100649438	0.03916737525542	0.07837578528167

Tab. 2.3: Error Results Analysis between Quad Functions and NIST pSTAR Database

HS algorithm in Quad, Quadl, and Gaussian Quad with 100 - one hundred random numbers - using random algorithm technique. Here is my results:

The different quadrature functions were applied on the same HS Algorithm to determine the physical density from energy lost of proton traversing the image object. The following chart 2.8 and the above table 2.6.3 show very acceptable results of the used algorithm, as the error range $err > .01mm$ was acceptable from the thesis committee members.

2.6.4 Results

I am adopting my HS Algorithm since it is giving me acceptable results when comparing it to the NIST pSTAR [5] Database, along with my home-made Gaussian Quad Algorithm. The result were presented to the Thesis Committee Members, and has there satisfaction and approval.

2.7 Summary and Conclusion

The performance that I gained from building my home-made Gaussian Quad function, supported by the correctness of successful implementation of my HS Algorithm, gave

[5] Refer to Appended C for more Information about NIST Database.

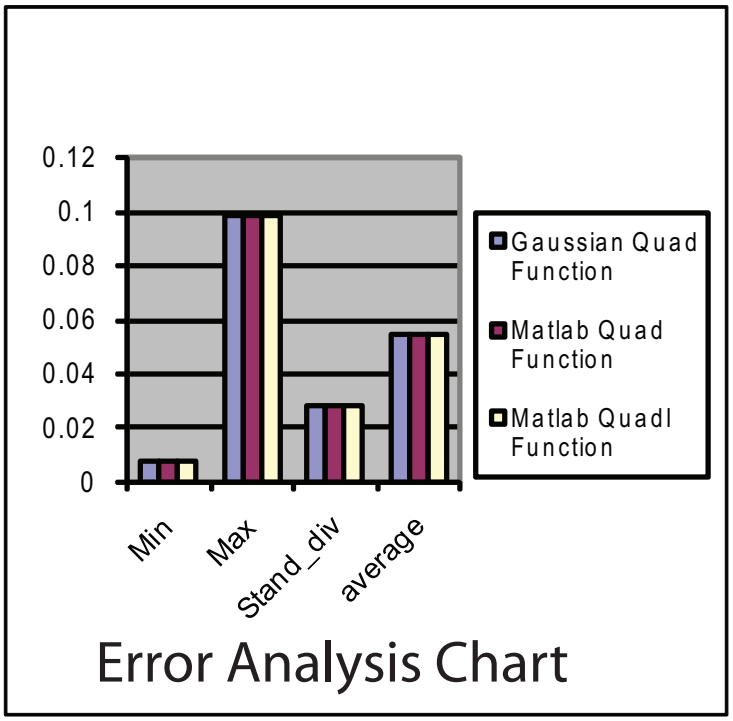

Fig. 2.8: Error Analysis Chart.

hope for Image Reconstruction using Proton Computed Tomography.

The proofed simplified version of the Beth-Bloche Equation, and the successful implementation of the HS algorithm will have huge impact on the future of Image reconstruction using pCT.

Graphic User Interface will be great improvement to the usability of such algorithm. Currently, I am feeding my data using the MatLab Environment, but GUI interface, implemented in C++ for example, will be great advantage to this application.

Since each proton is a separate entities; therefore, we can apply our calculations on each proton individually. Currently, I am only using my 3^{rd} victim laptop, after burning two computers, to do all my calculation on all the protons. Future continuing of the project is to use parallel programming applied on distributed system.

3. MOST LIKELY PATH DERIVATION

3.1 Introduction and Background

One of the challenges of pCT is the tendency for protons to undergoe scattering in the object by a process called a multiple Coulomb scattering (MCS). The uncertainty in the exact path of the proton leads to blurring of the image. One can minimize this spatial uncertainty of proton tracks by measuring the trajectory of individual incoming and outgoing protons using modern particle detector technology (Kleinknecht 1998) [**?**].

Particle detectors can measure the trajectory of a proton before entering and after leaving the object with better than 0.1 mm accuracy and precision. However, no direct information is available while the proton is traveling within the object. Therefore, some type of extrapolation of the external trajectories is required for optimal spatial resolution in pCT imaging. The best way to do this is to calculate the most likely path (MLP) of each proton along with its probability envelope using all available information. This chapter presents the theory of the MLP derivation and derives a closed-form expression for the MLP when the entrance and exit trajectories are known. It differs from previous derivations (Schneider and Pedroni 1994, Williams 2004), in that a compact matrix notation will be introduced.

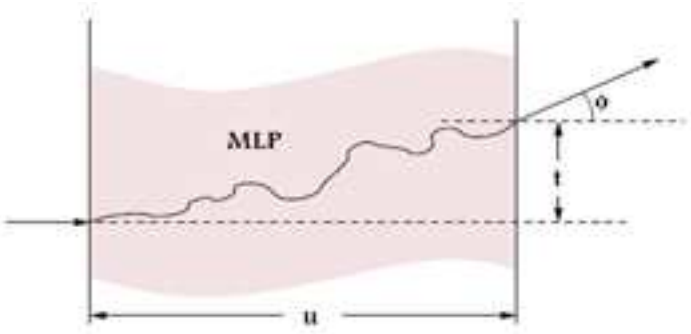

Fig. 3.1: Representation of Most Likely Path MLP

3.2 Multiple-Coulomb Scattering in the Gaussian Approximation

MCS is a physical process that leads to a statistical (or random) change of the direction of charged particles as they cross matter without changing their energy and velocity. Such scattering events are called "elastic". Most high-energy physicists are familiar with this process since it is often the limiting factor in the spatial resolution of charged-particle detectors. A summary of this process can be found in the Review of Particle Physics from the Particle Data Group (Yau 2006) [**?**]. The most relevant features of MCS are also described in Williams' paper [**?**].

When a proton traverses the object, many individual elastic interactions with the nuclei of the object material take place. The outcome of each individual nuclear interaction is a sample from a complicated statistical distribution of scattering angles governed by the laws of quantum mechanics. However, after undergoing many of these interactions, the combined result of the angular and spatial deviation from the initial trajectory is a probability density distribution that is approximately Gaussian or

normal. Therefore, a Gaussian approximation of the lateral and angular displacement at any given depth will be assumed in what follows. In this approximation, the we only need to know the variances and covariances of the scattering variables at any penetration depth in order to fully describe the distribution [**?**].

3.3 Derivation of the Matrix Form of the Most Likely Path

A closed analytical form of the 2D-projected MLP for protons traversing a homogeneous medium when their entry and exit positions and angles are known can be found in the work of Schneider et al [**?**] and Williams [**?**]. Here, we will derive a closed analytical form of the MLP using a compact matrix notation, which is advantageous considering the lengthy equations of the previous works. The scattering of a proton in the object can be described by the lateral displacement and the angle relative to the initial position and displacement at the entry into the object, i.e., by the two-dimensional vector function

$$y(u) \quad = \quad \begin{pmatrix} \theta(u) \\ \\ t(u) \end{pmatrix} \tag{3.1}$$

where: $t(u)$ is the lateral displacement and $\theta(u)$ the angle relative to the initial position and direction of the proton at depth u. At the boundaries $u = 0$ and $u = u_2$ of the object, $y(u)$ approaches the values

$$y(0) \quad = \quad \begin{pmatrix} 0 \\ \\ 0 \end{pmatrix} \tag{3.2}$$

and

$$y(u_2) \equiv y_2 \quad = \quad \begin{pmatrix} \theta(u_2) \\ t(u_2) \end{pmatrix} \equiv \begin{pmatrix} \theta_2 \\ t_2 \end{pmatrix} \tag{3.3}$$

Note that $t(u)$ and $\theta(u)$ are statistical variables, which acquire increasing spread with increasing depth. Also, these variables are not independent from each other, and therefore have a covariance that is different from zero. The amount of lateral and angular spread, t_1 and θ_1, accumulated at an intermediate depth u_1 in the object and the covariance of these quantities can be described by the first variance-covariance matrix

$$\Sigma_1 \quad = \quad \begin{pmatrix} \sigma^2_{\theta_1} & \sigma^2_{t_1\theta_1} \\ \sigma^2_{t_1\theta_1} & \sigma^2_{t_1} \end{pmatrix} \tag{3.4}$$

Explicit expressions for the matrix elements will be derived later. In the Gaussian approximation of small-angle Coulomb scattering [3], the probability density function of y at depth u_1,

$$y_1 \quad = \quad \begin{pmatrix} \theta(u_1) \\ t(u_1) \end{pmatrix} \equiv \begin{pmatrix} \theta_1 \\ t_1 \end{pmatrix} \tag{3.5}$$

is given by the bivariate Gaussian probability density function [**?**]

$$f_1(y_1) \quad = \quad k_1 e^{-\frac{1}{2} y_1^T \Sigma_1^{-1} y_1} \equiv k_1 e^{-\chi_1^2} \tag{3.6}$$

where k_1 is a constant needed for normalization. Since this probability density is based on the knowledge of the proton prior to entering the object, we may call it "prior probability density".

Next we are interested in the conditional probability density function of y_2 at the exit depth u_2 given y_1 at intermediate depth u_1. Starting from the intermediate depth

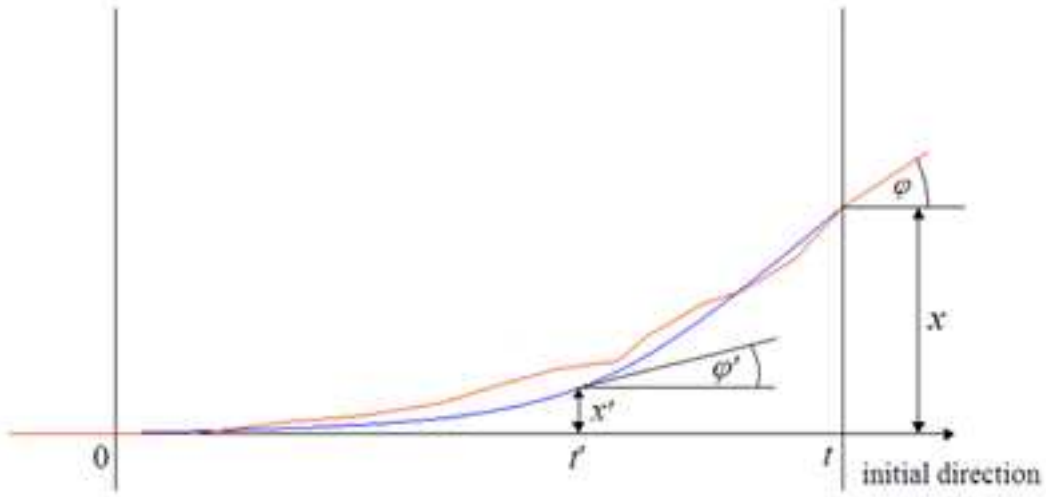

Fig. 3.2: Mathematical Relationship between Proton Scattering Matrix Penetration Depth

u_1, the proton will be further scattered between the depth u_1 and the exit depth u_2, which can be described by the second variance-covariance matrix

$$\Sigma_2 = \begin{pmatrix} \sigma^2_{\theta_2} & \sigma^2_{t_2\theta_2} \\ \sigma^2_{t_2\theta_2} & \sigma^2_{t_2} \end{pmatrix} \tag{3.7}$$

The probability density function, illustrated in figure[3.2 of y_2 is best described by changing the local coordinate system to the location and orientation of the proton path at depth u_1. This requires the following coordinate transformation:

$$\begin{pmatrix} u_2^{'} \\ t_2^{'} \end{pmatrix} = \begin{pmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{pmatrix} \begin{pmatrix} u_2 - u_1 \\ t_2 - t_1 \end{pmatrix} \tag{3.8}$$

from which we get

$$u_2^{'} = \cos\theta_1(u_2 - u_1) + \sin\theta_1(t_2 - t_1) \tag{3.9}$$

$$t_2^{'} = -\sin\theta_1(u_2 - u_1) + \cos\theta_1(t_2 - t_1) \tag{3.10}$$

In addition, the exit angle needs to be expressed relative to the angle θ_1 at depth u_1,

thus

$$\theta_2^{'} = \theta_2 - \theta_1^{'} \tag{3.11}$$

The conditional probability density function of of

$$y_2^{'} = \begin{pmatrix} \theta(u_2^{'}) \\ t(u_2^{'}) \end{pmatrix} \equiv \begin{pmatrix} \theta_2^{'} \\ t_2^{'} \end{pmatrix} \tag{3.12}$$

is given by the bivariate Gaussian probability density function

$$f_{21}(y_2^{'}|y_1) = k_2 e^{-\frac{1}{2} y_2^{'T} \Sigma_2^{-1} y_2^{'}} \equiv k_2 e^{-\chi_2^2} \tag{3.13}$$

The MLP and most likely angle are defined by the vector function

$$y_{mlp}(u) = \begin{pmatrix} \theta_{mlp}(u) \\ t_{mlp}(u) \end{pmatrix} \tag{3.14}$$

which maximizes the conditional probability density $f_{12}(y_1|y_2^{'})$ of y_1 given y_2 at any intermediate depth u_1 between 0 and u_2. This may be called the "posterior probability" because it uses information of the proton after exiting the object. An expression for the posterior probability can be found by using Bayes' theorem for continuous probability density functions [**?**],

$$f_{12}(y_1|y_2^{'}) = k_{12} f_1(y_1) f_{21}(y_2^{'}|y_1) \tag{3.15}$$

$$= k_{tot} e^{-\chi_1^2 - \chi_2^2} \equiv k_{tot} e^{-\chi^2} \tag{3.16}$$

where k_{12} is another normalization constant and k_{tot} is the product of normalization constants used in the different probability density functions.

Recall that

$$\chi^2 = \frac{1}{2} y_1(u)^T \Sigma_1^{-1} y_1(u) + \frac{1}{2} y_2^{'T} \Sigma_2^{-1} y_2^{'} \geq 0$$

To maximize $f_{12}(y_1|y_2^{'})$, we need to make χ^2 as small as possible. Therefore:

$$\left.\frac{\partial\chi^2}{\partial\theta}\right|_{\theta=\theta_{mlp}} = 0 \tag{3.17}$$

$$\left.\frac{\partial\chi^2}{\partial t}\right|_{t=t_{mlp}} = 0 \tag{3.18}$$

The solution θ_{mlp}, t_{mlp}, which satisfies this system of two equations are the most likely angle and MLP. The latter is what we need in particular for pCT reconstruction. Using vector notation and introducing the gradient vector we can also write:

$$\nabla\chi^2 = \begin{pmatrix}\partial\theta\\ \\ \partial t\end{pmatrix}\chi^2 = \begin{pmatrix}0\\ \\ 0\end{pmatrix} \tag{3.19}$$

In the following I will show two ways how to find the solution of this vector equation.

3.3.1 Small-Angle Approximation to Find the MLP Solution

When θ is smaller than a few degrees, which is the case for most MCS scattering events, then $\sin\theta \approx \theta$ and $\cos\theta \approx 1$. Further, because t is usually much smaller than u_1 and u_2 terms involving $t\theta$ can be ignored but terms involving $u_1\theta$ and $u_2\theta$ can not. This leads to the simplified equation

$$u_2^{'} = u_2 - u_1 \tag{3.20}$$

$$t_2^{'} = -\theta_1(u_2 - u_1) + (t_2 - t_1) \tag{3.21}$$

We may then use the following simplified notation

$$y_2^{'} = y_2 - Ry_1 \tag{3.22}$$

where

$$R = \begin{pmatrix}1 & u_2 - u_1\\ \\ 0 & 1\end{pmatrix} \tag{3.23}$$

With this notation, we can rewrite χ^2 as

$$\chi^2 = \frac{1}{2} y_1^T \Sigma_1^{-1} y_1 + \frac{1}{2}(y_2^T - y_1^T R^T)\Sigma_2^{-1}(y_2 - R y_1) \quad (3.24)$$

$$= \frac{1}{2}(y_1^T \Sigma_1^{-1} y_1 + y_2^T \Sigma_2^{-1} y_2 - 2 y_1^T R^T \Sigma_2^{-1} R y_1 + y_1^T R^T \Sigma_2^{-1} R y_1) \quad (3.25)$$

Carrying out the derivation with respect to y_1 results in

$$\nabla\chi^2 = \Sigma_1^{-1} y_1 + R^T \Sigma_2^{-1} R y_1 - R^T \Sigma_2^{-1} y_2 \quad (3.26)$$

Thus, the MLP equation becomes

$$0 = (\Sigma_1^{-1} + R^T \Sigma_2^{-1} R) y_{mlp} - R^T \Sigma_2^{-1} y_2 \quad (3.27)$$

which has the solution

$$y_{mlp} = (\Sigma_1^{-1} + R^T \Sigma_2^{-1} R)^{-1} R^T \Sigma_2^{-1} y_2 \quad (3.28)$$

3.3.2 Exact Solution of the MLP Problem

Using Numeric Analysis techniques and Math skills we can derive the exact solution to the MLP.

The following variables have known values for each proton:

- entry and exit energy E_0 and E_2;
- entry and exit lateral displacement t_0 and t_2;
- entry and exit depth u_0 and u_2;
- entry and exit angle in the u-t plane θ_0 and θ_2;
- entry and exit vertical displacement v_0 and v_2;

- entry and exit angle in the u-v plane ϑ_0 and ϑ_2.

The last two items will be used in future 3-D image reconstruction.

Given $y_1 = \begin{bmatrix} u_1 & t_1 & \theta_1 \end{bmatrix}^T$, which describes the parameter vector of the proton at depth u_1, and $y_2 = \begin{bmatrix} u_2 & t_2 & \theta_2 \end{bmatrix}^T$, which describes the known parameter vector at the exit location, the solution for y_1 that minimizes $\chi^2 = \chi_1^2 + \chi_2^2$, where:

$$\chi^2 = y_1^T \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_1^{-1} \end{bmatrix} y_1 + \tag{3.29}$$

$$y_2'^T \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_2^{-1} \end{bmatrix} y_2' \tag{3.30}$$

where

$$y_2' = \begin{bmatrix} Q_{\theta_1} & 0 \\ 0 & 1 \end{bmatrix} (y_2 - y_1) \tag{3.31}$$

and

$$Q_{\theta_1} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \tag{3.32}$$

will be given as:

$$y_{mlp} = -\begin{bmatrix} 0 & 0 \\ 0 & \Sigma_1^{-1} \end{bmatrix}^+ \frac{dy_2'}{dy_1} \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_2^{-1} \end{bmatrix} y_2' \tag{3.33}$$

where

$$\frac{dy_2'}{dy_1} = \begin{bmatrix} 0 & 0 & \begin{bmatrix} R_{\theta_1} & 0 \\ 0 & 0 \end{bmatrix} (y_2 - y_1) \end{bmatrix} - \begin{bmatrix} Q_{\theta_1} & 0 \\ 0 & 1 \end{bmatrix} \tag{3.34}$$

and

$$R_{\theta_1} = \frac{dQ_{\theta_1}}{d\theta_1} \tag{3.35}$$

$$= \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 \\ -\cos\theta_1 & -\sin\theta_1 \end{bmatrix} \tag{3.36}$$

3.3.3 Mathematical Relationship between Proton Scattering Matrix Elements and Penetration Depth

As we have seen above, the path of an individual proton through the object and projected into the u-t plane can be described by the two parameters θ and t as a function of the depth of penetration u. The lateral and angular displacements are statistically correlated, and, therefore, have non-zero covariance. In the Gaussian approximation of small-angle MCS [**?**], the joint probability density function of the vector

$$y_1 = \begin{pmatrix} \theta_1 \\ t_1 \end{pmatrix} \tag{3.37}$$

at depth u_1 is given by the bivariate Gaussian:

$$f_1(y_1) = k_1 e^{-\frac{1}{2} y_1^T \Sigma_1^{-1} y_1} \tag{3.38}$$

where Σ_1^{-1} is the inverse variance-covariance matrix of the θ_1 and t_1:

$$\Sigma_1 = \begin{pmatrix} \sigma_{\theta_1}^2 & \sigma_{t_1\theta_1}^2 \\ \sigma_{t_1\theta_1}^2 & \sigma_{t_1}^2 \end{pmatrix} \tag{3.39}$$

Similarly, the joint probability density function of the vector

$$y_2^{'} = \begin{pmatrix} \theta_2^{'} \\ t_1^{'} \end{pmatrix} \tag{3.40}$$

at depth u_2 given the vector y_1 at depth u_1 is given by the bivariate Gaussian:

$$f_{21}(y_2^{'}|y_1) = k_2 e^{-\frac{1}{2} y_2^{'T} \Sigma_2^{-1} y_2^{'}} \tag{3.41}$$

where Σ_2^{-1} is the inverse variance-covariance matrix of the $\theta_2^{'}$ and $t_2^{'}$:

$$\Sigma_2 = \begin{pmatrix} \sigma_{\theta_2}^2 & \sigma_{t_2\theta_2}^2 \\ \sigma_{t_2\theta_2}^2 & \sigma_{t_2}^2 \end{pmatrix} \tag{3.42}$$

Since we are dealing with relatively thick objects in pCT, one needs to take energy loss of the proton inside the object into account. Then, the individual variance and covariance elements of the matrices Σ_1 and Σ_2 can be expressed by the following integrals [**?**]:

$$\sigma_{t_1}^2(u_1) = \Theta_0^2 \int_0^{u_1} \frac{(u_1-u)^2}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.43}$$

$$\sigma_{\theta_1}^2(u_1) = \Theta_0^2 \int_0^{u_1} \frac{1}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.44}$$

$$\sigma_{t_1\theta_1}^2(u_1) = \Theta_0^2 \int_0^{u_1} \frac{u_1-u}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.45}$$

$$\sigma_{t_2}^2(u_1,u_2) = \Theta_0^2 \int_{u_1}^{u_2} \frac{(u_1-u)^2}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.46}$$

$$\sigma_{\theta_2}^2(u_1,u_2) = \Theta_0^2 \int_{u_1}^{u_2} \frac{1}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.47}$$

$$\sigma_{t_2\theta_2}^2(u_1,u_2) = -\Theta_0^2 \int_{u_1}^{u_2} \frac{u_1-u}{\beta^2(u)p^2(u)} \frac{du}{X_0} \tag{3.48}$$

where the terms $\beta^2(u)$, $p^2(u)$ are the squared velocity relative to the speed of light and momentum of the proton at depth u, respectively, and $\Theta_0 \cong 13.6$ MeV/c is a physical constant. The quantity X_0 is the radiation length, which is a physical constant for

a given material. Throughout this thesis, we will assume that the scattering object consists of water, for which $X_0 = 36.08$ cm.

The integrals in ***** the last six equations ***** have to be calculated numerically. The product $\beta^2(u)p^2(u)$ for protons is related to the proton energy by the relativistic formula:

$$\beta^2(u)p^2(u) = \frac{(E(u) + 2E_p)^2 E^2(u)}{(E(u) + E_p)^2 c^2} \tag{3.49}$$

where $E_p = 938.295$ MeV s the proton rest energy.

The change of energy with penetration depth is governed by the energy loss of the proton, which, in case of water, is described by the differential Bethe Bloch equation:

$$-\frac{dE}{du}(u) = F(I, E(u)) \tag{3.50}$$

where I is the mean excitation proton of water ($I_{water} = 75$ eV) and the function $F(I, E)$ is defined as before. Starting with an initial energy E_0, the proton has acquired reduced energy $E(u_1)$ at depth u_1, which is described by the solution of the integral form of Bethe-Bloch equation:

$$u_1 = \int_{E(u_1)}^{E_0} \frac{1}{F(I, E(u))} du \tag{3.51}$$

Three ways to solve this equation for $E(u_1)$ were investigated in this thesis:

- solving the integral equation numerically.
- use of the NIST PSTAR Database program to find the energy corresponding to a penetration depth u;
- use of a Monte Carlo simulation program such as GEANT4 to and find the parameters of a polynomial relationship between energy and penetration depth.

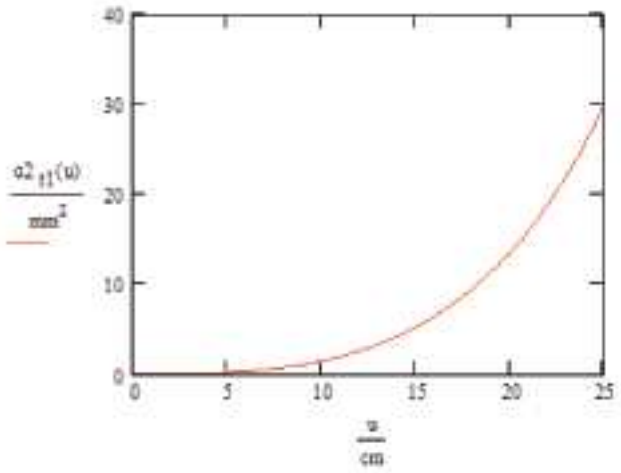

Fig. 3.3: Scattering Matrix Elements of $\Sigma_1 t^2$ as a Function of Depth

Numerical Solution

One possibility to solve the integral equation for obtaining the energy-depth relationship is using the minerr function of the Mathcad software (Mathsoft, Inc., Cambridge, MA). This was initially used to calculate the matrix elements of the two scattering matrices The results for the elements of the matrix Σ_1 are shown graphically in fig 3.3, 3.4, and 3.5and are also tabulated in 0.1 cm intervals in lables 3.3-3.15 of the Appendix.

Later in this research, I have implemented my own function to calculate the residual energy as a function of depth u. For now, I am using the Interval Bisection method (see Matlab code below, [**?**]), since it has good time performance over the other methods. Future work should also test Newton's and other methods.

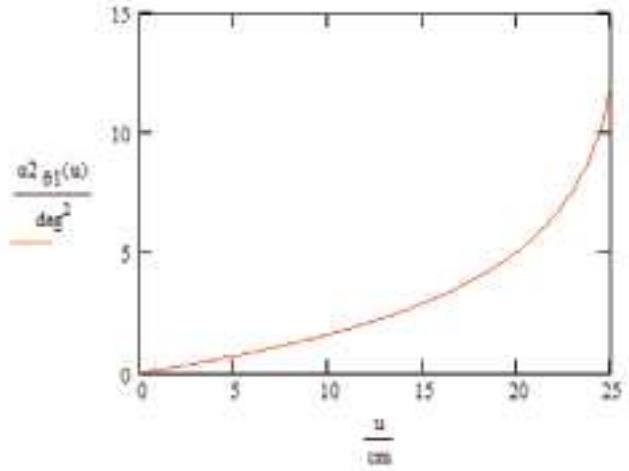

Fig. 3.4: Scattering Matrix Elements of $\Sigma_1\theta^2$ as a Function of Depth

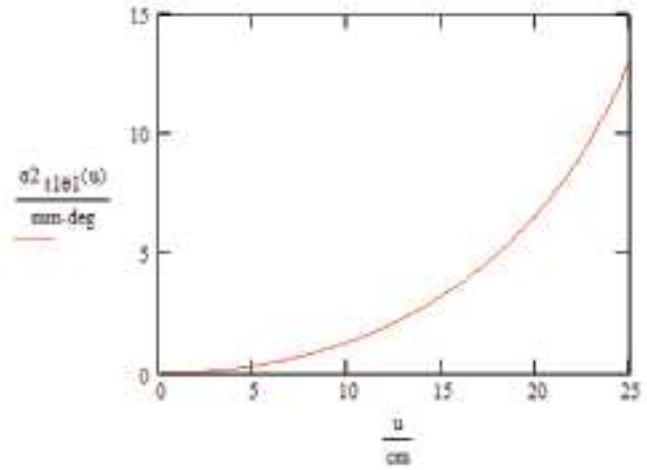

Fig. 3.5: Scattering Matrix Elements of $\Sigma_1\theta^2 t^2$ as a Function of Depth

National Institute of Standards and Technology Database Solution

NIST, or National Institute of Standards and Technology, has a database called pSTAR, which is the other possibility to obtain residual energy as function of depth is to utilize the data tabulated in the NIST PSTAR database [**?**]. I used this to verify the results of my numerical solution for the intial energy of 200 MeV, because the NIST database is based on a more complicated and therefore more accurate model than the Bethe Bloch equation. The NIST database lists the projected range of protons, i.e., the average value of the depth to which protons of a certain energy will penetrate in a material of choice, including water. The data for water for proton energies between 1 MeV and 200 MeV are shown in the first two columns of the following table.

From these data, I derived the relationship between residual proton energy and penetration depth starting from an initial energy of 200 MeV as follows. The initial proton energy of 200 MeV corresponds to a projected range of 25.93 cm in water as seen at the bottom of the table; at the next lower tabulated energy of 175 MeV, the range is reduced to 20.60 cm. From the range difference of 5.33 cm between 200 MeV protons and 175 MeV protons, one can conclude, in turn, that a proton of 200 MeV initial energy after penetrating depth of 5.33 cm in water has a residual energy of 175 MeV. The penetration depth derived from the difference between the projected range at the initial energy (here 200 MeV) and that at lower energies is included in the third column of Tables 3.1 and 3.2. Thus the third column of the table gives depth and the first column gives energy corresponding to that depth. Figure 3.6 shows the relationship between residual energy and penetration depth as derived from numerical

Kinetic Energy MeV	Projected Range cm	Depth cm	Kinetic Energy MeV	Projected Range cm	Depth cm
1	0	25.93	8	0.08	25.85
1.25	0	25.93	8.5	0.09	25.84
1.5	0	25.93	9	0.1	25.83
1.75	0.01	25.92	9.5	0.11	25.82
2	0.01	25.92	10	0.12	25.81
2.25	0.01	25.92	12.5	0.18	25.75
2.5	0.01	25.92	15	0.25	25.68
2.75	0.01	25.92	17.5	0.33	25.6
3	0.01	25.92	20	0.43	25.5
3.5	0.02	25.91	25	0.64	25.29
4	0.02	25.91	27.5	0.76	25.17
4.5	0.03	25.9	30	0.88	25.05
5	0.04	25.89	35	1.17	24.76
5.5	0.04	25.89	40	1.49	24.44
6	0.05	25.88	45	1.84	24.09
6.5	0.06	25.87	50	2.22	23.71
7	0.07	25.86	55	2.64	23.29
7.5	0.07	25.86	60	3.09	22.84

Tab. 3.1: Relationship between Initial Proton Energy and Projected Range in Water.

Kinetic Energy MeV	Projected Range cm	Depth cm	Kinetic Energy MeV	Projected Range cm	Depth cm
65	3.57	22.36	95	7.04	18.9
70	4.08	21.86	100	7.71	18.22
75	4.61	21.32	125	11.44	14.49
80	5.18	20.75	150	15.76	10.17
85	5.77	20.16	175	20.6	5.33
90	6.39	19.54	200	25.93	0

Tab. 3.2: Continue ... Relationship between Initial Proton Energy and Projected range in water.

solution using Mathcad's minerr function and from the NIST database values. The good agreement between both data sets confirms the numerical values.

The figure 3.6 illustrates energy-depth relationship for 200 MeV Protons in water. The points are derived From the NIST data in tables 3.1 and 3.2 and the line is derived by solving equation with Mathcad's minerr function.

3.3.4 GEANT4 Solution

Geant4 which is a toolkit for the simulation of the passage of particles through matter. Its areas of application include high energy, nuclear and accelerator physics, as well as studies in medical and space science. The two main reference papers for Geant4 are published in Nuclear Instruments and Methods in Physics Research A 506 (2003) 250-303, and IEEE Transactions on Nuclear Science 53 No.1 (2006) 270-278 [**?**].

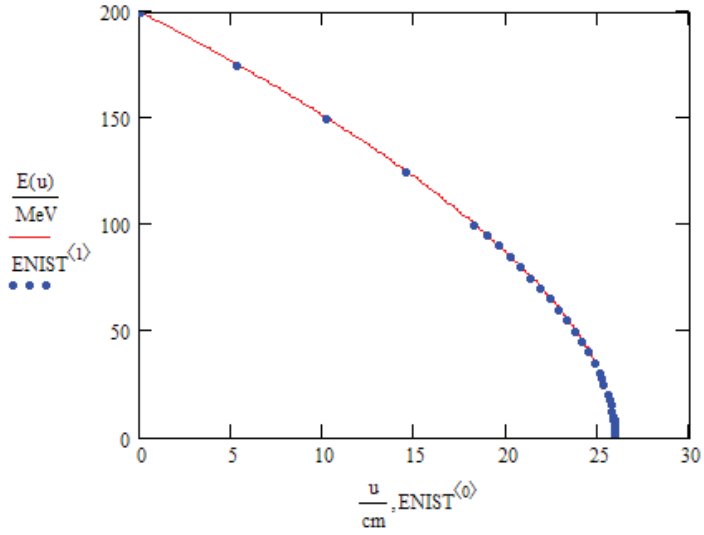

Fig. 3.6: Energy-depth Relationship for Protons in Water.

In order to facilitate the integration for the scattering matrix elements, [?] and [?] used a five-degree polynomial, $1/\beta^2 p^2 = \Sigma_{i=1}^{5} a_i u^i$, where the polynomial coefficients a_i were obtained by least-squares fitting the values obtained for a GEANT4 Monte Carlo simulation for 200-MeV protons traversing a uniform water phantom of 20 cm diameter. Thereby, these investigators avoided numerical integration altogether. For depths larger than 20 cm, the values for $1/\beta^2 p^2$ have to be extrapolated from this polynomial. The derived polynomial coefficients were $a_0 = 7.507 \times 10^{-4}$, $a_1 = 3.320 \times 10^{-5}$, $a_2 = -4.171 \times 10^{-7}$, $a_3 = 4.488 \times 10^{-7}$, $a_4 = -3.739 \times 10^{-8}$, and $a_5 = 1.455 \times 10^{-9}$, where the units are c^2/MeV divided by powers of cm according to the power of u. Figure 3.7 compares the relationship between $1/\beta^2 p^2$ and the penetration depth for the present calculation and those in [?]. The two results agree well up to a depth of 20 cm, when they start diverging. This difference can be explained by the fact that the polynomial fit was only for data up to a depth of 20

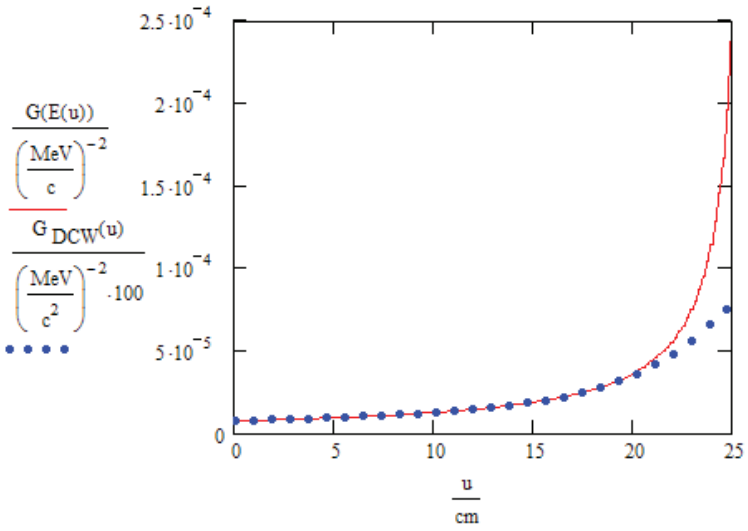

Fig. 3.7: Depth Dependence of the Product $1/\beta^2 p^2$ for a Polynomial Approximation and the Present Calculation

cm.

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
0	0	0	0
0.1	1.28E-06	0.012641	0.00011
0.2	1.03E-05	0.025334	0.000442
0.3	3.47E-05	0.03808	0.000995
0.4	8.23E-05	0.05088	0.001771
0.5	0.000161	0.06376	0.002771
0.6	0.000278	0.076667	0.003997
0.7	0.000443	0.089629	0.005448
0.8	0.000661	0.102646	0.007126
0.9	0.000943	0.11572	0.009031
1	0.001295	0.128848	0.011166
1.1	0.001725	0.142033	0.013529
1.2	0.002242	0.155273	0.016124
1.3	0.002853	0.168576	0.01895
1.4	0.003568	0.181932	0.022009
1.5	0.004392	0.195348	0.025301
1.6	0.005337	0.208819	0.028828
1.7	0.006408	0.222358	0.03259
1.8	0.007615	0.235952	0.03659
1.9	0.008965	0.249605	0.040828

Tab. 3.3: Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
2	0.010468	0.263321	0.045303
2.1	0.012131	0.277098	0.050019
2.2	0.013962	0.290936	0.054977
2.3	0.015972	0.304838	0.060175
2.4	0.018166	0.318755	0.065617
2.5	0.020555	0.33283	0.071304
2.6	0.023147	0.346923	0.077235
2.7	0.02595	0.361079	0.083414
2.8	0.028974	0.375304	0.089841
2.9	0.032225	0.389593	0.096515
3	0.035714	0.403944	0.103438
3.1	0.03945	0.418367	0.110616
3.2	0.043439	0.432842	0.118042
3.3	0.047694	0.447414	0.125725
3.4	0.05222	0.462041	0.133662
3.5	0.057028	0.476727	0.141856
3.6	0.062126	0.491503	0.150243
3.7	0.067524	0.506355	0.159012
3.8	0.073231	0.521249	0.167977
3.9	0.079253	0.536231	0.177204

Tab. 3.4: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
4	0.085605	0.551285	0.186698
4.1	0.092291	0.566414	0.196451
4.2	0.099322	0.581613	0.206468
4.3	0.106708	0.596891	0.216754
4.4	0.114459	0.612246	0.227306
4.5	0.122579	0.627677	0.238115
4.6	0.131086	0.643179	0.249216
4.7	0.139971	0.658766	0.260574
4.8	0.149279	0.674379	0.272211
4.9	0.158989	0.690168	0.284119
5	0.169119	0.70599	0.296303
5.1	0.179678	0.721892	0.308763
5.2	0.190676	0.737878	0.321344
5.3	0.202109	0.753946	0.334462
5.4	0.214034	0.770097	0.347819
5.5	0.226413	0.786334	0.361403
5.6	0.23927	0.802655	0.375271
5.7	0.252615	0.81906	0.389422
5.8	0.266594	0.835791	0.40386
5.9	0.280814	0.852271	0.418589

Tab. 3.5: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
6	0.295684	0.868806	0.433604
6.1	0.311087	0.885565	0.448916
6.2	0.32703	0.902415	0.46452
6.3	0.34352	0.919356	0.480416
6.4	0.360513	0.936359	0.496595
6.5	0.378331	0.953593	0.513101
6.6	0.396397	0.970558	0.529898
6.7	0.41519	0.988054	0.546985
6.8	0.434579	1.005279	0.564344
6.9	0.454596	1.022981	0.582085
7	0.475255	1.040548	0.600137
7.1	0.496459	1.0583	0.618385
7.2	0.518206	1.076111	0.637058
7.3	0.54098	1.094043	0.655971
7.4	0.564253	1.112007	0.675299
7.5	0.58811	1.130157	0.6948
7.6	0.612712	1.148381	0.714678
7.7	0.638005	1.16649	0.73488
7.8	0.664013	1.185155	0.755394
7.9	0.69071	1.203719	0.776209

Tab. 3.6: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
8	0.71822	1.222357	0.797422
8.1	0.746417	1.241131	0.818914
8.2	0.775406	1.260036	0.840689
8.3	0.805058	1.279001	0.862832
8.4	0.835639	1.298116	0.885394
8.5	0.866956	1.317353	0.908214
8.6	0.898977	1.336751	0.931288
8.7	0.93197	1.356156	0.954876
8.8	0.965719	1.375746	0.978718
8.9	1.000379	1.395436	1.002919
9	1.034917	1.415294	1.027437
9.1	1.072087	1.435236	1.052301
9.2	1.109202	1.455323	1.077507
9.3	1.147254	1.475546	1.103097
9.4	1.186304	1.495886	1.129031
9.5	1.226028	1.516361	1.155314
9.6	1.266881	1.536934	1.18197
9.7	1.308545	1.557504	1.208892
9.8	1.35176	1.578597	1.236314
9.9	1.394878	1.599621	1.264077

Tab. 3.7: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
10	1.439536	1.62078	1.292177
10.1	1.485143	1.642081	1.320659
10.2	1.531799	1.663459	1.349576
10.3	1.579358	1.685122	1.378711
10.4	1.628	1.706971	1.408319
10.5	1.677614	1.728747	1.438306
10.6	1.72854	1.750796	1.468768
10.7	1.780219	1.772991	1.499417
10.8	1.833099	1.795423	1.530561
10.9	1.88686	1.817938	1.56209
11	1.942161	1.840607	1.594019
11.1	1.998362	1.863423	1.626457
11.2	2.055712	1.886447	1.659069
11.3	2.11421	1.909602	1.69219
11.4	2.174035	1.932937	1.725715
11.5	2.234682	1.956448	1.759719
11.6	2.297506	1.98206	1.794621
11.7	2.359925	2.003984	1.828784
11.8	2.423783	2.028159	1.864264
11.9	2.489766	2.052242	1.899575

Tab. 3.8: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
12	2.557117	2.076646	1.935897
12.1	2.625194	2.101242	1.972627
12.2	2.694683	2.126032	2.00896
12.3	2.76564	2.150375	2.046281
12.4	2.837444	2.176174	2.083759
12.5	2.910995	2.201539	2.122406
12.6	2.985707	2.227165	2.160745
12.7	3.061885	2.2529	2.199542
12.8	3.139203	2.278837	2.239463
12.9	3.217986	2.304152	2.279005
13	3.298459	2.331578	2.319998
13.1	3.380564	2.3587	2.361136
13.2	3.463293	2.384998	2.402327
13.3	3.547816	2.411825	2.445261
13.4	3.634136	2.439184	2.48651
13.5	3.721638	2.467044	2.529393
13.6	3.809812	2.494073	2.572116
13.7	3.900891	2.522811	2.616363
13.8	3.993531	2.550933	2.663424
13.9	4.086931	2.579542	2.705456

Tab. 3.9: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
14	4.182014	2.608282	2.750705
14.1	4.278484	2.637323	2.79658
14.2	4.376949	2.666611	2.842764
14.3	4.4762	2.696262	2.889574
14.4	4.579149	2.725992	2.936908
14.5	4.682111	2.756287	2.984558
14.6	4.788226	2.786541	3.033539
14.7	4.893922	2.815181	3.081858
14.8	5.002484	2.848283	3.13132
14.9	5.112661	2.879215	3.181356
15	5.224732	2.910976	3.232538
15.1	5.338265	2.942794	3.282919
15.2	5.453902	2.97508	3.334671
15.3	5.571283	3.007537	3.387823
15.4	5.69036	3.040188	3.439661
15.5	5.811322	3.073528	3.49308
15.6	5.934047	3.107149	3.546721
15.7	6.059063	3.141083	3.601491
15.8	6.185633	3.17497	3.656683
15.9	6.314041	3.209827	3.712223

Tab. 3.10: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
16	6.444814	3.24473	3.770548
16.1	6.577284	3.280002	3.825567
16.2	6.711924	3.319998	3.882706
16.3	6.84846	3.352289	3.938815
16.4	6.987082	3.388197	4.000248
16.5	7.127758	3.425224	4.058986
16.6	7.269588	3.462337	4.119244
16.7	7.415441	3.500367	4.180422
16.8	7.56404	3.537464	4.241884
16.9	7.708393	3.57755	4.304191
17	7.862757	3.616373	4.366794
17.1	8.01647	3.654884	4.43031
17.2	8.172165	3.695681	4.494706
17.3	8.328794	3.736403	4.559321
17.4	8.490444	3.777592	4.624824
17.5	8.654897	3.818988	4.691157
17.6	8.817969	3.86126	4.758148
17.7	8.986203	3.904005	4.82695
17.8	9.153198	3.946948	4.894542
17.9	9.32664	3.990446	4.963648

Tab. 3.11: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
18	9.500945	4.03498	5.033681
18.1	9.67926	4.080137	5.104356
18.2	9.857957	4.125707	5.175818
18.3	10.04048	4.171975	5.248617
18.4	10.22422	4.219528	5.321795
18.5	10.41048	4.265722	5.395191
18.6	10.59601	4.313981	5.470855
18.7	10.79322	4.363355	5.546332
18.8	10.9865	4.412959	5.623007
18.9	11.18603	4.463292	5.700584
19	11.38584	4.51438	5.779123
19.1	11.58895	4.56619	5.858003
19.2	11.79507	4.619031	5.938164
19.3	12.00391	4.67262	6.019165
19.4	12.21513	4.72684	6.101032
19.5	12.43049	4.781982	6.184111
19.6	12.64653	4.838593	6.268461
19.7	12.86773	4.895688	6.353101
19.8	13.09046	4.954129	6.439677
19.9	13.317	5.013147	6.527303

Tab. 3.12: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
20	13.54627	5.073395	6.614662
20.1	13.77746	5.135851	6.70267
20.2	14.01464	5.197655	6.79375
20.3	14.24694	5.262294	6.884907
20.4	14.49515	5.326137	6.976739
20.5	14.74057	5.393588	7.070963
20.6	14.9873	5.458995	7.165559
20.7	15.24026	5.529707	7.261757
20.8	15.4939	5.598903	7.359513
20.9	15.75407	5.668858	7.456739
21	16.01614	5.74103	7.557176
21.1	16.28433	5.818029	7.658071
21.2	16.55071	5.895899	7.759371
21.3	16.82337	5.973428	7.863933
21.4	17.09905	6.053269	7.967996
21.5	17.37894	6.135502	8.074264
21.6	17.66343	6.220639	8.182023
21.7	17.95417	6.305674	8.295543
21.8	18.24132	6.392631	8.401447
21.9	18.53333	6.483033	8.51333

Tab. 3.13: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
22	18.83731	6.574559	8.628469
22.1	19.1513	6.671115	8.744384
22.2	19.45268	6.769341	8.861356
22.3	19.75654	6.87042	8.975476
22.4	20.07419	6.972037	9.104156
22.5	20.3939	7.080605	9.21893
22.6	20.71788	7.190889	9.348356
22.7	21.048	7.309589	9.477093
22.8	21.38037	7.41792	9.603357
22.9	21.71723	7.542977	9.733464
23	22.05992	7.669872	9.861692
23.1	22.40826	7.800529	10.00174
23.2	22.76102	7.937617	10.14393
23.3	23.11371	8.07578	10.28107
23.4	23.47424	8.223391	10.42099
23.5	23.8432	8.377318	10.56621
23.6	24.21907	8.535913	10.714
23.7	24.59234	8.700757	10.86385
23.8	24.97309	8.877659	11.02178
23.9	25.33077	9.067096	11.17668

Tab. 3.14: Continued ... Sigmas Verses Depth

Depth	s^2t^1	s^2q^1	$s^2t_1q_1$
cm	mm^2	deg^2	mmdeg
24	25.75127	9.255748	11.33259
24.1	26.14975	9.460161	11.49843
24.2	26.56196	9.675525	11.6639
24.3	26.96409	9.905364	11.8347
24.4	27.38303	10.15001	12.01593
24.5	27.80323	10.41203	12.17715
24.6	28.23052	10.69017	12.37328
24.7	28.66821	10.99825	12.56305
24.8	29.10374	11.32817	12.75761
24.9	29.55757	11.69821	12.96016
25	30.0012	12.09181	13.15106
25.1	30.4813	12.53968	13.38146
25.2	30.94884	13.04281	13.60422
25.3	31.42808	13.62588	13.83626
25.4	31.91643	14.299	14.08046
25.5	32.41103	15.12239	14.33837
25.6	32.91706	16.12973	14.6079
25.7	33.44924	17.52398	14.89933
25.8	33.95201	19.451	15.21766
25.9	34.50148	24.22958	15.59279

Tab. 3.15: Continued ... Sigmas Verses Depth

4. RECONSTRUCTION

4.1 Introduction and Background

The pCT reconstruction problem differs in some respects from that of xCT, PET and SPECT, and requires new approaches, although some of the underlying principles are the same [**?**]. In xCT, data collection is usually considered as the Radon transform of the object source function. In this case, the object data represent the attenuation coefficient map and the projection data the log values of the detected x-ray count [**?**]. The main goal of pCT for therapy application is the determination of the volume electron density, ρ_e, by measuring the energy loss of protons after traversing the object. Ionization and atomic excitation are the main processes for energy loss of protons as we have seen in the "Energy Loss Chapter" of this thesis.

In his paper [**?**] Dr. Schulte explain that in pCT, multiply scattered protons traversing the object travel along a curved zigzag path,which may deviate significantly from a straight line and is not confined to a 2D plane. Furthermore, protons usually do not get absorbed but traverse the object completely. thus, the proton counting rate used in x-ray CT, PET, and SPECT has to be replaced by the energy loss measurement for proton traveling along tracks : that lead to the same image pixel. Given the known proton entrance energy and the measured exit energy, the energy integral can be computed, resulting in the projection data. The image reconstruction

problem for pCT is then to obtain the best estimate for the relative electron density map from the measured proton data.

Using the Beth-Block Equation and the difference between the entry and exit energy for every proton, in the "Energy Loss chapter" we can get the projected traversity of a proton. On the other hand, and by using the same Beth-Block Equation and the entry and Exit Energy, and by knowing the entry and exit angle θ_0 and θ_2, we can calculate our sigmas Σ_1 and Σ_2 and then the MLP as that was covered in the "Most Likely Path Chapter". Now it is the time to introduce my third step of my research mission, which is reconstruct the image using the Path for each proton.

4.2 Steps Toward Solving the Reconstruction Problem

The following steps must be executed in order for us to reconstruct an image from a collected data. Some of the equations will be precalculated, while others will be calculated for each proton.

1. Calculate the line integral of relative electron density (must be calculated for each proton)
2. Pre-calculate sigma(s) for the object. (sigma will be precalculated one time only, and stored in a data file, and loaded in before calculating the object's data.)
3. Calculate the object boundary, using each proton in each data file.
4. Calculate the MLP for each proton at each angle ϕ of the rotation
5. Consider the rotational angle ϕ, and transfer the rotated coordinate into the original coordinate ϕ

6. Map the MLP into the path matrix map

7. Victorize the path map (for each proton)

8. Insert the path map vector into the sparse matrix

The solution for the first, second, and third step has been explained in the previous chapters of this thesis document. Let me explain the work done in each of the step remaining.

4.3 Identifying the Shape of The Object

Non in the previous research done any job related to identifying the shape of the object. Tianfang Li [**?**] assumes that the shape of the object is known in advance. In his experiment to calculate and reconstruct an image, he assumes that the outer shape and the coordinates of that shape is known too.

In my research I didn't assume that I know anything about the object or its coordinates in the experiment space. I just knew that currently I have a two dimensional (2D) square of 30_{cm} by 30_{cm}, and the object that we would like to reconstruct relay somewhere in this space.

The technique, which I came up with, to get the object's boundary is very simple. My technique depends on the energy loss for the protons, which travels through the space. Most likely that the outer protons in the top and the bottom of each measurement, therefore they will not hit the object; therefore, these protons will loss a very small amount of energy. This small amount of energy lost indecate that they most likely will travel in a straight line without hitting the object.

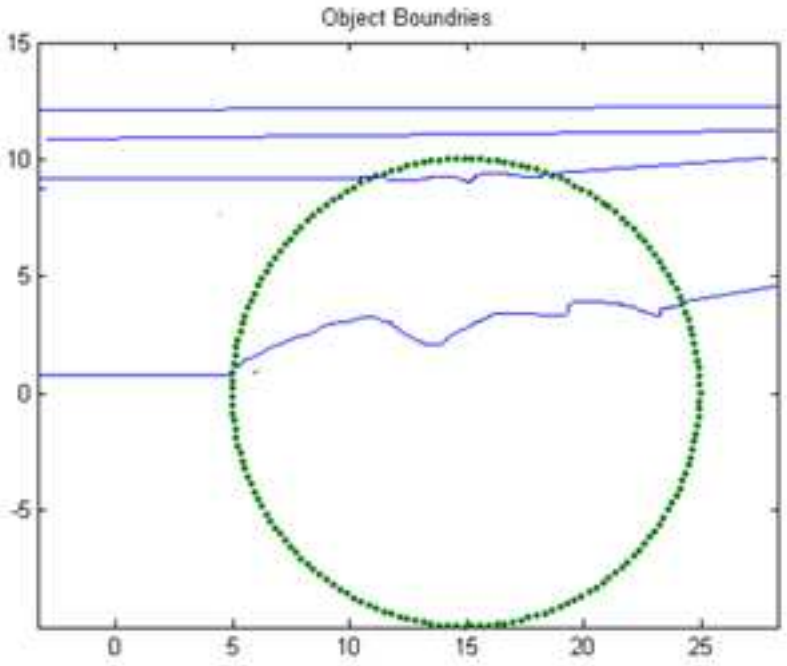

Fig. 4.1: Object Boundary Image Resulted from Proton Energy Loss

A proof for this fact will by analyzed by just looking at the entry energy and exit energy, the entry height and the exit height for each one of those protons. I stated a tolerance, if the proton loss more than 5_{MeV} (five) it means that this proton hit an object, so at the height of that proton there is an object. So by moving from the edges toward the center, I can calculate the last two point from the top and bottom where there is no object. I will mark those two points as they belong to the edge of that object.

Since I have 180 trajectories. I can get 360 points around the object. which will lead me to know the edges of that object. Look at figure 4.1 to see the result.

4.4 Calculating the MLP for Each Proton at Each Angle:

Since we know the object outer shape, I can calculate the MLP for that object by using the entry and exit height of that proton.

From the entry point at the level zero, I draw a straight line to the point where it will hit the surface of the object. On the other hand I will draw a straight line from the point, of which the proton will leave the object surface, to the to the exit point where it hit the sensors. Taking in consideration the entry and exit angle. Currently we assume that the entry angle is zero, and the exit angle is given in the data set.

4.5 Map the Most Likely Path(MLP) into the Path Matrix

The idea behind mapping the MLP is to isolate the most likely path for one proton and map that path on a pixelated map where if the proton visit the pixel then we get one (1), while it will mark the pixel with zero (0) otherwise. since we know the hight (t) at every depth (u) we will know the pixels, which the proton has visited at every depth.

4.6 Consider the Rotational Angle, And Transfer Back to The Original Coordinate

For the first read of the protons where $\phi = 0$ we don't have to rotate. The MLP that we got from the equation will be the MLP for that read. but how about the other group of protons where ϕ will equal an incremental number of 2.

Therefore before we apply our MLP to our reconstruction matrix, we must rotate our coordinates back to the original coordinates.

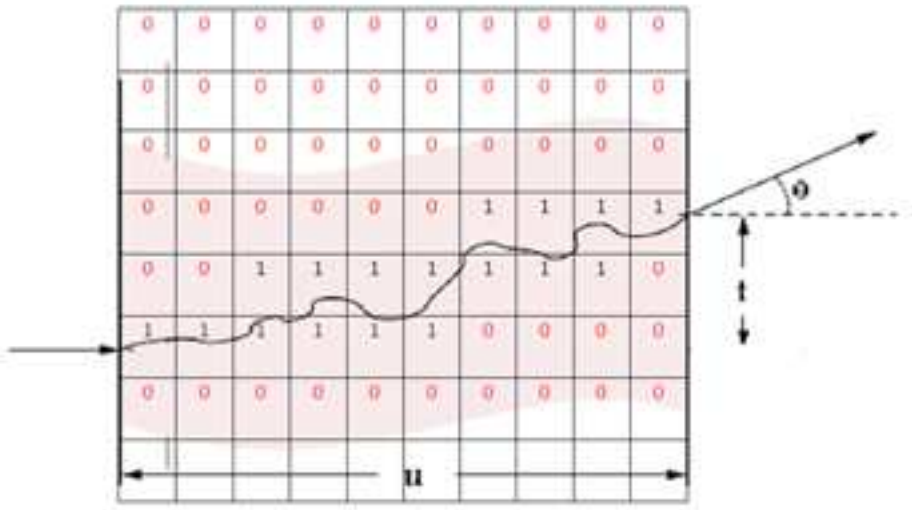

Fig. 4.2: A Pixelated Most Likely Path

Thanks to rotational matrix which will rotate our map to the original coordinates. As we will adopt the same when moving to three diminutions 3D objects.

4.7 Convert the Mapped MLP Matrix into a String of Zeros and Ones.

After having a map of zeros and ones for each protons. we will each map into a string vector of zeros and ones. Also then we will insert the resulted string vector into a bigger matrix which will hold all our MLPs.

I also have implemented an algorithm to covert the canvas into a vector. I have tested the algorithm against number of MLP and it worked.

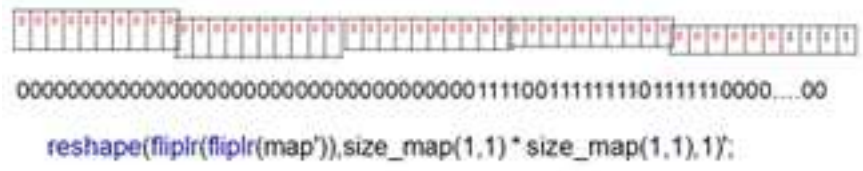

Fig. 4.3: A Victorized Most Likely Path with the Solving Algorithm

4.8 Image Reconstruction Matrix and Function

4.8.1 The Need for Sparse Matrix

As I mentioned before, I am dealing with a huge number of proton data sets; on which, I am applying many complex integrations.

Currently, I have one hundred and eighty (180) data files (for each angle), each includes seventeen registration for each on of the forty five thousand (45000) proton. Each proton will be mapped to three hundred by three hundred (300×300) matrix, which will be converted to a victor of zeros and ones of size ninety thousands (90000). Each on of those bits need eight bytes. Taking in consideration each proton, and ignoring the number of protons that register error information, we must have $180 \times 45000 \times 90000 \times 8$ which would equal to $5832000000000 bytes \simeq 5.3 Terabyte$ of Random access memory needed to hold this matrix.

Until now in the end of 2007, we don't have a machine which will hold that much of RAM to be allocated by this matrix, not to mention the RAM needed for the the operating system OS, other applications, this current application which will calculate the reconstruction and other matrixes needed for this application.

All of the above created the need to use sparse matrix, which reduces the size of

the protons' matrix from 5.3 TB to 10.x GB, which couldn't fit in the RAM of my current laptop or my desktop, but it will fit in a cluster or future laptop.

For this experiment I only used a sampled data set of the current data set.

4.8.2 The Need for Algebraic Reconstruction Technique

Algebraic Reconstruction Technique (ART) is a well known technique or method in solving sparse systems of linear equations. Thanks to Kaczmarz, this method is inherently sequential according to its mathematical definition since, at each step, the current iteration is projected toward one of the hyperplanes defined by the equations.

ART has many advantage in the world of solving sparse systems of linear equations. The main advantages of ART are its robustness, its cyclic convergence on inconsistent systems, and its relatively good initial convergence [**?**].

An entirely different approach for tomographic imaging consists of assuming that the cross section consists of an array of unknowns, and then setting up algebraic equations for the unknowns in terms of the measured projection data. Although conceptually this approach is much simpler than the transform-based methods discussed in previous sections, for medical applications it lacks the accuracy and the speed of implementation [**?**]. However, there are situations where it is not possible to measure a large number of projections, or the projections are not uniformly distributed over 180 or 360) both these conditions being necessary requirements for the transform based techniques to produce results with the accuracy desired in medical imaging [**?**].Problems of this type are sometimes more amenable to solution by algebraic techniques. Algebraic techniques are also useful when the energy propagation

paths between the source and receiver positions are subject to ray bending on account of refraction, or when the energy propagation undergoes attenuation along ray paths as in emission CT, which we have talked about in the first chapter of this thesis.

ART is widely used as an interactive solution to the problem of image reconstruction from projections in computerized tomography (CT), since it produces successful results when implemented with a small relaxation parameter produces.

In many ART implementations the path visit to cell c_{ik}, from the MLP, are simply replaced by ls and Os, depending upon whether the center of the k^{th} image cell is within the i^{th} row. This makes the implementation easier because such a decision can easily be made at computer run time. In this case the denominator in 4.1 The correction to the j^{th} cell from the i^{th} equation will be determined in 4.3 for all the cells whose centers are within the i^{th} row:

$$\triangle \int_j^i = \int_j^i - \int_j^{i-1} \tag{4.1}$$

$$\triangle \int_j^i = \frac{p_i - q_i}{\sum_{k=1}^{N} w_{ik}^2} w_{ij} \tag{4.2}$$

$$\triangle \int_j^i = \frac{p_i - q_i}{N_i} \tag{4.3}$$

ART reconstructions usually suffer from salt and pepper noise [**?**], which is caused by the inconsistencies introduced in the set of equations by the approximations commonly used for the proton position at each depth u.

Fig. 4.4: Simulated Object with Density Represented by High(Four), Medium(Two), and Low(Zero)

4.8.3 Use the Numerical Algebraic Reconstruction Technique $A = bx$ to Solve for the Reconstruction

For this thesis I only used numerical ART that is build in MatLab application. Due to the lake of memory, I am only using a small portion of my data set.I have tried my algorithms on this small data set and apply the resulted data to the MatLab build-in least square technique.

Illustrated in figure[4.4],My small data set consist of a simulated object with low density where it shows zeros 0s, and high density where it shows fours (4s), and medium density where it shows twos.

After calculating the MLP for number of proton travels through the object I got the matrix illustrated in figure[4.5].

Fig. 4.5: Sample of Protons' Vectorized Most Likely Path Constructed on a Matrix

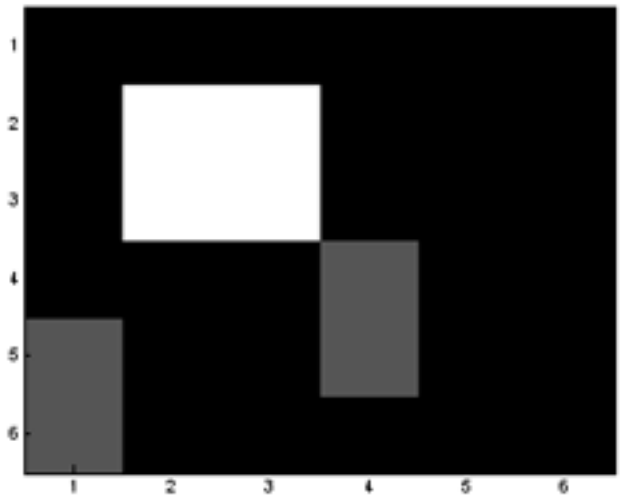

Fig. 4.6: Reconstructed Image of Sample Data Collection

This will result the final picture of the reconstructed sample of figure[4.4]. The picture in figure[4.6].

4.8.4 Results

Although I couldn't my target object from the given data set due to lake of memory, but I am very confident that each step in my reconstruction technique was carefully planned, strongly implemented, and successfully tested, and match the expected result when compaired to the NIST database result.

Fortunately, the home made functions, which I implemented through out the last two and a half year working on this research, has the support of my thesis committee members. With better performance, tt produced same result, and sometime better, which was produced by other build-in functions; but with better performance.

4.9 Summary of Future Work

The performance that I gained from building my home-made Gaussian Quad function, supported by the correctness of successful implementation of my HS algorithm, gave hope for Image Reconstruction using Proton Computed Tomography. However, better hardware must be used to implement this algorithm to handle Memory leak, which will be huge when upgrading to 3-D computed tomography.

Since each proton is a separate entities; we can apply our calculations on each proton individually. Currently, I am only using my 3^{rd} victim laptop, after burning two computers, to do all my calculation on all the protons. Future continuing of the project is to use parallel programming applied on distributed system.

Graphic User Interface will be great future improvement to the usability of such algorithm. Currently, I am feeding my data using the MatLab Environment. A future GUI interface, implemented in C++ for example, will be great advantage to this application.

Although not reached, but good continuing in this research is to implement different ART algorithm for image reconstruction and compare their performance.

4.10 Conclusion

I am honored to work on such a research, among few other challengers who didn't hesitate to take this challenge. It was big research to accomplish. I started with zero experience about biology, proton, X-RAY, pCT, MatLAB, LaTeX, thesis templates ... etc, but looking at myself now, I gained a lot of experience from this thesis, and I didn't, as I will never, gave up.

This project will be a good help for brain and body imaging. Especially in the field of tumors imaging. A lot has been done, and a lot more are waiting

Any challengers...

JT

Seattle, WA

$NOV 29^{th}, 200704 : 59 : 49 AM$

APPENDIX A

MATRIX

Looking around, we will analyze that everything can be digitalized into a matrix. Spreadsheet in Excel file, table or any other respectable data file that contain set of numbers related to each other. Matrices Make presentation of numbers clearer and make calculations easier to program.

Kay [**?**] define matrix as a rectangular array of elements, the elements can be symbolic expressions or numbers. For example Matrix [A] is denoted by:

$$[A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & ..... & a_{1n} \\ a_{21} & a_{22} & a_{23} & ..... & a_{2n} \\ . & & & & . \\ . & & & & . \\ . & & & & . \\ a_{m1} & a_{m2} & a_{m3} & ..... & a_{m4} \end{bmatrix} \tag{1.4}$$

We define **a row** i as it has n elements, which are $[a_{i1}a_{i3}a_{i3}.....a_{in}]$ and we define a **column** j as it has m elements, which are: $\begin{bmatrix} a_{1j} \\ a_{2j} \\ . \\ . \\ a_{mj} \end{bmatrix}$

The **size of a matrix** is the result of multiplying the number of rows m by the number of column n; it is donated by (m x n).

Element of matrix is donated by a_{ij}. The following is an matrix example:

$$[A] = \begin{bmatrix} 30 & 24 & 55 & 11 & 27 & 45 \\ 21 & 10 & 9 & 8 & 11 & 2 \\ 3 & 15 & 16 & 32 & 20 & 24 \\ 6 & 17 & 3 & 5 & 8 & 13 \end{bmatrix} \tag{1.5}$$

The above matrix [A] has a size of $4x6 = 24$. The elementa_{46} has a value of 13.

Note that for regular matrix we always write the name of a matrix in capital letters.

The proper study of matrix computation begins with the study of the matrix-matrix multiplication problem [**?**]. Although this problem is very simple mathematically, it is very rich from the computational point of view.

Matrix computation are build upon hierarchy of linear algebraic operations. Dot products involve the scalar operation of addition and multiplication. Matrix vector multiplication is mad up of dot products. Matrix-matrix multiplication amounts to a collection of matrix vector products [**?**]. All of these operation and more can be described in algorithmic from or in the language of linear algebra. If it was to me, I will ask Education Department to teach Matrix computation with the very First Mathematics Classes.

1.2 Definition of Matrix

There is obvious importance in adopting a methodical arrangement of equations and all such polynomial expressions, involving several variables *x,y,z* [**?**]. Also, because of the convenient fact that many of the properties of a square of oblong formation can be illustrated by arranging for or six things two by two in a square, or two by three

in a oblong, we can continue to extract useful general notations from our equations above (1.6). The set of coefficients:

$a_1 b_1 c_1$

$a_2 b_2 c_2$ of (1.6), arranged in their relative positions, is an example of a matrix of order two and three. **In definition** *a matrix of orders* m *and* n *simply means a set of* mn *numbers arranged in rectangular array with* m *rows and* n *columns* [**?**].

1.3 Special Types of Matrix

Vector: a vector is a matrix that include only one row or one column. This results two types of vector matrices, row matrices and vector matrices.

Row Vector Row vector is a matrix that has one row. As standard we always choose vector name to be capital letters. Example of a row matrix $[B] = [1123581321]$. We describe matrix [B]as a row vector of dimension 8.

Column Vector Column Vector is a matrix that include only one column. As a standard we also choose vector name to be capital letters. Example of a vector matrix

$$[C] = \begin{bmatrix} 8 \\ 13 \\ 21 \\ 34 \\ 55 \\ 89 \end{bmatrix}$$

We describe matrix [C] as a column vector with 6 rows.

Square Matrix We call a matrix a Square matrix if the number of the rows (m)

is equal to the number of columns (n) of the matrix (m=n). where the entries a_{11},a_{22}, ... , a_{nn} are the diagonal elements of a square matrix.

Diagonal Matrix We call matrix a Diagonal Matrix if all non-diagonal elements equal to zero.On the other hand, only the diagonal entries of the square matrix can be non-zero, ($a_{ij} = 0$, $i \neq j$)

Example: $[A] = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 3.5 & 0 \\ 0 & 0 & 6 \end{pmatrix}$ **Zero Matrix** Zero Matrix is a matrix, of which all its entries are equal to zero. ($a_{ij} = 0$ for all i and j)

Example $[A] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$[B] = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

$[C] = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

$[D] = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}$

A, B, C, and D are all zero matrixes.

Note: if any matrix is multiplied by the Zero Matrix the answer is the Zero Matrix.

Matrix of Ones Matrix of Ones is a matrix, of which all its entries are equal to one. ($a_{ij} = 1$ for all i and j)

Example: $[A] = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$

$[B] = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

$[C] = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

$[D] = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}$

A, B, C, and D are all zero matrixes.

Note: if any matrix is multiplied by the Ones Matrix the answer is that same matrix.

Diagonally Dominant Matrix This only applied for matrix of $n \times n$ squar matrixes. and it has to meet the following condition:

$|a_{ii} \geq \Sigma_{j=1,i=j}^{n}|$ for all i=1,2,3 ...n and

$|a_{ii}| > \Sigma_{j=1,i=j}^{n}|a_{ij}|$ for at least one i,

that is, for each row, the absolute entry value of the diagonal element is greater than or equal $\geq$ to the sum of the absolute values of the rest of the elements of that row, and that the inequality is strictly greater than for at least one row. Such matrix are very important in ensuring convergence in iterative schemes of solving simultaneous linear equations, and liner problems.

Magic Square Matrix Magic Matrix is a square matrix where the sum of the row elements equal to the sum of the column elements.

Some special magic square matrix have the sum of element is a row equal to the sum of the element of column equal to the sum of the diagonals elements.

Example

$$[A] = \begin{pmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{pmatrix}$$

Another example o a square matrix:

$$[B] = \begin{pmatrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{pmatrix}$$

1.4 Notation

The fundamental importance of determinants as working tools in mathematics has come to be so widely recognized that it may be assumed that the reader has some practical knowledge of them, and in particular that he has realized their value in providing a simple general rule for the solution of linear equations. Certain introductory results may therefore be given without undue emphasis on intermediate steps, which can easily be supplied [**?**]. Let us learn about notations and go through some definitions. Suppose there are two homogeneous linear equations in three variables

x,y,z,

$$a_1x + b_1y + c_1z = 0 \tag{1.6}$$

$$a_2x + b_2y + c_2z = 0 \tag{1.7}$$

Then in general they have a solution

$$\frac{x}{b_1c_2 - b_2c_1} = \frac{y}{c_1a_2 - c_2a_1} = \frac{z}{a_1b_2 - a_2b_1} \tag{1.8}$$

We call those "**denominators**", which are called determinants of the second order [**?**], can be written shortly in various ways, all of them have great value.

- $|b_1c_2|$, $|c_1a_2|$, $|a_1b_2|$,
- (bc)12, (ca)12, (ab)12,
- (bc), (ca), (ab).

the last of these ways makes use of the obvious fact that if two letters bc are written down side by side, one if first and the other is second, read from left to right. We agree to drop the suffixes in the last item, whenever they are 1,2, for exactly the reason that we drop the index 1 in writing a^p when $p = 1$. In fact we define $(bc)_{ij}$ to mean $b_ic_i - b_jc_j$ and merely suppress the suffixes ij on the case when $i = 1$ and $j = 2$. A Fourth and more familiar notation for the determinant $b_1c_2 - b_jc_i$ is the well-known square array, introduced by Cayley in 1841 [**?**] long after determinants were first invented. It is:

$$\begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix}$$

which has the advantage of showing such coefficients of the original equations, as appear in the first determinant, exactly in their same relative positions.

APPENDIX B

NUMERICAL ANALYSIS

2.5 Introduction

In numerical analysis, numerical integration constitutes a broad family of algorithms for calculating the numerical value of a definite *integral*, and by extension, the term is also sometimes used to describe the numerical solution of differential equations [**?**]. The term **quadrature** is more or less a synonym for *numerical integration*, especially as applied to one-dimensional integrals. Two- and higher-dimensional integration is sometimes described as cubature, although the meaning of quadrature is understood for higher dimensional integration as well [**?**].

The most basic problem to be solved using numerical integration is to compute the approximate solution of a definite integral:

$\int_a^b f(x)dx$

2.6 Why Numeric Integration? Why not?

Several facts and reasons affect scientist to use numeric integrations. For example, the integrand function f, from the previous equation, may be known only at certain points, such as obtained by sampling. Many computer applications and computer embedded systems often use numerical integration for such a reason.

The absente of finding an antiderivative is also another reason of using Numeric Integration. A formula for the integrand may be known, but it may be difficult or impossible to find an antiderivative. example $\exp(t^{-2})$.

It may be possible to find an antiderivative symbolically, but it may be easier to compute a numerical approximation than to compute the antiderivative. That may be the case if the antiderivative is given as an infinite series or product, or if its

evaluation requires a special function which is not available.

Quadrature is a numerical method used to find the area under the graph of a function, that is, to compute a definite integral. In MatLab, there is a build in numerical integration functions that helps do the numerical integration. Those functions are called Quadrature Functions. Quad and Quadl are MatLab Build-in functions.

2.7 MatLab Build-in Quadrature Functions

2.7.1 Quad

quad q = quad(fun,a,b) tries to approximate the integral of function fun from a to b to within an error of 1e-6 using recursive adaptive Simpson quadrature. fun is a function handle. See Function Handles in the MATLAB Programming documentation for more information. The function y = fun(x) should accept a vector argument x and return a vector result y, the integrand evaluated at each element of x.

2.7.2 Quadl

Numerically evaluate integral, adaptive Lobatto quadrature $q = quadl(fun, a, b)$ approximates the integral of function fun from a to b, to within an error of 10-6 using recursive adaptive Lobatto quadrature. fun is a function handle. (write this myself)

2.7.3 Gaussian Quadrature

The numerical integration methods described so far are based on a rather simple choice of evaluation points for the function f(x). They are particularly suited for regularly tabulated data, such as one might measure in a laboratory, or obtain from

computer software designed to produce tables. If one has the freedom to choose the points at which to evaluate f(x), a careful choice can lead to much more accuracy in evaluating the integral in question.

APPENDIX C

THE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY DATABASE

"From automated teller machines and atomic clocks to mammograms and semiconductors, innumerable products and services rely in some way on technology, measurement, and standards provided by the National Institute of Standards and Technology" [?, ?]." from the NIST official website.

The National Institute of Standards and Technology NIST,Founded in 1901,is a non-regulatory federal agency within the U.S. Department of Commerce. NIST's mission is to promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life [**?**].

NIST ccarries out its mission in four cooperative programs, which are **The NIST Laboratories**, **The Bladrige National Quality Program**, **The Hollings Manufacturing Extension Partnership**, and most important **The Technology Innovation Program** which is planned to provide cost-shared awards to industry, universities and consortia for research on potentially revolutionary technologies that adress critical national and societal needs. More common information about the NIST org can be found on its website http://www.nist.gov.

3.8 Advance Technology Program (ATP)

Managed by NIST between 1990 and 2007, the Advanced Technology Program (ATP) bridges the gap between the research lab and the market place., simulating prosperity through innovation [**?**].

3.9 pSTAR Database

The PSTAR database is a program which will calculate stopping power and range tables for protons in various materials.

With that program a user an select a material and enter the desired input energies, or select default.The energies' unit will be in MeV and must be within the range[1]of 0.001 MeV to 10000 MeV.

[1] For our experiment, our input energy 250 MeV or 200 MeV, and or exit Energy will be less that 250 or 200 MeV since the proton has to travel through an object and it must loose at least a fraction of an energy

REFERENCES

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeXCompanion*. Addison–Wesley Pub. Co., Reading, MA, 1994.

[2] Donald E. Knuth. *The TeXbook*. Addison–Wesley Pub. Co., Reading, MA, 1984.

[3] Leslie Lamport. *LaTeX: A Document Preparation System*. Addison–Wesley Pub. Co., Reading, MA, 1986.

[4] F. Rellich. Darstellung der Eigenwerte von $\Delta u + \lambda u = 0$ durch ein Randintegral. *Math Z.*, 46:635–636, 1940.

[5] E. Zeidler. *Nonlinear Functional Analysis*, volume IIa. Springer Verlag, New York, 1988.