Modeling the tagged neutron technique for the identification of Unexploded Ordnance

A Thesis Presented

by

Yuanyuan Wang

to

The Graduate School
in partial fulfillment of the Requirements
for the degree of

Master of Science
In
Mechanical Engineering

Stony Brook University
August 2010

Stony Brook University

The Graduate School

Yuanyuan Wang

We, the thesis committee for the above candidate for the Master of Science degree,
Hereby recommend acceptance of this thesis.

Dr. Yu Zhou, Advisor,
Assistant Professor, Mechanical Engineering Department

Dr. Anurag Purwar, Advisor, Chairman of Thesis Committee,
Research Assistant Professor, Mechanical Engineering Department

Dr. Sudeep Mitra, Co-advisor,
Nuclear Engineer, Department of Environmental Sciences, Brookhaven National Laboratory

This thesis is accepted by the Graduate School

Lawrence Martin

Dean of the Graduate School

Abstract of the Thesis

Modeling the tagged neutron technique for the identification of Unexploded Ordnance

by

Yuanyuan Wang

Master of Science

in

Mechanical Engineering

Stony Brook University

2010

This research is a collaboration between the Department of Mechanical Engineering at Stony Brook University (PI: Yu Zhou) and the Department of Environmental Sciences at Brookhaven National Laboratory (Collaborator: Sudeep Mitra). The ultimate goal of this research is to develop the associated particle neutron time-of-flight (APTOF)-based UXO detection and discrimination technique and construct a prototype compact and portable neutron interrogation probe that will search for UXO in a target volume, locate targets in three dimensions, and identify the major elemental constituents of each target for discrimination. This project focuses on feasibility study, mainly consisting of proof-of-concept, experimentally modeling and data analysis. An associated particle time-of-flight (APTOF) system has been developed by using the GEANT4 4.9.2.p01 toolkit.

In this study, significant efforts have been made to develop a neutron interrogation UXO sensing system model based on GEANT4 simulation Toolkit (G4). This simulation includes modeling of the neutron beam, alpha beam sources and

collimation, the samples, the neutron interaction within the samples, the emission of characteristic gammas, and the detection of these gammas, as well as the detection of the alpha particles. Continuous coincident neutron flux and alpha particles were generated from D-T fusion reaction model and were captured by physical and geometric condition strict detectors in G4. A fast, time-saving associated particle imaging algorithm has been developed by correlating the time and direction of alpha particles and emission of gamma rays from fast neutrons, thus making the system capable of multi-dimensional imaging, as well as sensing elements such as C, N, O, Si, Al, Ti, etc.. Optimized time window has also been employed due to variable soil background can severely affect the signal-to-noise ratio for elemental measurement.

From the images and spectrums we can accurately reconstruct the shape, location and components of objects of interest that are hidden from view. Useful signals from an UXO buried to about 20cm in the sub-surface can be measured. Results from the simulation experiments demonstrate that GEANT4 is an effective simulation platform to develop the APTOF sensing system and to facilitate future development and optimization. In particular it can model the geometry of the system for optimum neutron reactions with nuclei in the sample, capture of gamma rays and alpha particles, as well as gamma energy deposition and 2D and 3D imaging.

Table of Contents

List of Figures vi

List of Tables viii

ACKNOWLEDGEMENTS 0

1. Introduction and Background 1

 1.1 Motivation 1

 1.2 Proposed Methods 4

 1.3 Concept 7

2. Neutron source and alpha particle generation 9

 2.1Deuterium and Tritium fusion reaction 9

 2.2 Kinematics 10

 2.3 Neutron source generation 11

 2.4 Alpha particle generation 12

 2.5 Discussion of incident deuteron energy 16

3. GEANT4 Simulation 17

 3.1 Introduction of GEANT4 17

 3.2 Class Categories in GEANT4 18

 3.2 GEANT4 Virtual System 22

  3.2.1 Definition of main( ) Function 23

  3.2.2 System Layout 26

  3.2.3 Primary Event 33

  3.2.4 Data Collection 36

  3.2.5 Visualization[13] 37

  3.2.6 Run 39

4. Data Analysis (APTOF Imaging) 41

 4.1 Introduction 41

 4.1Kinematics 42

5. Experiments and Simulation Results 47

 5.1 Simulation Results 47

 5.2 Time Window 56

6. Conclusion and Future Work 63

Bibliography 65

List of Figures

1.1 Diagram of APTOF concept **7**

2.1 D+T reaction (http://en.wikipedia.org/wiki/Nuclear_fusion) 9

2.2 Two-body reaction 10

2.3 Angle between alpha particle and incident beam for Deuteron bombarding energy E_0 14

2.4 A snapshot of DT reaction result 16

3.1 GEANT4 class categories 20

3.2 A snapshot of G4 main function 25

3.3 A snapshot of APTOF system layout source file 27

3.4 (a) (b) G4 World volume 29

3.5 Examples of UXO in G4 29

3.6 Examples of environment surround UXO 30

3.7 A snapshot of APTOF detectors construction source file 31

3.8 Front-view and top-view of APTOF detectors 32

3.9 A snapshot of APTOF physics list source file 34

3.10 A snapshot of APTOF primary event source file 35

3.11 APTOF primary event 36

3.12 A snapshot of APTOF hit source file 37

3.13 A snapshot of APTOF event action source file 38

3.14 User Interface for GEANT4 program 39

3.15 A G4 display with neutrons and alpha particles 40

4.1 Correlation between neutron and alpha particle 45

4.2 Correlation between neutron and alpha particle 45

4.2 Air definition in G4 47

5.2 The outputs of APTOF system
(a) Sample made of Carbon under investigation in GEANT4
(b) 2D and 3D reconstruction of carbon-made object
(c) Gamma-ray spectrum of Carbon.
(d) 2D and 3D reconstruction of Iron-made object
(e) Gamma-ray spectrum of Iron. 50

5.3 Soil definition in G4.......... 50

5.4 The outputs of APTOF system
(a) System running with object in soil
(b) 2D and 3D reconstruction of bomb in air
(c) 2D and 3D reconstruction of bomb in soil
(d) Colorful Gamma-ray spectrum.......... 52

5.5 Energy spectrum of bomb buried 5cm in soil 54

5.6 3D and 2D image reconstruction of bomb buried 5cm in soil 54

5.7 Energy spectrum of bomb buried 5cm in soil 55

5.8 3D and 2D image reconstruction of bomb buried 10cm in soil 55

5.9 Intensity Decay V.s Depths in Soil and error bar 56

5.10 First time window simulation 57

5.11 Second time window simulation 57

5.12 Histogram of the first time window 58

5.13 Histogram of the second time window 59

5.14 Comparison in spectrum between with-time-window and without-time-window for Hexagen 5cm in soil 60

5.15 Comparison in spectrum between with-time-window and without-time-window for Hexagen 10cm in soil.......... 60

5.16 Comparison in spectrum between with-time-window and without-time-window for Hexagen 15cm in soil.......... 61

5.17 Comparison in spectrum between with-time-window and without-time-window for Hexagen 20cm in soil 61

List of Tables

2.1 Neutron Energy Vs Lab Angle for Deuteron Bombarding Energy E_0 11

2.2 Angle between Alpha Particle and Incident Beam for Deuteron Bombarding Energy E_0 14

2.3 and azimuthal angle and cone angle of neutron and alpha particle 15

3.1 Configurations of APTOF alpha detector and gamma detectors .... .31

4.1 List of particles 43

5.1 Configuration of Carbon in Air simulation.. 47

5.2 Configuration of Bomb in Soil simulation 50

5.3 Composition of Hexagen 53

ACKNOWLEDGEMENTS

I take this opportunity to thank my advisor, Prof. Yu Zhou, for his guidance and invaluable support throughout my academic and research term at Stony Brook University. He has extensive knowledge of subject, which he willing shared with me. It is because of him I have successfully been able to complete my thesis.

I would also like to extend my thanks to Dr. Sudeep Mitra for being as my co-advisor during this project. It is because of his tremendous research experience and valuable suggestions that helped with the completion of my thesis.

Also I want to thank Prof. Anurag Purwar for agreeing to be my thesis committee chair.

My parents and cousins have been my backbone and have stood by me throughout. I am indebted to them, all this would not have been possible without their encouragement and faith in my capabilities. Last but not the least, my close friends at Stony Brook University made experience a memorable one, I cannot thank them enough.

As a graduate student of Stony Brook University, I have learned so much and grown as a mature person, in the past two years. The environment here has been educating, inspiring and pleasant. I am now confident and well rounded in all specialties and have a vision of how I see myself grow in the future. I owe a lot to Prof. Yu Zhou, my advisor and mentor.

Chapter 1

1. Introduction and Background

1.1 Motivation

Cleaning up military sites suspected of containing munitions that have been armed and fired yet remain unexploded is one of the most pressing environmental problems. In particular, military downsizing after the Cold War has resulted in the closure of numerous military bases. Land that was once host to training ranges for such activities as firing practice and weapons testing is prepared for transfer to civilian ownership. The presence of unexploded ordnance (UXO) has interfered with the efforts to transfer and sell this land because of the potential hazards it poses to civilians[1].

The term “unexploded ordnance (UXO)” refers to any kind of munition that was fired but failed to explode upon impact; it can also refer to buried but long forgotten weapons caches or disposal sites[1]. Types of UXO span the full array of ammunition employed on the battlefield, including tank shells, artillery rounds, bombs, rockets, missiles, mortars, hand grenades, rifle grenades, bulk explosives, detonators, aircraft cannon, torpedoes, mines, pyrotechnics, chemical munitions, submunitions, and small-arms ammunition. UXO has posed a worldwide risk both in former combat areas and on military firing ranges. In US, the military tests and trains with live munitions to maintain readiness at all times; however, not all munitions detonate as designed. A large portion of UXO results from the failure of weapons fired during testing or training to detonate, due to either a malfunction in the arming process or operator error. On average, the failure rate in the field is about 10 percent. Moreover, “for troop practice firings at training sites, there is normally no record of number of rounds fired vs. number of duds”, where a dud is an ammunition round or explosive that fails to fire or detonate on time or on command. This lack of record-keeping also increases the level of risk[2].

UXO presents both explosion and contamination risks. As long as UXO is in the ground, the risk of someone disturbing the munition and causing an unexpected explosion remains. Hundreds of UXO-caused civilian injuries and deaths have been documented. It is expected that casualties will increase as civilians gain access to closed bases. Besides the obvious danger of explosion, buried UXO also entails the risk of environmental contamination. Munitions constituents include residue resulting from a munition that has partially detonated, the open burning of excess explosives, the corrosion of UXO items, and the breakage of munitions without detonation. Common explosives that may remain in soil at former military training sites include trinitrotoluene (TNT), royal demolition explosive (RDX), high-melting explosive (HMX), and various isomers of dinitrotoluene (DNT)[2]. While the concentrations found in the study were generally low and decreased as the distance from the target areas increases, most ranges did have localized sources with high concentrations. Munitions chemicals can enter and contaminate groundwater. According to US Environmental Protection Agency (EPA) documents released in late 2002, UXO at 16,000 domestic inactive military ranges within the United States pose an “imminent and substantial” public health risk and could require the largest environmental cleanup ever, at a cost of at least $14 billion.

Cleaning up military sites suspected of containing UXO has become one of the most pressing environmental problems. Unfortunately, it has proved technically challenging and extraordinarily expensive. Two classes of sensors are mostly used for UXO detection. They are magnetometers and electromagnetic induction (EMI) devices[3]. Magnetometers detect ferrous metal objects by measuring changes in the Earth's magnetic field caused by the object. They are passive devices that respond to ferrous materials, such as iron or steel, but will not respond to metals that are not ferromagnetic, such as copper, tin, and aluminum, and non-magnetic materials. EMI transmits an electromagnetic field, which in turn induces a secondary magnetic field in objects that are conductive. When secondary magnetic fields of military munitions and other conductive items exceed background responses, they can be identified as potential anomalies requiring further investigation. However, it is particularly difficult to distinguish UXO from magnetic rocks and soils, especially in complex geological settings throughout US. Much of the cost of UXO removal comes from removing non-explosive items that the metal-detectors identified, so improved discrimination is critical.

Neutron interrogation has recently been considered as a potential technique for distinguishing explosive materials from each other and from innocuous materials, based on the principle that different materials should have different quantities and ratios of nitrogen (N), oxygen (O) and carbon (C). Neutron interrogation can identify N, O and C present in UXO.

Currently there are two neutron interrogation systems under development and test, the Portable Isotopic Neutron Spectroscopy (PINS) system[4] and the Pulsed Elemental Analysis with Neutrons (PELAN) system [5]. These systems exploit the two types of neutron interactions with nuclei — inelastic scattering and neutron capture, followed by detection of induced element-specific high-energy prompt gamma rays. However, the application of the neutron interrogation technique has been hindered by some unresolved issues. The most important problem is that neutrons interact with not only the interrogated material but also all surrounding materials, resulting in a very high background gamma-ray spectrum. The detecting system requires bulky shielding, and the signal-to-noise ratio for element measurement is poor. In fact, it has been pointed out in the 2004 phase 1 demonstration report summary of the PELAN system that since soil contains many of the same elements (likely to be dominated by O) that are also inside a munitions shell, the signal-to-noise ratio is affected by the background[5]. We believe that this variable soil background can severely affect the sensor's capability. Moreover, since there is no information about the position sensitivity of the detected elements, suppressing the effect of the background is difficult.

To address the inadequacy of conventional neutron interrogation techniques described above, we propose an innovative sensing technique for non-destructively discriminating UXO from non-hazardous materials based on a novel neutron interrogation technique with APTOF capability.

1.2 Proposed Methods

We propose to use the associated particle time-of-flight (APTOF) technique to detect and discriminate near surface UXO. APTOF is an active neutron probe technique which has the potential to provide a 3-D image with elemental composition of the material under interrogation. D+T reaction produces coincident neutrons at 14.1 MeV and alpha particles at 3.5 MeV, traveling in opposite directions. The two particles are associated in space and time with respect to the neutron generation site in the tritium target inside the neutron generator. This correlation is used to tag a specific fraction of the neutrons that are emitted. By detecting the arrival of the alpha particle on a position-sensitive photomultiplier tube (PS-PMT) and its position in the x-y plane, the time and direction of the corresponding neutron emission can be determined.

The GEANT4 simulation toolkit has been developed for particle physics, and an ever growing list of laboratories and institute are using GEANT4 as baseline for their simulation needs concerning detector design and radiation shielding problems. More importantly, there are a growing number of high energy physics experiments that use the technology provided by this program in the preparation of the detector and the interpretation of the data.

The logistics of generating the neutron beam source and alpha particle source, as well as the time-consuming nature of performing experiments make the development of a simulation environment a desirable design methodology for our detecting system. The GEANT4 toolkit's main design purpose is the simulation of high-energy particles interacting with matter in high energy physics applications. It was selected for this research primarily because its design encompasses our domain of interest: high-energy neutron interactions in matter. It includes facilities for handling geometry, tracking, detector response, run management, visualization and user interface. For many physics simulations, this means less time need be spent on the low level details, and researchers can start immediately on the more important aspects of the simulation. GEANT4 toolkit is a collection of C++ class libraries and GEANT4 Neutron Data

Libraries (G4NDL)[6]. In our research, the simulated system contains the target (UXO), neutron and alpha particle generators, alpha detectors, and gamma ray detectors. Also in the PhysicsList file, we constructed different types of particles generated from different reaction functions.

Associated Particle Time-Of-Flight technique(APTOF) is used to analyze the gamma ray, alpha particle information obtained from the G4 program. The traditional approach has been used for decades, however, it has some severe limitations. Gamma rays resulting from surrounding clutter, such as, the ground, structural components, items intended to hide the threat, the barrier material, etc., may all be sources of background "noise" in the neutron interrogation. The APTOF method adds one important additional feature to the general neutron interrogation method. This improvement relies on "tagging" the individual neutrons used in the interrogation so that only those gamma rays that are produced in the region of interest that contains the threat material are counted[7]. The neutron tagging is accomplished by detecting the direction and time of arrival of the alpha particle that is associated with the generation of an individual neutron in the neutron generator. This allows one to focus on only those neutrons (and any gamma rays that they may produce) from the region of interest containing the threat material. The influence of gamma rays from background material is significantly reduced; this greatly improves the probability of detection and reduces the incidence of false alarms.

A data analysis framework-ROOT software has been used to obtain 3D and 2D images of the target, as well as the gamma-ray energy spectrum of its composition of material. From the images and spectrum we can clearly tell the shape, location and the components of the target of interest that are hidden from view.

1.3 Concept

The proposed APTOF-based UXO detecting system uses a small generator to produce a continuous, monoenergetic flux of neutrons. See Figure 1.1.

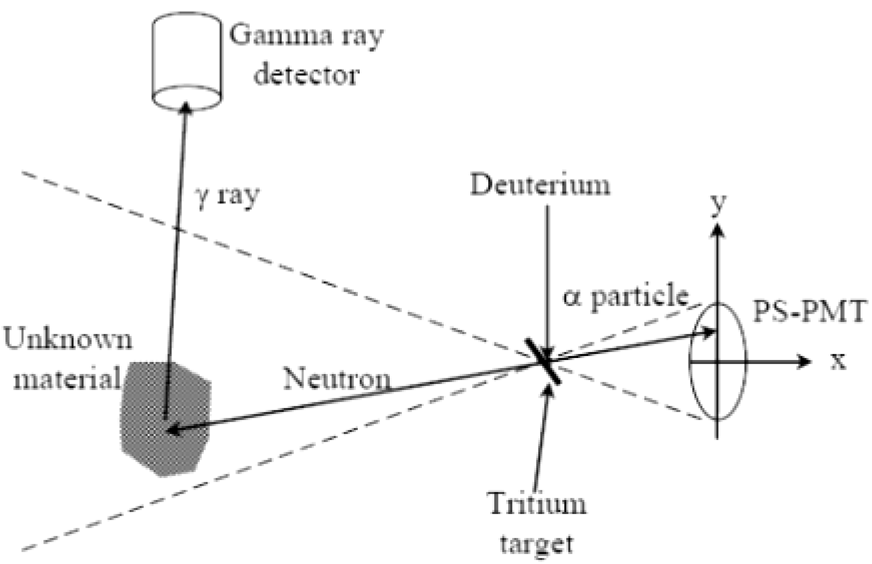


Figure1.1: Diagram of APTOF concept.

Interactions between a neutron and the material in its path often produce a gamma ray whose energy is characteristic of that material. Many elements, especially N, O and C which are of particular interest in the context of UXO detection and discrimination, produce prompt gamma rays under fast neutron bombardment with sufficient probability and energy (500 keV) for the application of APTOF. These gamma rays will be detected by a suitable detector. For those detected gamma rays coincident within a short time after the alpha particle emission, the energy and the time-of-arrival of the gamma rays relative to the emission time of the alpha particle will be recorded. Since the velocities of the 14.1

MeV neutron and the resulting gamma ray are both known constants (14 MeV neutrons travel at 5 cm/ns while gamma rays travel at 30 cm/ns), given the known position of the gamma ray detector relative to the neutron generator, the travel distance of the neutron can be inferred from the arrival time of the gamma ray.

The thesis is organized as follows. A description of D+T reaction as well as generation of coincident neutrons and alpha particles can be found in chapter 2. The GEANT4 simulation environment created to model the detecting process is presented in chapter 3. The methodology of APTOF analysis for spectral interrogation and imaging process are detailed in chapter 4. Simulations and optimized results of are illustrated in chapter 5. The final chapter summarizes the work of this research and makes a few salient points regarding the future development of this work.

Chapter 2

2. Neutron source and alpha particle generation

2.1Deuterium and Tritium fusion reaction

The most common type of fusion reaction discussed for fusion energy in the near future is the fusion of two hydrogen isotopes: deuterium (^2H) and tritium (^3H). It is the easiest fusion reaction to achieve on Earth, and will most likely be the type of reaction found in first generation fusion reactors[8]. Figure 3.1, shows the actual reaction involves a deutrerium nucleus fusing with a tritium nucleus to form an alpha particle (^4He nucleus) and a neutron. The products contain around 17.6 million electron volts (MeV) of released kinetic energy through the loss of mass in the fusion process (see Eqn. 2.1)[9].

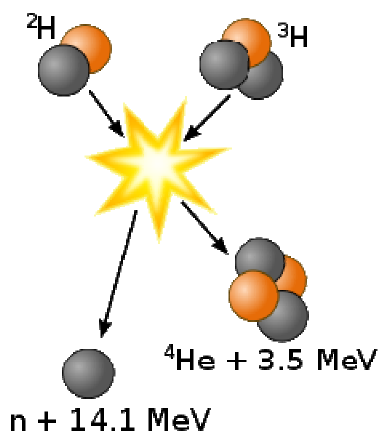


Figure 2.1: D+T reaction. (http://en.wikipedia.org/wiki/Nuclear_fusion)

$$ {}_2^2H + {}_3^3H \rightarrow {}_2^4He + {}_0^1n + 17.6MeV \qquad \text{Eqn.2.1} $$

The D-T reaction is the easiest because the extra neutrons on the nuclei of the deuterium and tritium increase their size and thus the probability of a fusion reaction. They also each have the smallest possible positive charge (since hydrogen has only one proton), making it relatively easy to have the two nuclei overcome their repulsion and fuse together.

2.2 Kinematics

Consider the following two-body reaction,see Fig.2.2. Initially nucleon 1 is incident on nucleon 2 which is at rest in the lab system; after the interaction only nucleons 3 and 4 emerge.

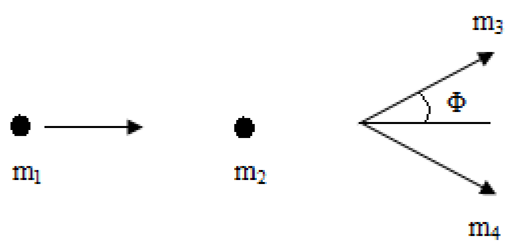


Figure 2.2: Two-body reaction.

The general expression for the energy of nucleon 3 in the lab system can be written in the form[9]:

$$E_l(m_3,\phi) = \left(\frac{m_1}{m_1+m_2}\right)^2 \frac{m_3}{m_1} E_0 \cos 2\phi \frac{m_4}{m_3+m_4}\left(\frac{m_2}{m_1+m_2}E_0+Q\right)$$
$$\pm \frac{2\cos\phi}{m_1+m_2}\sqrt{m_1 m_3 \frac{m_4}{m_3+m_4}E_0\left(\frac{m_2}{m_1+m_2}E_0+Q\right)} \qquad \text{Eqn.2.2}$$
$$\times\sqrt{1-\frac{m_1 m_3}{m_1+m_2}\frac{E_0 \sin^2\phi}{m_4\left(\frac{m_2}{m_1+m_2}E_0+Q\right)}}$$

Where E_0 is the energy of nucleon 1 in the lab system, Φ is the lab angle made by the velocity vector of nucleon 3 with the beam direction, the m_i's are the respective nucleon masses, and the Q of the reaction is defined customarily as $Q = m_3+m_4-m_1-m_2$.

2.3 Neutron source generation

For the case of the reaction $d+T \rightarrow n+He^4$, equation 2.2 yields for the neutron energy[10]:

$$E_L(E_0,\phi_n) = 0.08E_0\cos 2\phi_n + 0.8(0.6E_0+17.6) + 0.8\cos\phi_n\sqrt{0.4E_0(0.6E_0+17.6)} \times \sqrt{1-\frac{E_0\sin^2\phi_n}{10(0.6E_0+17.6)}} \qquad \text{Eqn.2.3}$$

Q= 17.6 Mev. It stands for the total released kinetic energy.

In Table 2.1, Eqn. 2.3 has been evaluated as a function of Φ_n for three value of the parameter E_0. Since the tritium target is at rest (relative to the range of the incident deuterons) deuterons of all energies from the bombarding energy to zero will yield neutrons.

Table 2.1: Neutron Energy Vs Lab Angle for Deuteron Bombarding Energy E_0.

Φ_n (lab)°	E_0=0.1MeV	E_0=0.2MeV	E_0=0.3MeV	E_0=0.4MeV	E_0=0.5MeV
0	14.808	15.144	15.417	15.655	15.874
10	14.797	15.129	15.398	15.632	15.849
20	14.766	15.083	15.34	15.566	15.774
30	14.714	15.009	15.248	15.458	15.65
40	14.644	14.908	15.123	15.312	15.486

50	14.546	14.785	14.971	15.134	15.286
60	14.46	13.644	14.795	14.931	15.056
70	14.352	14.49	14.605	14.71	14.806
80	14.238	14.326	14.404	14.485	14.544
90	14.12	14.166	14.2	14.24	14.28
100	14.004	13.996	13.998	13.999	14.02
110	13.892	13.838	13.807	13.786	13.772
120	13.788	13.692	13.629	13.581	13.544
130	13.695	13.561	13.445	13.398	13.34
140	13.614	13.45	13.333	13.244	13.168
150	13.55	13.359	13.224	13.118	13.03
160	13.502	13.293	13.144	13.026	12.974
170	13.473	13.253	13.096	12.972	12.867
180	13.464	13.243	13.079	12.953	12.819

A program in Matlab was written to generate random, uniformly distributed Φ angles(the azimuthal angle) within (0-2π) direction. Based on equation 2.3, data of E_0, Φ_n is collected, θ is the cone angle.

2.4 Alpha particle generation

In the lab frame of reference, the neutron and associated α-particle from a given D-T encounter are to be found. Eq. 2.4[10] relates Φ_α, the angle made by the the α-particle with beam direction, to Φ_n, that made by the neutron and the beam direction, see Fig.2.3.

$$\tan\phi_\alpha = \frac{\frac{1}{2}\sin^2\phi_n + \sin\phi_n\sqrt{\frac{1}{\gamma^2} - \sin^2\phi_n}}{-\sin^2\phi_n + \cos\phi_n\sqrt{\frac{1}{\gamma^2} - \sin^2\phi_n} - \frac{m_\alpha}{m_n}} \qquad \text{Eqn.2.4}$$

Here, $\frac{1}{\gamma^2} = \frac{Vn}{Vc.m.} = \frac{m_\alpha}{m_n}\frac{(m_1+m_2)}{m_1}(\frac{m_2}{m_1+m_2}+\frac{Q}{E_0})$

Where m_1= mass of incident particle

m_2= mass of target particle

m_n= mass of neutron

m_α= mass of alpha

E_0= incident energy

In Table 2.2, Eqn. 2.4 has been evaluated for several values of the deuteron bombarding energy, E_0.

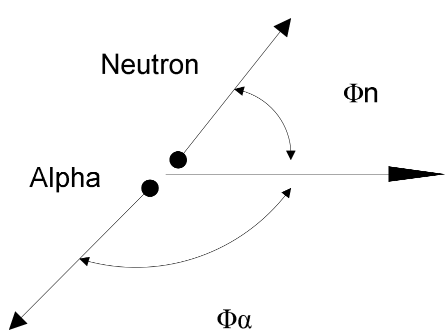


Figure 2.3: Angle between alpha particle and incident beam for Deuteron bombarding energy E_0

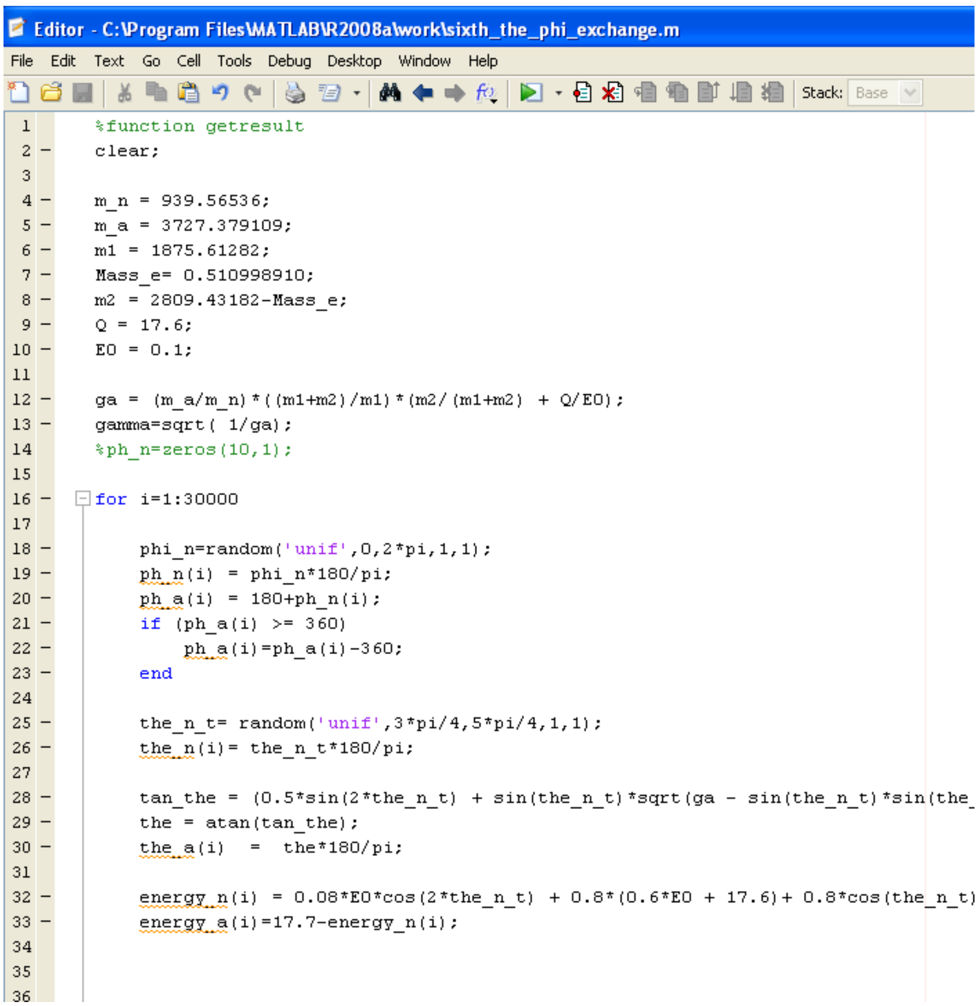
Table 2.2: Angle between Alpha Particle and Incident Beam for Deuteron Bombarding Energy E_0.

Φ_n (lab)°	E_0=0.1MeV	E_0=0.2MeV	E_0=0.3MeV	E_0=0.4MeV	E_0=0.5MeV
0	180°	180°	180°	180°	180°
10	168°41’	168° 04’	167° 36’	167° 06’	166° 40’
20	157°26’	156°14’	154° 58’	154° 22’	153° 33’
30	146° 17’	144° 34’	143° 10’	141° 56’	140° 48’
40	135° 17’	133° 08’	131° 24’	129°54’	128° 32’
50	124° 27’	121° 58’	120° 01’	118° 19’	116° 48’
60	113° 49’	111° 07’	109° 01’	107° 13’	105° 37’
70	103° 24’	100° 36’	98° 26’	96° 36’	94° 59’
80	93° 12’	90° 22’	88° 13’	86° 25’	84° 51’
90	83° 12’	80° 27’	78° 23’	76° 39’	75° 10’
100	73° 25’	70° 48’	68° 51’	67° 16’	65° 53’
110	63° 49’	61° 24’	59° 58’	58° 11’	56° 56’
120	54° 22’	52° 14’	50° 39’	49° 23’	48° 17’
130	45° 05’	43° 14’	41°53’	40° 48’	39° 53’
140	35° 55’	34° 24’	33° 18’	32° 25’	31° 40’
150	26°51’	25° 41’	24° 51’	24° 11’	23° 37’
160	17° 51’	17° 04’	16° 31’	16° 04’	15° 31’
170	8°55’	8° 31’	8° 12’	8° 00’	7° 49’
180	0	0	0	0	0

From Table 2.1 and Table 2.2, Table 2.3 is plotted, which includes six columns of data, each stands for En, θn $\Phi_{n,}$ Eα, θα, Φα, respectively,see Fig 2.4. These datum will be imported to G4 program later, as for the position and energy definition of neutrons and alpha particles generated from the generators in G4 program.

Table 2.3: Energy and azimuthal angle and cone angle of neutron and alpha particle.

$E_{neutron}$(MeV)	θn	Φn	E_{alpha}(MeV)	θ_α	Φα
13.51	202.13	68.76	4.19	19.77	248.76
13.6	217.45	163.07	4.1	33.61	343.07
13.56	212.51	72.46	4.14	29.13	252.46
13.6	141.46	279.2	4.1	-34.6	99.2
13.63	137.49	82.23	4.07	-38.22	262.23
13.49	196.84	354.75	4.21	15.03	174.75
13.49	197.38	55.61	4.21	15.51	235.61
13.63	222.53	244.15	4.07	38.24	64.15
13.6	141.59	289.2	4.1	-34.48	109.2
13.47	186.92	352.47	4.23	6.17	172.47




```
%function getresult
clear;

m_n = 939.56536;
m_a = 3727.379109;
m1 = 1875.61282;
Mass_e= 0.510998910;
m2 = 2809.43182-Mass_e;
Q = 17.6;
E0 = 0.1;

ga = (m_a/m_n)*((m1+m2)/m1)*(m2/(m1+m2) + Q/E0);
gamma=sqrt( 1/ga);
%ph_n=zeros(10,1);

for i=1:30000

    phi_n=random('unif',0,2*pi,1,1);
    ph_n(i) = phi_n*180/pi;
    ph_a(i) = 180+ph_n(i);
    if (ph_a(i) >= 360)
        ph_a(i)=ph_a(i)-360;
    end

    the_n_t= random('unif',3*pi/4,5*pi/4,1,1);
    the_n(i)= the_n_t*180/pi;

    tan_the = (0.5*sin(2*the_n_t) + sin(the_n_t)*sqrt(ga - sin(the_n_t)*sin(the_
    the = atan(tan_the);
    the_a(i)  =  the*180/pi;

    energy_n(i) = 0.08*E0*cos(2*the_n_t) + 0.8*(0.6*E0 + 17.6)+ 0.8*cos(the_n_t)
    energy_a(i)=17.7-energy_n(i);
```

Figure 2.4: A snapshot of DT reaction result

2.5 Discussion of incident deuteron energy

In T(d,n) He^4 reaction, generally, the tritium target is fixed with zero initial energy. It is interesting to note that the initial energy of deuterium E_0 will affect the folding angle between neutron and alpha particle ($\theta_{n\alpha}$). If both D and T are at rest, the generated neutron and alpha will be moving exactly in opposite direction. If D has some initial energy, From practical experience, the angular dependence of neutrons in the c.m. system for several bombarding energies may be represented as[11]:

$$+\frac{\sigma(\theta)}{\sigma_n(0)} = 1 \qquad \text{E<200 keV}$$

$$+\frac{\sigma(\theta)}{\sigma_n(0)} = 0.998 + 0.0213\cos\theta_n - 0{,}0190\cos^2\theta_n \qquad \text{E=350 keV}$$

$$+\frac{\sigma(\theta)}{\sigma(0)} = 0.992 + 0.0382\theta_n - 0.303\cos^2\theta_n \qquad \text{E=500 keV}$$

Since the deuterium energy of most portable commercial neutron generators works around 0.1MeV, we will only simulate situation when E_0=0.1MeV, when the folding angle between neutron and alpha particle $\theta_{n\alpha}$ is ideally considered to be 180˚.

Chapter 3

3. GEANT4 Simulation

3.1 Introduction of GEANT4

GEANT4 is a free software package composed of tools which can be used to accurately simulate the passage of particles through matter[12]. It was developed at CERN for high-energy, nuclear and accelerator physics, as well as studies in medical and space sciences. GEANT4 uses the C++ object-oriented computer language. Users can implement the description of the geometry and the materials of the detector, particle transport and the physics processes of interaction with materials and a particle generator with flexible visualization of geometries, tracks and interactions.

All aspects of the simulation process have been included in the toolkit:

- the geometry of the system,
- the materials involved,
- the fundamental particles of interest,
- the generation of primary events,
- the tracking of particles through materials and electromagnetic fields,
- the physics processes governing particle interactions,
- the response of sensitive detector components,
- the generation of event data,

- the storage of events and tracks,
- the visualization of the detector and particle trajectories, and
- the capture and analysis of simulation data at different levels of detail and refinement.

Users may construct stand-alone applications or applications built upon another object-oriented framework. In either case the toolkit will support them from the initial problem definition to the production of results and graphics for publication.

At the heart of GEANT4 is an abundant set of physics models to handle the interactions of particles with matter across a very wide energy range. Data and expertise have been drawn from many sources around the world and in this respect, GEANT4 acts as a repository which incorporates a large part of all that is known about particle interactions.

3.2 Class Categories in GEANT4

In object-oriented analysis class categories are used to create logical units. The class category diagram designed for GEANT4 is shown in the figure below. Each box in Fig. 3.1 represents a class category, and a "uses" relation by a straight line. The circle at an end of a straight line means the class category which has this circle uses the other category[13].

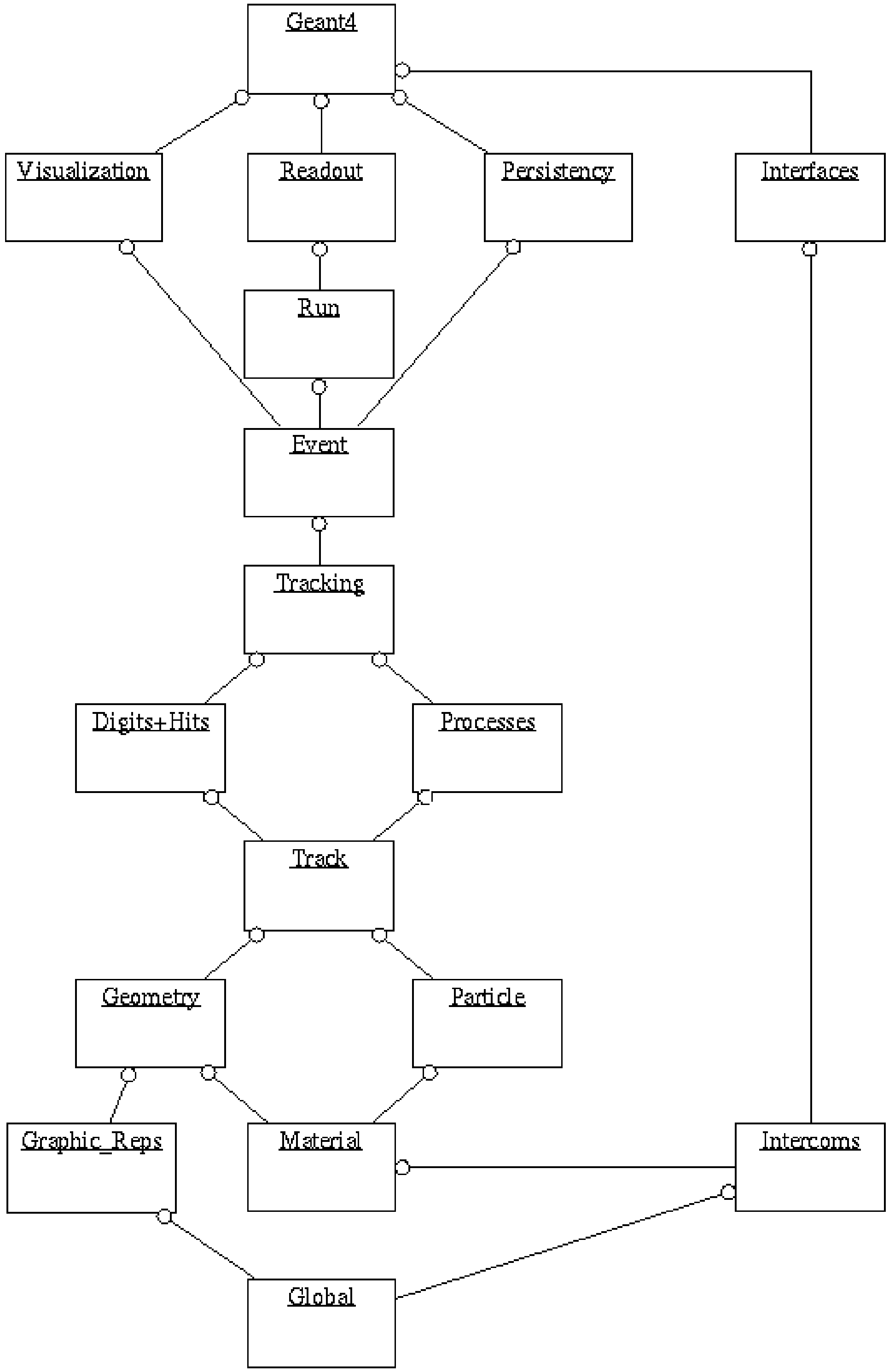


Figure 3.1: GEANT4 class categories

The following is a brief summary of the role of each class category in GEANT4[14].

(1) Run and Event

These are categories related to the generation of events, interfaces to event generators, and any secondary particles produced. Their roles are principally to provide particles to be tracked to the Tracking Management.

(2) Tracking and Track

These are categories related to propagating a particle by analyzing the factors limiting the step and applying the relevant physics processes. The important aspect of the design was that a generalized GEANT4 physics process (or interaction) could perform actions, along a tracking step, either localized in space, or in time, or distributed in space and time (and all the possible combinations that could be built from these cases).

(3) Geometry and Magnetic Field

These categories manage the geometrical definition of a detector (solid modeling) and the computation of distances to solids (also in a magnetic field). The GEANT4 geometry solid modeler is based on the ISO STEP standard and it is fully compliant with it, in order to allow in future the exchange of geometrical information with CAD systems. A key feature of the GEANT4 geometry is that the volume definitions are independent of the solid representation. By this abstract interface for the G4 solids, the tracking component works identically for various representations. The treatment of the propagation in the presence of fields has been provided within specified accuracy. An OO design allows us to exchange different numerical algorithms and/or different fields (not only B-field), without affecting any other component of the toolkit[14].

(4) Particle Definition and Matter

These two categories manage the definition of materials and particles.

(5) Physics

This category manages all physics processes participating in the interactions of particles in matter. The abstract interface of physics processes allows multiple implementations of physics models per interaction or per channel. Models can be selected by energy range, particle type, material, etc. Data encapsulation and polymorphism make it possible to give transparent access to the cross sections (independently of the choice of reading from an ascii file, or of interpolating from a tabulated set, or of computing analytically from a formula). Electromagnetic and hadronic physics were handled in a uniform way in such a design, opening up the physics to the users[14].

(6) Hits and Digitization

These two categories manage the creation of hits and their use for the digitization phase. The basic design and implementation of the Hits and Digitization had been realized, and also several prototypes, test cases and scenarios had been developed before the alpha-release. Volumes (not necessarily the ones used by the tracking) are aggregated in sensitive detectors, while hits collections represent the logical read out of the detector. Different ways of creating and managing hits collections had been delivered and tested, notably for both single hits and calorimetry hits types. In all cases, hits collections had been successfully stored into and retrieved from an Object Data Base Management System.

(7) Visualization

This manages the visualization of solids, trajectories and hits, and interacts with underlying graphical libraries (the Visualization class category). The basic and most frequently used graphics functionality had been implemented already by the alpha-release. The OO design of the visualization component allowed us to develop several drivers independently, such as for OpenGL and OpenInventor (for X11 and Windows), DAWN, Postscript (via DAWN) and VRML[14].

(8) Interfaces

This category handles the production of the graphical user interface (GUI) and the interactions with external software (OODBMS, reconstruction etc.)[14].

3.2 GEANT4 Virtual System

In our APTOF simulation, we constructed a stand-alone application, in which primary generators will generate fast neutron flux and alpha particles at the same time with energy at 14.1MeV and 3.5MeV respectively; gamma rays are produced and recorded when neutron hits the target of interest. The target is designed to user's preference. In the next chapter we will do several experiments with different objects under investigation, as well as different surrounding environments. Additionally, gamma detectors and alpha detector are designed to capture gamma rays and alpha particles. The time and data of each particle obtained from the G4 detectors will be analyzed in ROOT for 2D/3D image reconstruction and energy spectrum production.

3.2.1 Definition of main( ) Function

The main( ) method is implemented by two toolkit classes, *G4RunManager* and *G4UImanager*, and three classes, *APTOFDetectorConstruction*, *APTOFPhysicsList*, *APTOFPrimaryGeneratorAction*, *APTOFEventAction.* In Fig.3.2, the first thing main() must do is create an instance of the G4RunManager class. It controls the flow of the program and manages the event loop within a run. When G4RunManager is created, the other major manager classes are also created. They are deleted automatically when G4RunManager is deleted. The run manager is also responsible for managing initialization procedures, including methods in the user initialization classes. Through these the run manager must be given all the information necessary to build and run the simulation, including[14]:

1. How the APTOF detector should be constructed.
2. All the particles and all the physics processes to be simulated.
3. How the primary particles in an event should be produced
4. Any additional requirements of the simulation.

```
int main(int argc,char** argv)
{
  // User Verbose output class
  //
//  G4VSteppingVerbose* verbosity = new APTOFSteppingVerbose;
//  G4VSteppingVerbose::SetInstance(verbosity);

  // Run manager
  //
  G4RunManager * runManager = new G4RunManager;

  // User Initialization classes (mandatory)
  //
  APTOFDetectorConstruction* detector = new APTOFDetectorConstruction;
  runManager->SetUserInitialization(detector);
  //
  G4VUserPhysicsList* physics = new LHEP_PRECO_HP;//APTOFPhysicsList;//LHEP_PRECO_HP;
  runManager->SetUserInitialization(physics);

#ifdef G4VIS_USE
  // Visualization, if you choose to have it!
  //
  G4VisManager* visManager = new G4VisExecutive;
  visManager->Initialize();
#endif

  // User Action classes
  //
  G4VUserPrimaryGeneratorAction* gen_action = new APTOFPrimaryGeneratorAction();
  runManager->SetUserAction(gen_action);
  //

//  G4UserRunAction* run_action = new APTOFRunAction;
//  runManager->SetUserAction(run_action);
  //
  G4UserEventAction* event_action = new APTOFEventAction();
  runManager->SetUserAction(event_action);
  //
  //G4UserSteppingAction* stepping_action = new APTOFSteppingAction;
  //runManager->SetUserAction(stepping_action);

  // Initialize G4 kernel
  //
  runManager->Initialize();
```

Figure 3.2: A snapshot of G4 main function

In the APTOF main file,

```
APTOFDetectorConstruction* detector = new APTOFDetectorConstruction;
runManager->SetUserInitialization(detector);
G4VUserPhysicsList* physics = new APTOFPhysicsList;
runManager->SetUserInitialization(physics);
```

These lines of commands create objects which specify the gamma ray detector and alpha particle detector and pass their pointers to the run manager. This is where we describes the entire detector setup, including:

- Geometry
- Materials used in its construction
- Definition of its sensitive regions
- Readout schemes of the sensitive regions

Similarly, *APTOFPhysicsList* requires the user to define the particles to be used in the simulation; all the physics processes to be simulated and the range cuts for these particles.

In line,

```
G4VUserPrimaryGeneratorAction* gen_action = new APTOFPrimaryGeneratorAction();
```

It creates an instance of a particle generator and passes its pointer to the run manager. APTOFPrimaryGeneratorAction is an example of a user action class which is derived from G4VUserPrimaryGeneratorAction. In this class the user must describe the initial state of the primary event. This class has a public virtual method named GeneratePrimaries( ) which will be invoked at the beginning of each event[15]. Details will be given in the next section. Note that GEANT4 does not provide any default behavior for generating a primary event.

```
runManager->Initialize();
```

This performs the detector construction, creates the physics processes, calculates cross sections and otherwise sets up the run.

With the command "*/run/BeamOn*", the system will run the event with user-defined number of events.

3.2.2 System Layout

In APTOFDetectorConstruction file, see Figure 3.2, it has:

(1) Definition of Materials

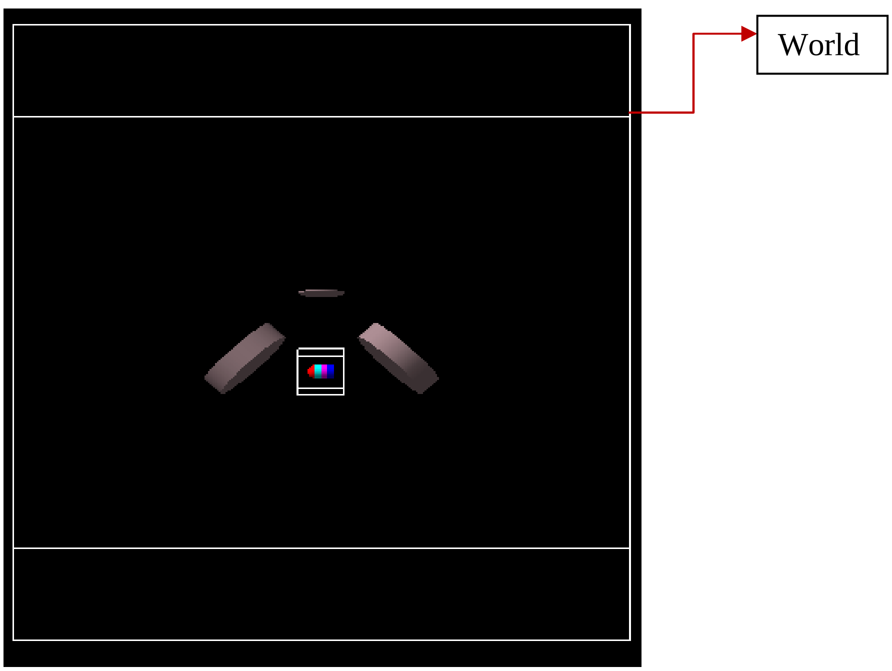
```
//--------- Definitions of Solids, Logical Volumes, Physical Volumes ---------

    //------------------------------
    // World
    //------------------------------

    G4double HalfWorldLength = 0.5*worldLength;

    solidWorld= new G4Box("world",HalfWorldLength,HalfWorldLength,HalfWorldLength);
    logicWorld= new G4LogicalVolume( solidWorld,/* Air*/ Vacuum, "World", 0, 0, 0);

//  G4Box * solidGround=new G4Box("ground",HalfWorldLength/2.,10*cm,HalfWorldLength/2.);
//  G4LogicalVolume *logicGround = new G4LogicalVolume(solidGround, Air, "Ground");

    //  Must place the World Physical volume unrotated at (0,0,0).
    //
    physiWorld = new G4PVPlacement(0,              // no rotation
                   G4ThreeVector(), // at (0,0,0)
                   logicWorld,        // its logical volume
                   "World",     // its name
                   0,           // its mother  volume
                   false,     // no boolean operations
                   0);             // copy number

    //------------------------------
    // Target
    //------------------------------

    G4ThreeVector positionTarget = G4ThreeVector(0,0,0);
    solidTarget = new G4Tubs("target",0,0.5*cm,0.1*cm,0,360*deg);
    G4Material* TargetMater = Vacuum;
    logicTarget = new G4LogicalVolume(solidTarget,TargetMater,"Target",0,0,0);

    physiTarget = new G4PVPlacement(0,//G4ThreeVector(0,1,-1),          // rotation
                    positionTarget, // at (x,y,z)
                    logicTarget,        // its logical volume
                    "Target",    // its name
                    logicWorld,     // its mother   volume
                    false,      // no boolean operations
                    0);         // copy number

    G4VisAttributes* targetAtt = new G4VisAttributes(G4Colour(146.0/256.0, 120./256.0, 125/256.0));
```

Figure 3.3: A snapshot of APTOF system layout source file

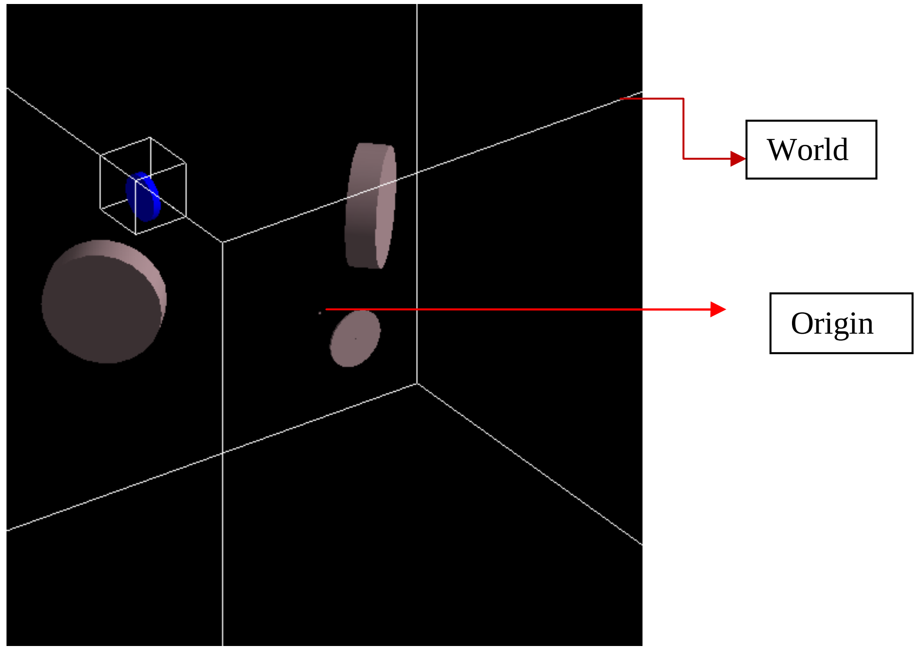
(2) World Volume, Solid Volume and Logical Volume

In this virtual sensing system the largest volume is called World volume. It must contain all other volumes in the detector geometry. To describe the volume's shape, we use the concept of a solid. A solid is a geometrical object that has a shape and specific value for each of that shape's dimensions[13]. To describe a volume full properties, we use a logical volume. It includes the geometrical properties, material of the solid, and adds physical characteristics: the material of the volume; whether is contains any sensitive detector elements; the magnetic field; etc.

We defined world volume "world" to be a cubic box with 800cm side length, where everything else of the system is placed inside it, see Figure 3.4. It is made of vacuum and has no practical use other than system configuration. The center of the box is placed at the origin of APTOF system's coordinate.



(a)



(b)

Figure 3.4 (a) (b): G4 World volume

(2) Object/UXO and Its Environment

In GEANT4, the shape of UXO can vary according to user's definition. The placement, tile angle, rotation of UXO can also be defined if necessary. Besides the large box shown in Figure 3.4, there is a 27000cm^3 small box outside the UXO. In our G4 program, it is designed as the environment around the UXO, see Figure 3.5.

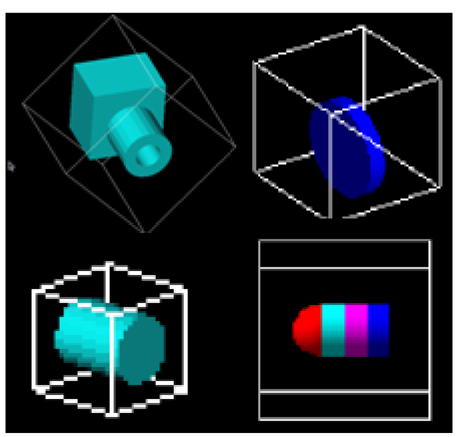
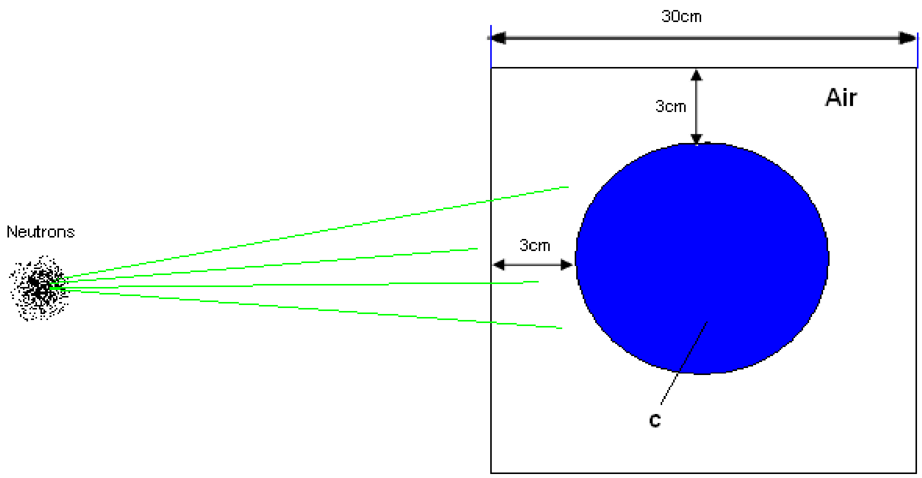


Figure 3.5. Examples of UXO in G4

To simulate UXO detecting processes, we need to define the environment around the UXO. It can be Air, which is mainly made of Nitrogen and Oxygen. Or it can be soil(SiO_2), where density varies between 1 and 2 g/cm³, see Figure 3.6.



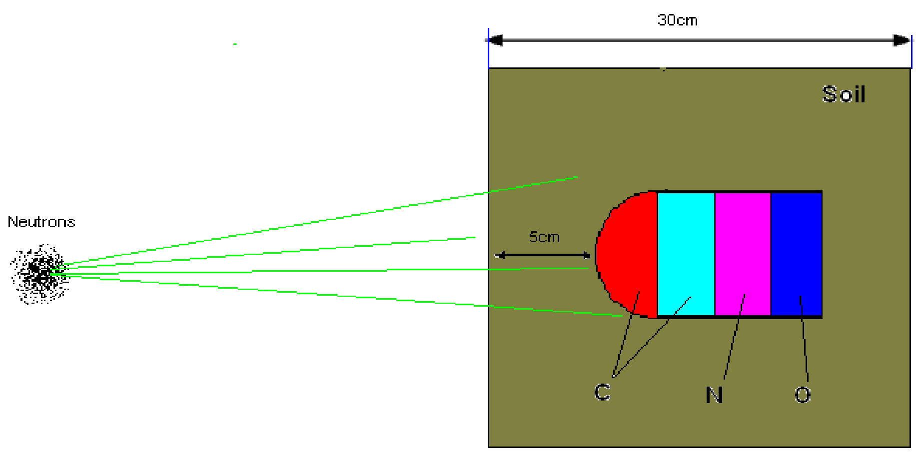


Figure 3.6: Examples of environment surround UXO

(3) Detector Construction.

For detector construction, we defined two types of detectors, alpha detector and gamma detectors, see Figure 3.7. Each of them will receive alpha particles and gamma rays respectively.

```
//the senesitive detector
APTOFSensitiveDetector *gammaSD = new APTOFSensitiveDetector("gamma");
APTOFSensitiveDetector *alphaSD = new APTOFSensitiveDetector("alpha");

G4SDManager::GetSDMpointer()->AddNewDetector(gammaSD);
G4SDManager::GetSDMpointer()->AddNewDetector(alphaSD);


G4double radius_scin=30*cm;//diameter=12.7cm
G4double length_scin=15.24*cm;

G4Tubs *solidgammaScin = new G4Tubs("solidgammaScin",0,radius_scin,length_scin/2.,0,360*deg);
G4LogicalVolume* logicgammaScin = new G4LogicalVolume(solidgammaScin, NaI, "logicgammaScin");

radius_scin=15*cm;
length_scin=1*cm;

G4Tubs *solidalphaScin = new G4Tubs("solidalphaScin",2,radius_scin,length_scin/2.,0,360*deg);
G4LogicalVolume* logicalphaScin = new G4LogicalVolume(solidalphaScin, Pb, "logicalphaScin");

logicgammaScin->SetSensitiveDetector(gammaSD);
logicalphaScin->SetSensitiveDetector(alphaSD);
```

Figure 3.7: A snapshot of APTOF detectors construction source file

The configurations of APTOF alpha detector and gamma detectors are as below, Table 3.1.

Table 3.1: Configurations of APTOF alpha detector and gamma detectors

Alpha Detector	Shape	Radius/cm	Thickness/cm	Position(x,y,z)	Color	Number
	Plate	15	1	(0, 0, 30)	Brown	1

Gamma Detector	Shape	Radius/cm	Thickness/cm	Position(x,y,z)	Color	Number
	Plate	30	15.24	(50,0,30); (-50,0,30)	Brown	2

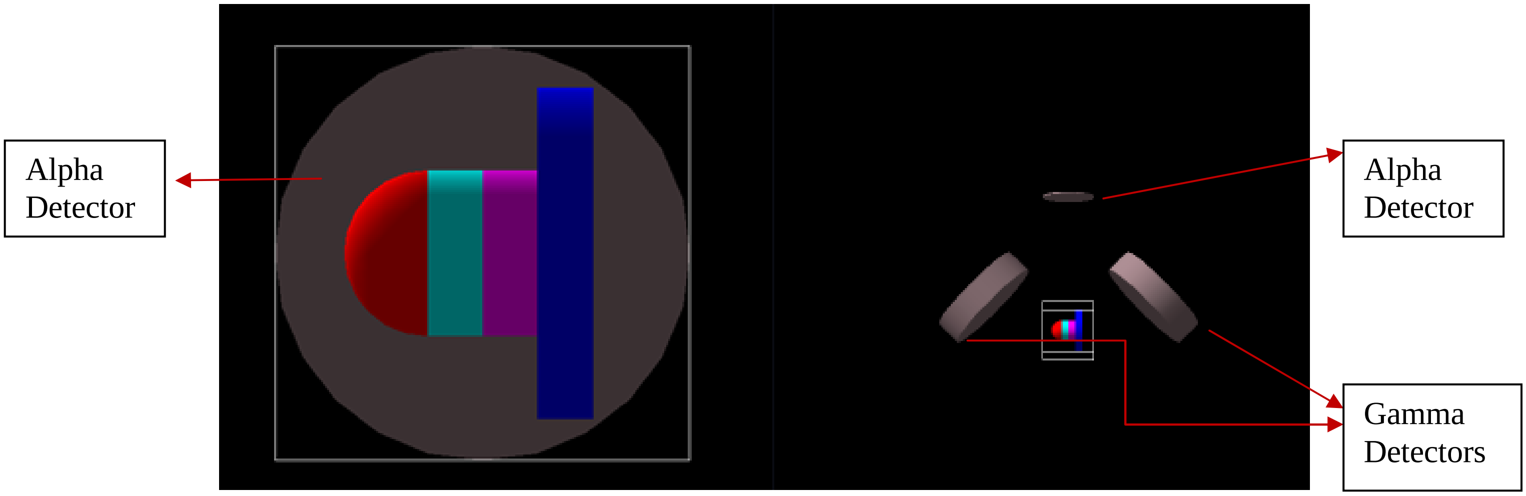


Figure 3.8: Front-view and top-view of APTOF detectors

As shown from Fig.3.8, the center of the alpha detector lies on the z axis, with 30cm distance from the world origin. In order to increase the efficiency to capture the emitting gamma rays, we placed two gamma detectors with their axis, which coincide with the world z axis in default, rotated 45°, -45° to the world y axis, respectively.

(5) All particles and physics processes required in a simulation must be registered.

The user must create a class derived from G4PhysicsList, see Figure 3.9. In our program, we registered bosons, leptons, mesons, barons, ions, which are provided by G4 by default. Each one of them is represented by its own class, which is derived from *G4ParticleDefinition.* Also we registered necessary processes such as: Electric and Magnetic Processes, hadronic processes, photonuclear processes, and general processes. Then we assign them to right particles. The LHEP Physics lists we are using are based on a parametrised modeling for all hadronic interactions for all particles. The parametrised model is an improved version of the Gheisha model. These lists combine the high energy parameterised (HEP) and low energy parameterised (LEP) models describing inelastic interactions for all hadrons. The modeling of elastic scattering off a nucleus and of capture of negative stopped particles and neutrons proceeds via parameterised models. Cross sections used are based on Gheisha parameterisations. To be more specific, in LHEP, we are using LHEP_PRECO_HP. This is a physics list for low energy dosimetric applications with neutrons. It uses a pre-equilibrium decay model for modeling the inelastic interactions of neutrons (and protons). It uses the Wellisch-Axen systematics for cross-section calculation of nucleon nuclear reaction cross-sections in the giant resonance region. Point-wise evaluated cross-section data are used to model neutron interactions from thermal energies to ~20MeV. This applies to capture, elastic scattering, fission and inelastic scattering. Please use together with G4NDL3.5 or higher. Note that doppler broadening is done automatically on the fly to the temperature specified for the local material. No pre-processing of the data is necessary[14].

```
void APTOFPhysicsList::ConstructEM()
{
    //the particle, it's process maneger and it's name
        G4ParticleDefinition* particle;
        G4ProcessManager* pmanager;
        G4String particleName;


    //reset the particle iterator so we can use it in the loop
    //the particle iterator is a global variable of G4VUserAPTOFPhysicsList
    theParticleIterator->reset();

    //iterate through all particles
        while( (*theParticleIterator)() )
    {

     //gets the particle's name and process manager
         particle = theParticleIterator->value();
         pmanager = particle->GetProcessManager();
         particleName = particle->GetParticleName();

     //gamma EM processes
         if (particleName == "gamma")
     {
        //standard processes: pair production, compton scattering and photoelectric effect
        G4GammaConversion * gammaConversion = new G4GammaConversion();
        G4ComptonScattering * comptonScattering = new G4ComptonScattering();
        G4PhotoElectricEffect * photoElectricEffect = new G4PhotoElectricEffect();

            //add processes
            pmanager->AddDiscreteProcess(gammaConversion);
        pmanager->AddDiscreteProcess(comptonScattering);
            pmanager->AddDiscreteProcess(photoElectricEffect);

     }
```

```
//gets the process manager for the neutron
    G4ProcessManager * neutronProcMan = G4Neutron::Neutron()->GetProcessManager();

G4cout << "APTOFPhysicsList: Loading high precision neutron data: Elastic scattering\n";
G4cout.flush();

//neutron-nucleus elastic scattering: High precision (from data sets)
G4HadronElasticProcess  * theNeutronElasticProcess  = new G4HadronElasticProcess();
G4LElastic      * theNeutronElastic         = new G4LElastic();
G4NeutronHPElastic  * theNeutronHPElastic       = new G4NeutronHPElastic();
G4NeutronHPElasticData  * theNeutronHPElasticData   = new G4NeutronHPElasticData();
    theNeutronHPElastic -> SetMinEnergy( 0.0*MeV);
    theNeutronHPElastic -> SetMaxEnergy(20.0*MeV);
theNeutronElastic    -> SetMinEnergy(19.9*MeV);
theNeutronElasticProcess->AddDataSet(theNeutronHPElasticData);
theNeutronElasticProcess->RegisterMe(theNeutronElastic);
theNeutronElasticProcess->RegisterMe(theNeutronHPElastic);
neutronProcMan->AddDiscreteProcess(theNeutronElasticProcess);

G4cout << "APTOFPhysicsList: Loading high precision neutron data: Inelastic scattering\n";
G4cout.flush();

//inelastic interactions: High precision (from data sets)
//High precision datasets from 0 to 20 MeV
//Pre-compound (PRECO) used for above 20MeV to 170 MeV
//Low-energy parameterised from 150 MeV to 55 GeV
//High-energy prameterised from 45 GeV and beyond
G4NeutronInelasticProcess * theNeutronInelasticProcess  = new G4NeutronInelasticProcess();
G4LENeutronInelastic      * theLENeutronInelastic    = new G4LENeutronInelastic();
G4HENeutronInelastic      * theHENeutronInelastic    = new G4HENeutronInelastic();
G4NeutronHPInelastic      * theNeutronHPInelastic    = new G4NeutronHPInelastic();
G4NeutronHPInelasticData  * theNeutronHPInelasticData   = new G4NeutronHPInelasticData();
    G4ExcitationHandler       * theNeutronExcitationHandler = new G4ExcitationHandler();
    G4PreCompoundModel    * theNeutronPRECOModel    = new G4PreCompoundModel(theNeutronExcitationHandler);
    G4NeutronInelasticCrossSection * theNeutronICS      = new G4NeutronInelasticCrossSection(); //for G4Pre(
    theNeutronHPInelastic -> SetMinEnergy(  0.0*MeV);
    theNeutronHPInelastic -> SetMaxEnergy( 20.0*MeV);
theNeutronPRECOModel  -> SetMinEnergy( 19.9*MeV);
theNeutronPRECOModel  -> SetMaxEnergy(170.0*MeV);
    theLENeutronInelastic -> SetMinEnergy(150.0*MeV);
    theLENeutronInelastic -> SetMaxEnergy( 55.0*GeV);
    theHENeutronInelastic -> SetMinEnergy( 45.0*GeV);
theNeutronInelasticProcess->AddDataSet(theNeutronHPInelasticData);
theNeutronInelasticProcess->AddDataSet(theNeutronICS);
theNeutronInelasticProcess->RegisterMe(theNeutronHPInelastic);
theNeutronInelasticProcess->RegisterMe(theNeutronPRECOModel);
theNeutronInelasticProcess->RegisterMe(theLENeutronInelastic);
theNeutronInelasticProcess->RegisterMe(theHENeutronInelastic);
neutronProcMan->AddDiscreteProcess(theNeutronInelasticProcess);
```

Figure 3.9: A snapshot of APTOF physics list source file

3.2.3 Primary Event

For a desired physics process to occur, the first thing is to generate the primary event. For each event, user must define all details of initial particles, see Fig.3.10.

```
APTOFPrimaryGeneratorAction::APTOFPrimaryGeneratorAction(){
    G4int n_particle = 1;
    neutronGun = new G4ParticleGun(n_particle);
    alphaGun   = new G4ParticleGun(n_particle);
    inf = new ifstream("dt.txt");

    neutronGun->SetParticleDefinition(G4ParticleTable::GetParticleTable()->FindParticle("neutron"));
    alphaGun  ->SetParticleDefinition(G4ParticleTable::GetParticleTable()->FindParticle("alpha"));
}

//....oooOO0OOooo........oooOO0OOooo........oooOO0OOooo........oooOO0OOooo......

APTOFPrimaryGeneratorAction::~APTOFPrimaryGeneratorAction()
{
    delete neutronGun;
    delete alphaGun;
    delete inf;
}

void APTOFPrimaryGeneratorAction::GeneratePrimaries(G4Event* anEvent)
{
    double EN,EA,thetaN,thetaA,phiN,phiA;
    if(*inf>>EN>>thetaN>>phiN>>EA>>thetaA>>phiA){
    G4ThreeVector pos(0,0,0);
    neutronGun->SetParticlePosition(pos);
    alphaGun->SetParticlePosition(pos);
    neutronGun->SetParticleEnergy(EN*MeV);
    alphaGun->SetParticleEnergy(EA*MeV);

    cout<<EN<<"\t"<<thetaN<<"\t"<<phiN<<"\t"<<EA<<"\t"<<thetaA<<"\t"<<phiA<<endl;


    phiN *=deg;
    thetaN *=deg;
    phiA *=deg;
    thetaA *=deg;

    G4ThreeVector dirN(sin(thetaN)*cos(phiN),sin(thetaN)*sin(phiN),cos(thetaN));
    G4ThreeVector dirA(sin(thetaA)*cos(phiA),sin(thetaA)*sin(phiA),cos(thetaA));

    neutronGun->SetParticleMomentumDirection(dirN);
    neutronGun->GeneratePrimaryVertex(anEvent);

    alphaGun->SetParticleMomentumDirection(dirA);
    alphaGun->GeneratePrimaryVertex(anEvent);
```

Figure 3.10: A snapshot of APTOF primary event source file

In the primary generation file,

```
"APTOFPrimaryGeneratorAction.cc"
APTOFPrimaryGeneratorAction::APTOFPrimaryGeneratorAction(){
        G4int n_particle = 1;
        neutronGun = new G4ParticleGun(n_particle);
        alphaGun   = new G4ParticleGun(n_particle);

        neutronGun->SetParticleDefinition(G4ParticleTable::GetParticleTable()-
>FindParticle("neutron"));
        alphaGun  ->SetParticleDefinition(G4ParticleTable::GetParticleTable()->FindParticle("alpha"));
}
```

……

Here we set up neutron gun and alpha gun, each of them will produce initial particles-neutrons and alpha particles in opposite direction with pre-defined emitting direction and energy. For each run, one pair of neutron and alpha particle will be produced at one time, and repeat itself till the last run ends.

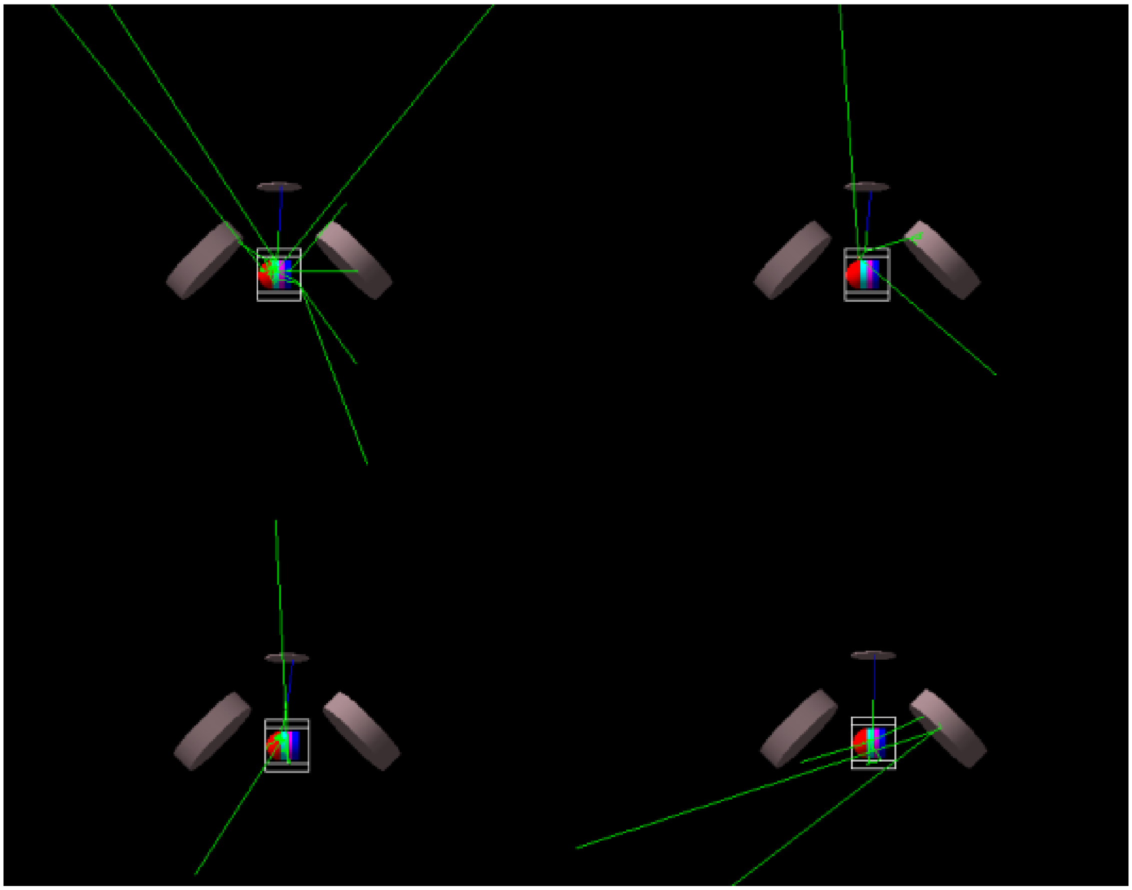


Figure 3.11: APTOF primary event

The direction and energy of alpha particle and neutron is defined by the previous calculations based on Eqn. 3.2, 3.3, 3.4. In Figure 3.11, the blue beam represents the trajectory of alpha, and the green beam stands for the neutron's trajectory.

3.2.4 Data Collection

When we define *APTOFDetectorConstruction* file, we also define sensitive detector for each detector. When we create the logical volume in the detectors we can describe its sensitive detector. Once a particle gets into the detectors, the SensitiveDetector will be triggered and it records all the necessary hit information such as position and time. Hit is a snapshot of the physical interaction of a track or an accumulation of interactions of

tracks in the sensitive region of the detector.

```
//constructor
APTOFHit::APTOFHit(){
    //Initially sets everything to zero
    edep = 0.0;
    time = 0.0;
    deltaTime = 0.0;
    trackID = -1;
    particle = "";
    kineticEnergy = 0.;
    localPrePos  = G4ThreeVector(0.0,0.0,0.0);
    localPostPos = G4ThreeVector(0.0,0.0,0.0);
}

//print the information about the hit
void APTOFHit::Print(std::ostream & os)
{
    //Displays ringNo/armNo in human convinient numering
    os << particle << " (" << trackID << "); ";
    os << "Deposited " << G4BestUnit(edep,"Energy") << "; ";
    os << "Time " << G4BestUnit(time,"Time") << "; ";
    os << "Length " << G4BestUnit(deltaTime,"Time") << ";\n";
}
```

Figure 3.12: A snapshot of APTOF hit source file

In the *APTOFHit* file, we initialized each parameter such as deposited energy, time for a step, energy at the start of this step and track ID to be zero. For each hit, we can record its name of the particle, energy deposited, time at the beginning of the step, amount of time for the step, the identification number of the track, as well as pre and post-step hit positions in the coordinates of the detector, and pre and post-step hit positions in the coordinates of the world.

```
void APTOFEventAction::EndOfEventAction(const G4Event* evt){

    G4int event_id = evt->GetEventID();
    // periodic printing
    if (event_id < 100 || event_id%10000 == 0)
        G4cout << ">>> Event " << evt->GetEventID() << G4endl;

    G4int CHCIDalpha = G4SDManager::GetSDMpointer()->GetCollectionID("alpha/HitCollection");

    APTOFHitsCollection * cellHitsCollectionalpha = (APTOFHitsCollection *) (evt->GetHCofThisEvent()->GetHC(CHCIDalpha));
    G4int cnHitalpha = cellHitsCollectionalpha->entries();

    G4int CHCIDgamma = G4SDManager::GetSDMpointer()->GetCollectionID("gamma/HitCollection");

    APTOFHitsCollection * cellHitsCollectiongamma = (APTOFHitsCollection *) (evt->GetHCofThisEvent()->GetHC(CHCIDgamma));
    G4int cnHitgamma = cellHitsCollectiongamma->entries();


    if(cnHitalpha>0 && cnHitgamma>0){

        G4ThreeVector v2 = (*cellHitsCollectiongamma)[0]->GetPreGlobalPosition();
        G4ThreeVector v1 = (*cellHitsCollectionalpha)[0]->GetPreGlobalPosition();

        os  <<evt->GetEventID()+1
            <<"\t"<<(*cellHitsCollectionalpha)[0]->GetParticle()
            <<"\t"<<(*cellHitsCollectionalpha)[0]->GetKineticEnergy()/MeV
            <<"\t"<<v1.x()/cm
            <<"\t"<<v1.y()/cm
            <<"\t"<<v1.z()/cm
            <<"\t"<<(*cellHitsCollectionalpha)[0]->GetInitTime()

            <<"\t"<<(*cellHitsCollectiongamma)[0]->GetParticle()
            <<"\t"<<(*cellHitsCollectiongamma)[0]->GetKineticEnergy()/MeV
            <<"\t"<<v2.x()/cm
            <<"\t"<<v2.y()/cm
            <<"\t"<<v2.z()/cm
            <<"\t"<<(*cellHitsCollectiongamma)[0]->GetInitTime()
            <<G4endl;
```

Figure 3.13: A snapshot of APTOF event action source file

Therefore, in the output file we have the track ID, particle name, x, y, z position, energy and time of filght for gamma, neutron and alpha in one data set. This will be used for data analysis in ROOT software later.

3.2.5 Visualization[13]

The GEANT4 visualization system was developed in response to a diverse set of requirements:

(1) Quick response to study geometries, trajectories and hits

(2) High-quality output for publications

(3) Flexible camera control to debug complex geometries

(4) Tools to show volume overlap errors in detector geometries

(5) Interactive picking to get more information on visualized object Simulation data can be visualized:

- Detector components
- A hierarchical structure of physical volumes
- A piece of physical volume, logical volume, and solid
- Particle trajectories and tracking steps
- Hits of particles in detector components

You have a choice of visualization drivers. The graphics system we are using to support GEANT4 is OpenGL. It is well suited for real-time fast visualization and demonstration. Put in commands to visualize G4 program.

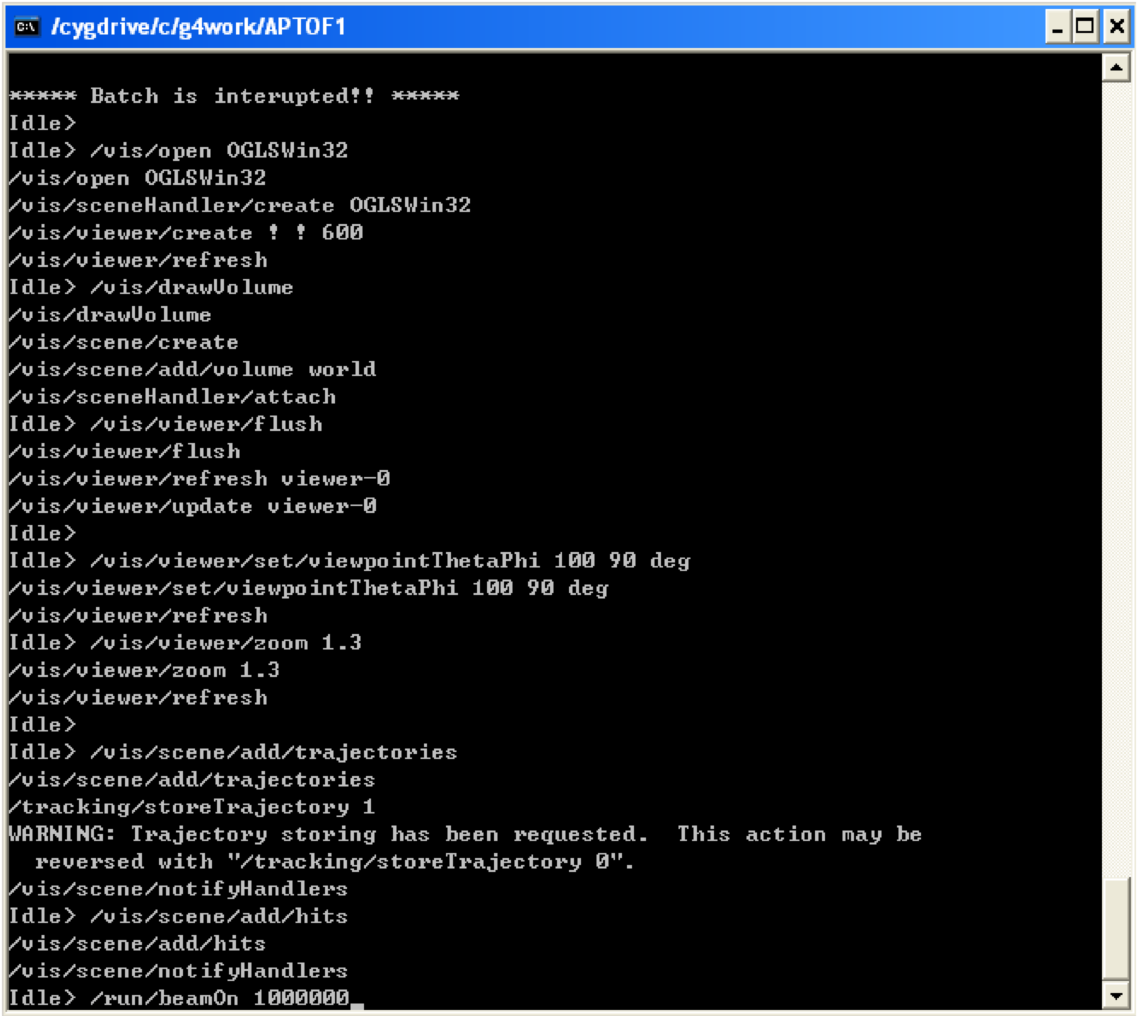


Figure 3.14: User Interface for GEANT4 program

3.2.6 Run

The largest unit of a simulation application is the run, which consists in terms of events, tracks, and steps. At the beginning of run, geometry is optimized for navigation. Within a run, the detector geometry, the set up of sensitive detectors, and the physics processes used in the simulation should be kept unchanged, see Figure 3.15. A run is represented by a *G4Run* class object. A run starts with BeamOn() method of *G4RunManager*.

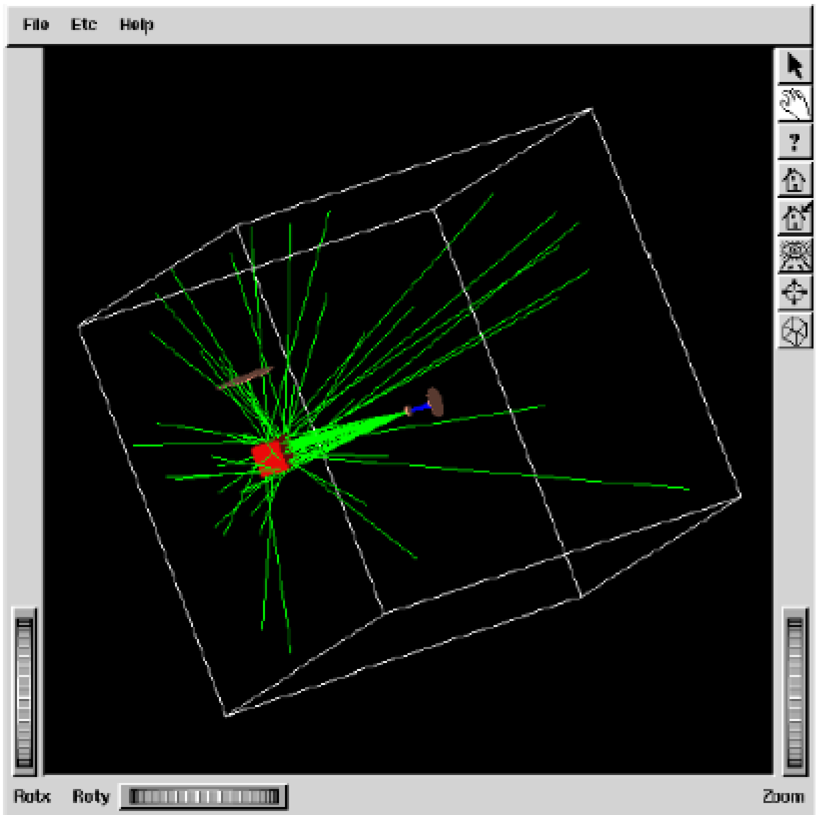


Figure 3.15: A G4 display with neutrons and alpha particles

Chapter 4

4. Data Analysis (APTOF Imaging)

4.1 Introduction

APTOF technique can be also called Associated Particle Imaging. It is an active neutron probe technique that provides a 3D image with elemental composition of the material under interrogation. API uses the direction and time correlation between the 14MeV neutron and the associated alpha particle produced by the D+T reaction. Detection of the alpha particle with a position sensitive detector provides direction and time of emission of the neutron. The neutron may then interact with the target nucleus to produce a gamma-ray whose energy is characteristic of the target. In a word, it follows the neutron in-gamma out approach.

API permits single-sided, non-intrusive inspection of the internal contents of sealed packages and containers, and will image all elements except hydrogen and helium. The technique images 14MeV neutron interactions in the material of interest[15].

The traditional neutron interrogation approach has been used for decades, however, it has some severe limitations. Gamma rays resulting from surrounding clutter, such as, the ground, structural components, items intended to hide the threat, the barrier material, etc., may all be sources of background "noise" in this approach[16]. The API method adds one important additional feature to the traditional method. This improvement relies on “tagging” the individual neutrons used in the interrogation so that only those gamma rays that are produced in the region of interest that contains the threat material are counted. The result is a greatly improved signal-to-noise ratio for API compared to other neutron in-gamma out devices[17][18].

The key API technical capabilities are[19]:

- Able to penetrate barriers concealing the threat material, including steel and concrete
- Provides material identification and/or classification of hidden threats
- Provides 3D mapping of the contents of sealed containers
- Can be used only one side of the container is available

4.1Kinematics

By capturing the alpha particle and noting both the position of interaction on the scintillator screen and the time of the event, information about the location of the associated neutron can be determined. Both cone and azimuthal (θ, φ) angles of the neutron trajectory may be determined.

Table 4.1: List of particles

	Energy(MeV)	Mass(MeV/c^2)	Speed(cm/ns)
Gamma	/	/	V_{gamma}=29.9792458
Alpha particle	~3.5	3737.37911	Va=1.29826
Neutron	~14.1	939.56536	Vn=5.13613

Here, $Vn = \left\{ 1 - \frac{1}{\left(\frac{m_n + E_n}{m_n} \right)^2} \right\}^{1/2} \times V_{light}$

From Geant4 we can record the energy, x,y,z position information and time of flight for both alpha particle and gamma-ray. Each of them are written as E_a, x_a, y_a, z_a, t_a, E_g, x_g, y_g, z_g, t_g.

One thing should be noted that t_a represents the time elapsed from once the particle gun emits alpha till it hits the alpha detector. t_g stands for the time once the primary events start till gamma-ray is generated and hits the gamma detector. d_n stands for the distance from the source origin to the target of interest. d_a is the distance from the source origin to the alpha detector. d_g is the distance from the target of interest to the gamma detector.

Based on the correlations between alpha particle and neutron[16], see Table 4.1 and Eqn. 4, we developed the APTOF algorithm in C++ to re-generate 2D and 3D images of UXO.

$$\left.\begin{aligned} T_{total} &= \frac{d_n}{V_n} + \frac{d_\gamma}{V_{light}} \\ d_n &= \sqrt{{x_n}^2 + {y_n}^2 + {z_n}^2} \\ d_\gamma &= \sqrt{(x_g - x_n)^2 + (y_g - y_n)^2 + (z_g - z_n)^2} \\ d_\alpha &= \sqrt{{x_a}^2 + {y_a}^2 + {z_a}^2} \end{aligned}\right\} \quad \text{Eqn.4}$$

$$\frac{x_n}{x_a} = \frac{y_n}{y_a} = \frac{z_n}{z_a} = \frac{d_n}{d_\alpha}$$

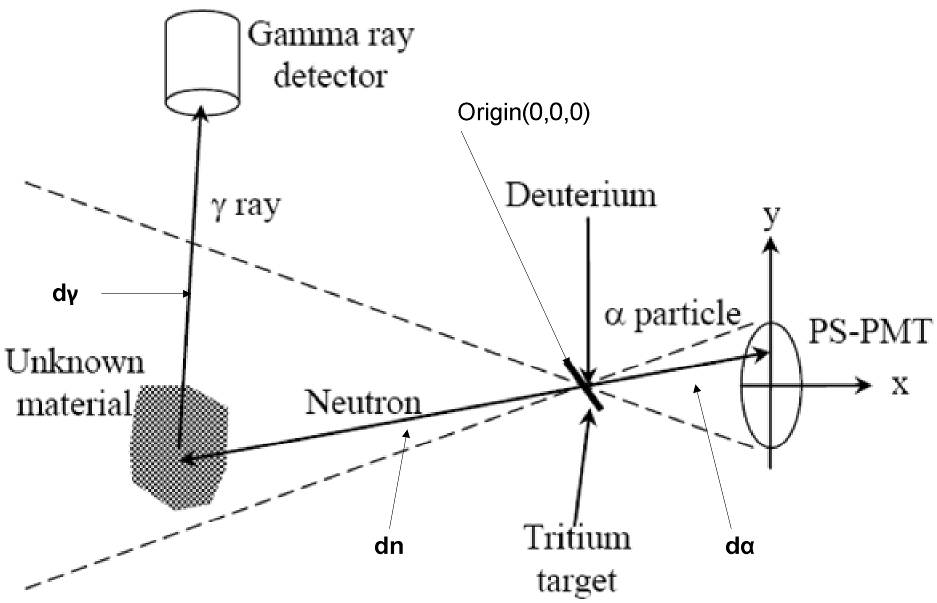


Figure 4.1: Correlation between neutron and alpha particle

The correlations between neutron and alpha particle are also shown in Fig. 4.1, 4.2.

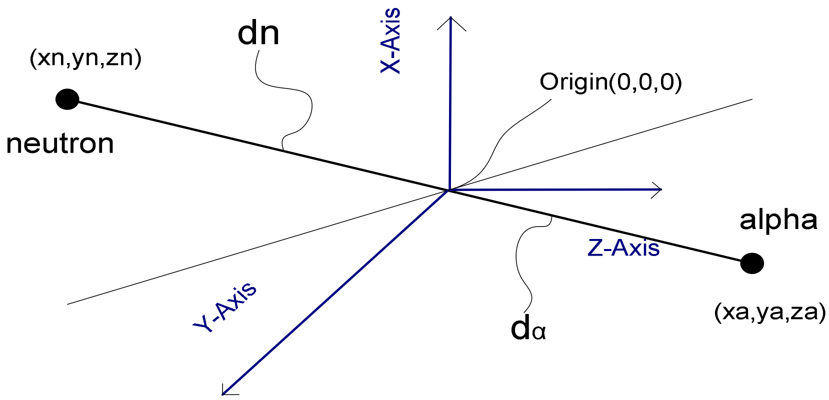


Figure 4.2: Correlation between neutron and alpha particle

From the five equations in Eqn. 4, knowing Ttotal, x_a, y_a, z_a, x_g, y_g, z_g, v_n, and speed of gamma, d can be calculated. This distance function is called the Euclidean metric. It can be viewed as a form of the Pythagorean theorem. Position information x_n, y_n, z_n can therefore be obtained.

By obtaining each neutron's (x_n, y_n, z_n), we can plot out the UXO in 3D and 2D image. E_g is also plotted as gamma energy spectrum. Out of all the gamma rays produced in the sample due to inelastic scattering of a neutron with a target nucleus, only a small fraction will enter the NaI detector. Of these gammas entering the detector, only a fraction will deposit their energy into the detector through one or more interactions in the sodium iodide crystal. The total energy deposited per gamma is the sum of the deposited energy from each of these interactions. This information is used to generate a gamma emission energy spectrum of the object for elemental composition analysis.

Back to our G4 program, each event starts from the primary generator, where one neutron and one alpha are generated with proper energy and direction. Then the G4 will tracks an individual neutron to an interaction site. If gamma-ray is produced, G4 will also track it from the interaction site to the gamma-ray detector. Therefore, in each loop only one neutron should be generated, otherwise the system has no way to know which neutron produced the gamma ray. In other words, the information we obtained is uncorrelated.

Chapter 5

5. **Experiments and Simulation Results**

5.1 Simulation Results

Experiments have been made on samples of Aluminum, Carbon, Nitrogen, Oxygen, Lucite, Silicon, etc. in air environment, see Fig. 5.1.

```
//Air
density = 1.29*mg/cm3;
G4Material * Air = new G4Material("Air", density, ncomponents=2);
Air->AddElement(elN, fractionmass=0.7);
Air->AddElement(elO, fractionmass=0.3);
```

Figure 5.1: Air definition in G4

(1) Carbon in Air

The configuration of this experiment is described in Table 5.1.

Table 5.1: Configuration of Carbon in Air simulation

Material	Shape	Environment	Number of Neutrons generated
Carbon	Cylinder + Cubic	Air	10,000

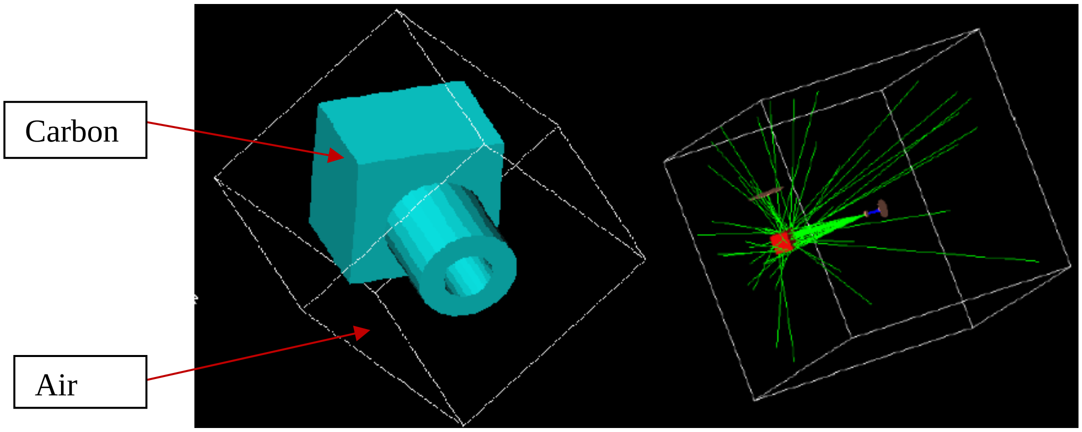


Figure 5.2 (a)

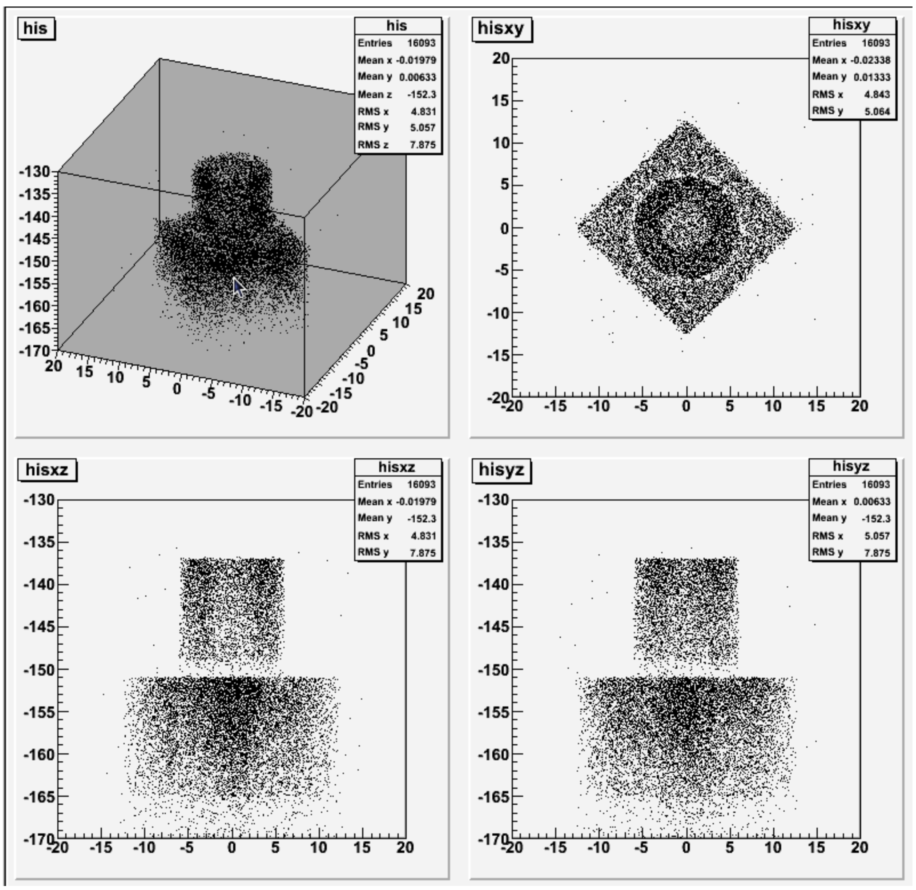


Figure 5.2 (b)

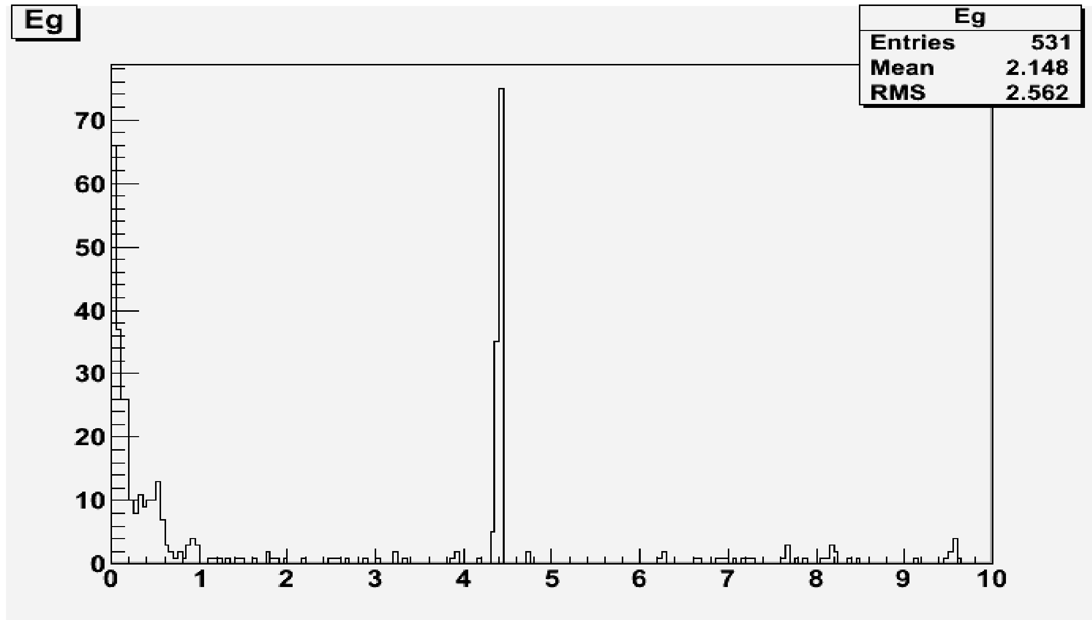


Figure 5.2 (c)

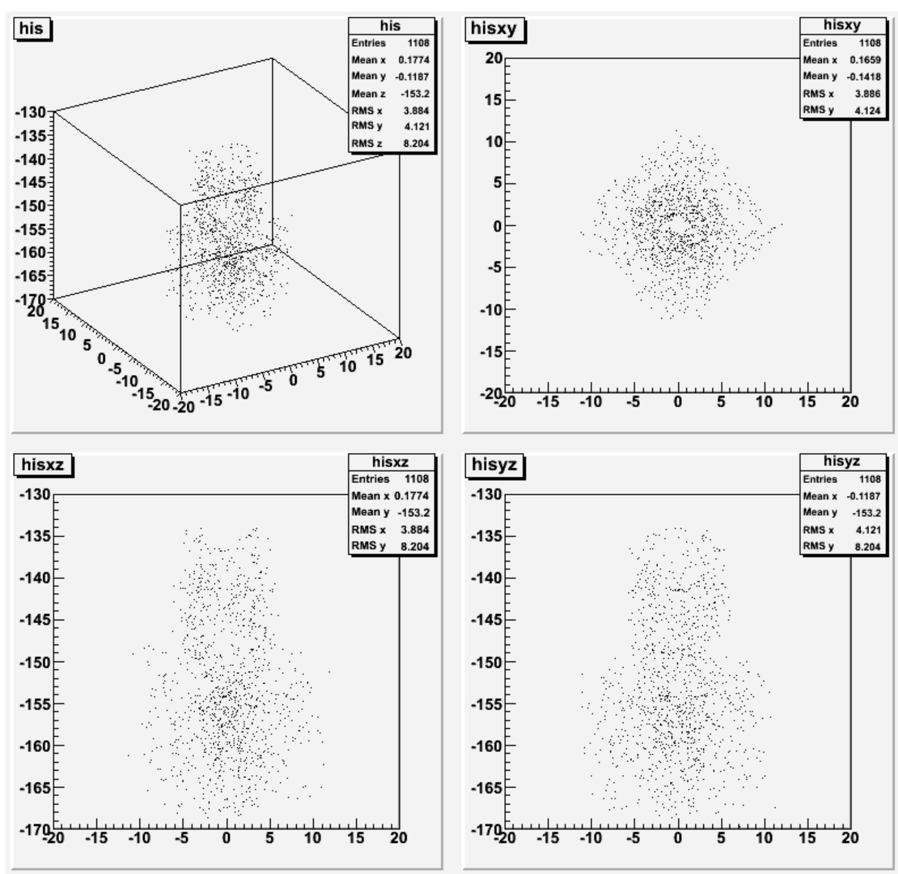


Figure 5.2 (d)

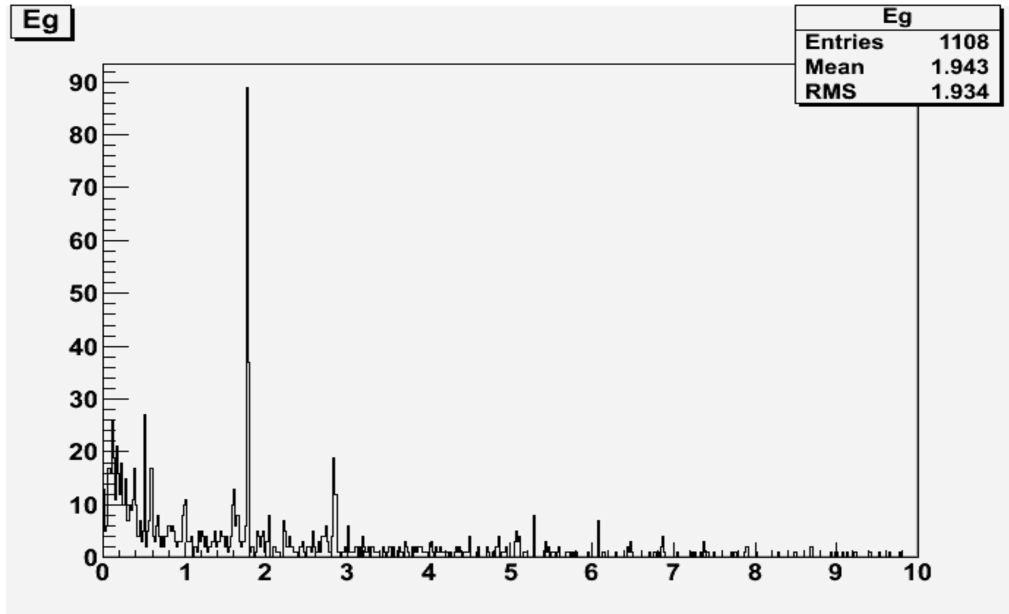


Figure 5.2 (e)

Figure 5.2: The outputs of APTOF system. (a) Sample made of Carbon under investigation in GEANT4. (b) 2D and 3D reconstruction of carbon-made object. (c) Gamma-ray spectrum of Carbon. (d) 2D and 3D reconstruction of Iron-made object. (e) Gamma-ray spectrum of Iron.

(2) To make the simulation more realistic, we also set the surrounding environment to be soil (SiO_2), see Fig. 5.3 and Table 5.2.

```
//Sio2
density = 1.2*g/cm3;
G4Material *Sio2 = new G4Material("Sio2", density, ncomponents=2);
Sio2->AddElement(elSi, natoms=1);
Sio2->AddElement(elO, natoms=2);
```

Figure 5.3 Soil definition in G4

Table 5.2: Configuration of Bomb in Soil simulation

Material	Shape	Environment	Number of Neutron beam generated
Carbon+Nitrogen+Oxygen+Iron	Cylinder+Hemi-sphere	SiO_2	1,000,000

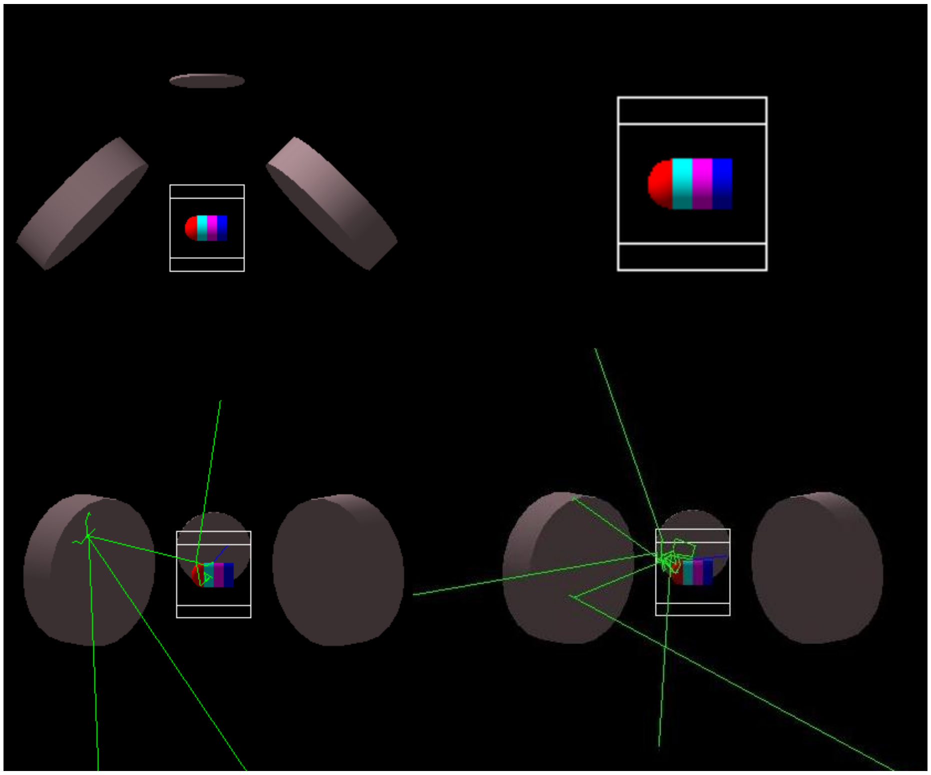


Figure 5.4 (a)

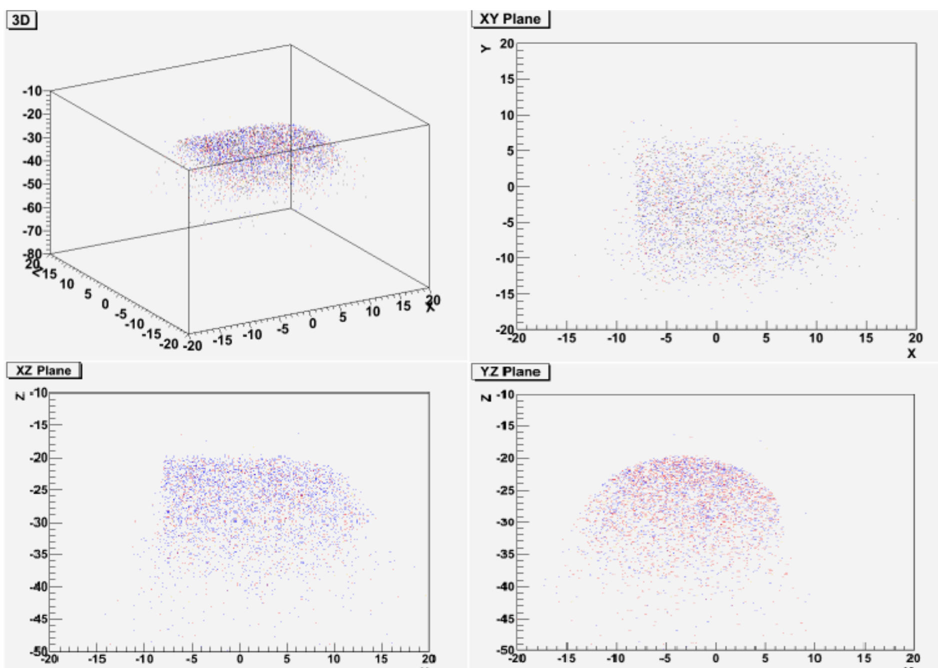


Figure5.4(b)

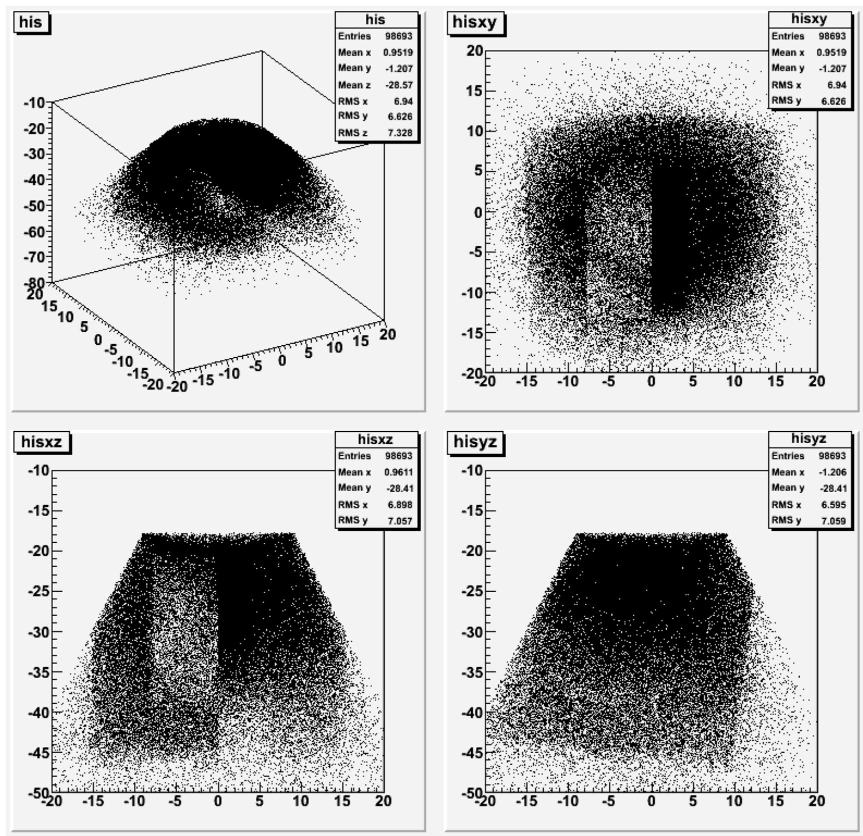


Figure 5.4(c)

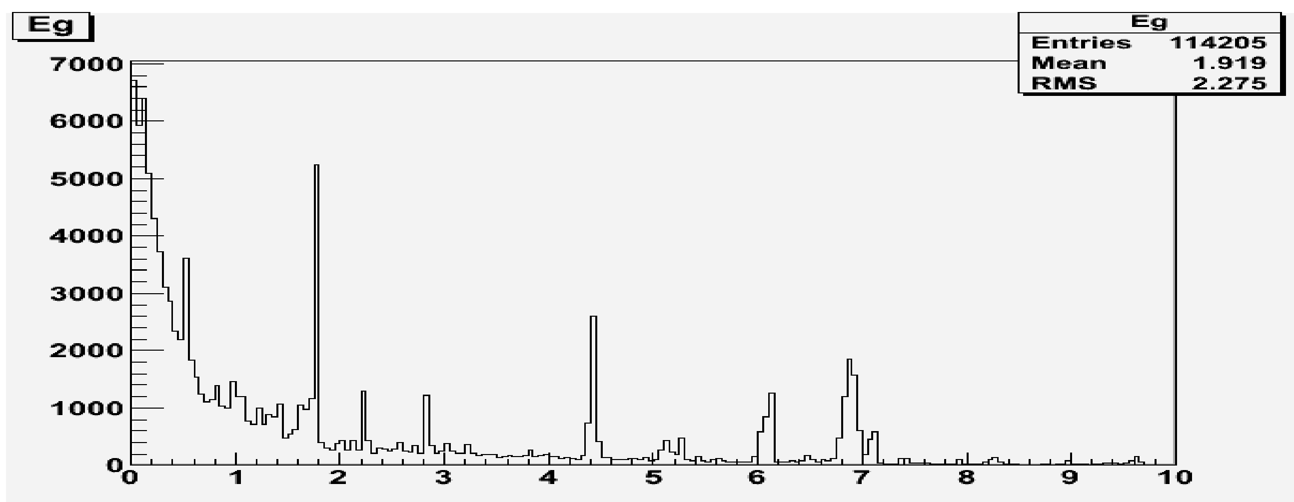


Figure 5.4 (d)

Figure 5.4: the outputs of APTOF system. (a) System running with object in soil. (b) 2D and 3D reconstruction of bomb in air. (c) 2D and 3D reconstruction of bomb in soil (d) Colorful Gamma-ray spectrum

(3) Simulation of Hexagen in soil with various depths

A bomb-shaped UXO was designed to be 10cm in radius and 16cm in length and is placed 30cm away from the neutron and alpha gun. The environment around the bomb is set to be air-a way to simulate condition when UXO is on surface. The elemental composition of the bomb is shown in Table 5.3:

Table 5.3: Composition of Hexagen

Name	Element	Mass Fraction
Hexagen ($C_3H_6O_6N_6$)	Carbon(C)	14.2%
	Hydrogen(H)	28.6%
	Oxygen(O)	28.6%
	Nitrogen(N)	28.6%

1 million pairs of neutron and alpha particle with direction and energy information were generated from MATLAB and were imported to the primary generator in GEANT4. UXO buried under 5cm, 10cm, 15cm, 20cm soil(SiO_2) was simulated to test the penetration ability of neutron. The cone angle of neutron flux is set to be within ±45˚, which the beam will cover the overall body of the bomb.

(i) 5cm-thick soil above the bomb

In this simulation, the bomb was placed inside a 5cm×5cm×5cm cubic box filled with soil(SiO_2). As we can see from Fig. 5.5, the surrounding soil absorbed a large amount of neutrons whereas the rest of the source went into the bomb. The yellow peak in Fig. 5.5 represents the signal of soil. We are expecting the soil peak to increase as the thickness of soil increases, and at the same time, the peaks representing other elements will also become weaker.

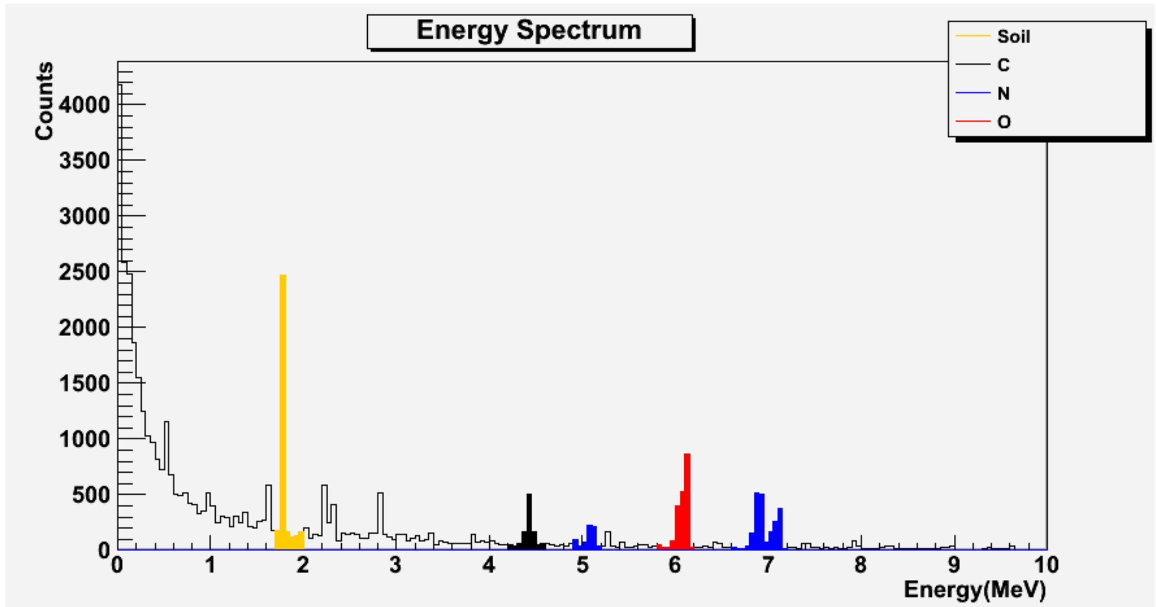


Figure 5.5: Energy spectrum of bomb buried 5cm in soil

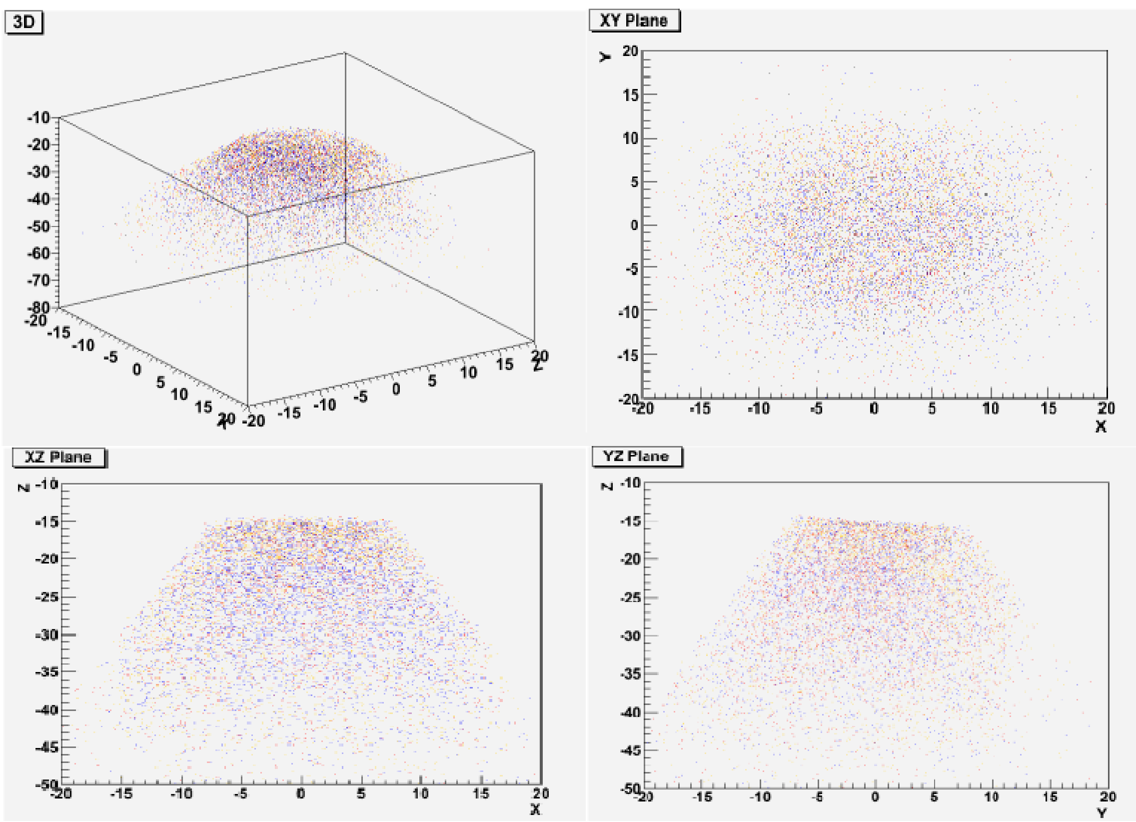


Figure: 5.6: 3D and 2D image reconstruction of bomb buried 5cm in soil

(ii) 10cm-thick soil above the bomb, see Fig. 5.7 and Fig. 5.8.

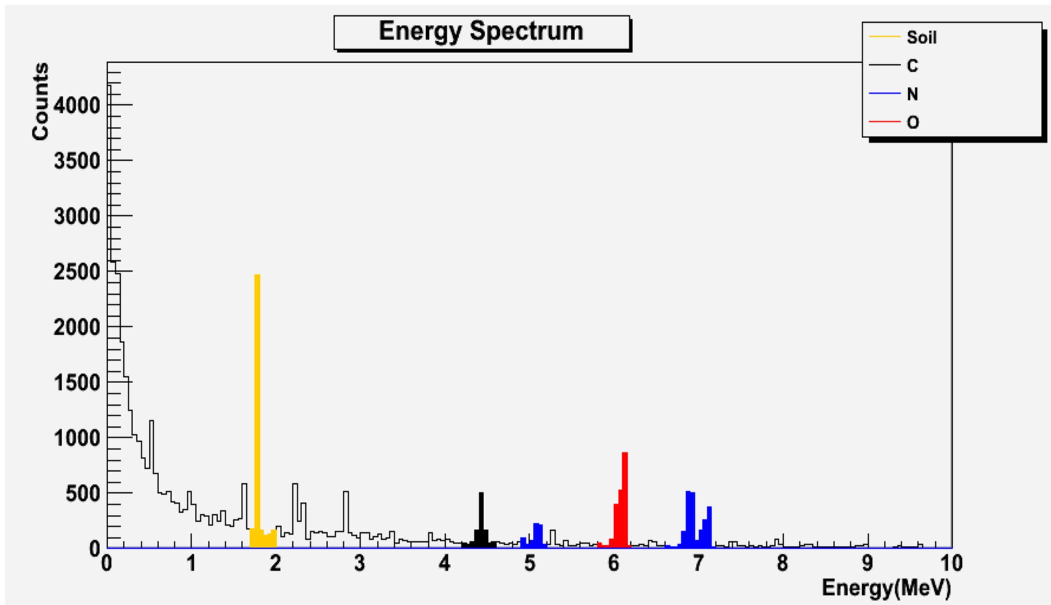


Figure 5.7: Energy spectrum of bomb buried 5cm in soil

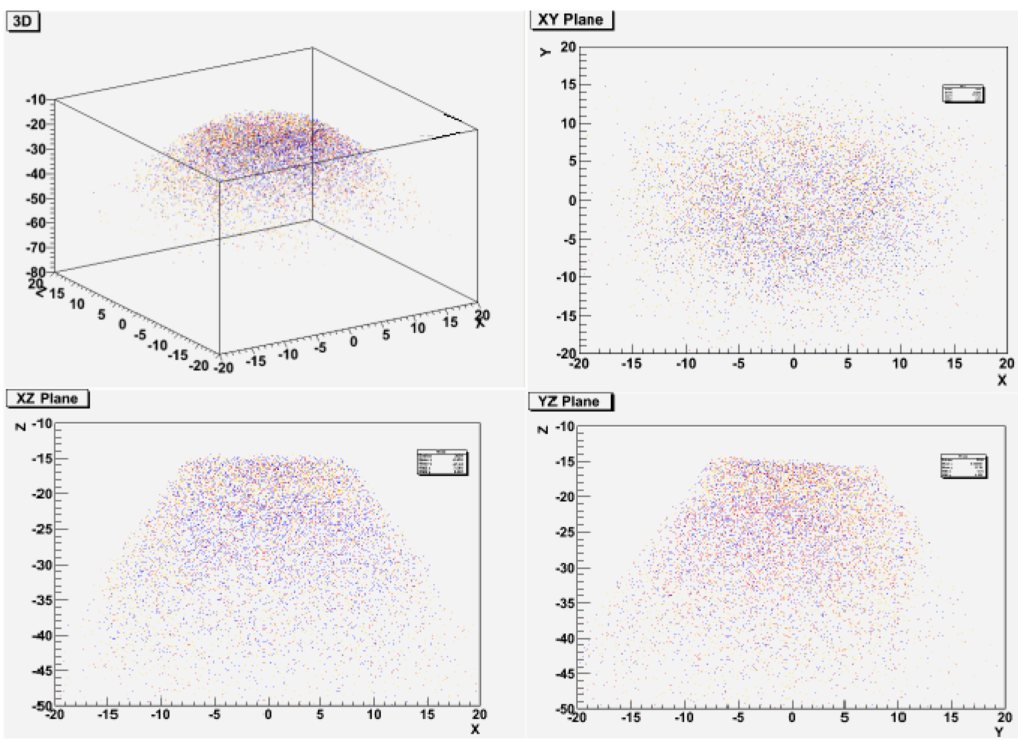


Figure 5.8: 3D and 2D image reconstruction of bomb buried 10cm in soil

Also we have buried the bomb under 15cm, 20cm, and 25cm thick soil depth. Fig. 5.9 shows us the intensity decay versus soil depths. and its error.

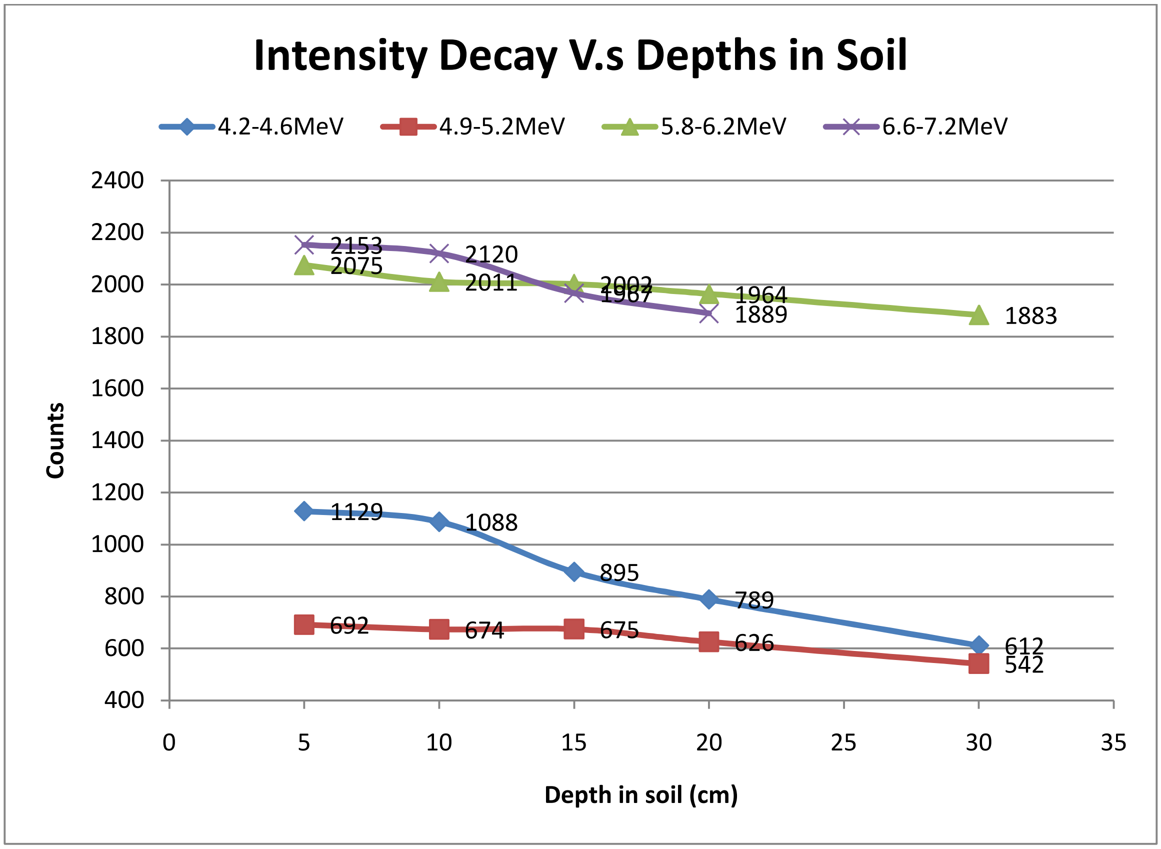


Figure 5.9: Intensity Decay V.s Depths in Soil and error bar

As we can see from the decay curves, with the depth in soil increases, each energy peak representing C, N, O decreases till detector can hardly capture any signals.

5.2 Time Window

Comparing soil peak (yellow) with other peaks, soil signal is over four times stronger than signals from the UXO. In order to attenuate the signal from the soil while increase the signal-to-noise ratio, we used probability analysis to create a time window to signify the signal from the area of the UXO.

Firstly, we put a 5cm-thick piece of carbon in front of the ‘bomb-area’ see Fig. 5.10, to create the first time window, see Fig 5.12. Then we remove the front piece, putting it at the bottom of the box, right behind the ‘bomb-area’ see Fig. 5.11. This is our second time window, see Fig. 5.13.

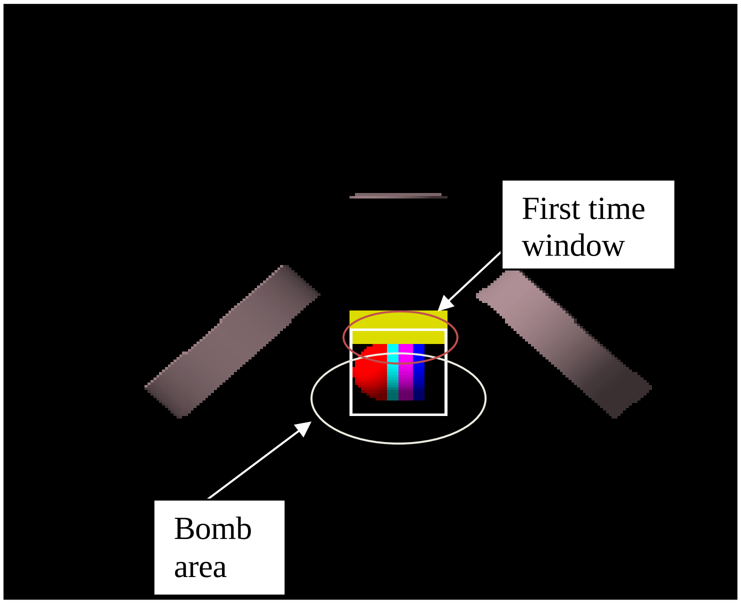


Figure 5.10: First time window simulation

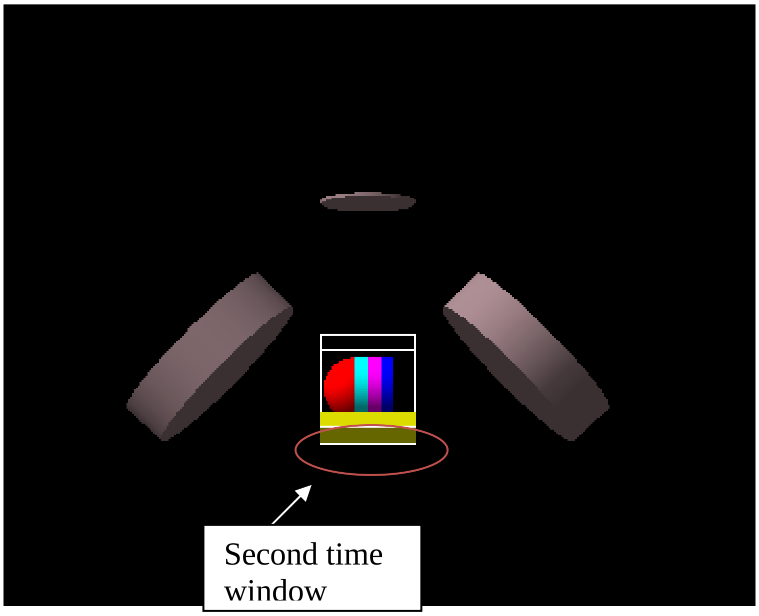


Figure 5.11: Second time window simulation

By analyzing the time of arrival of gamma rays from the neutrons were generated to the gamma rays hit the detectors, we have the histogram as below:

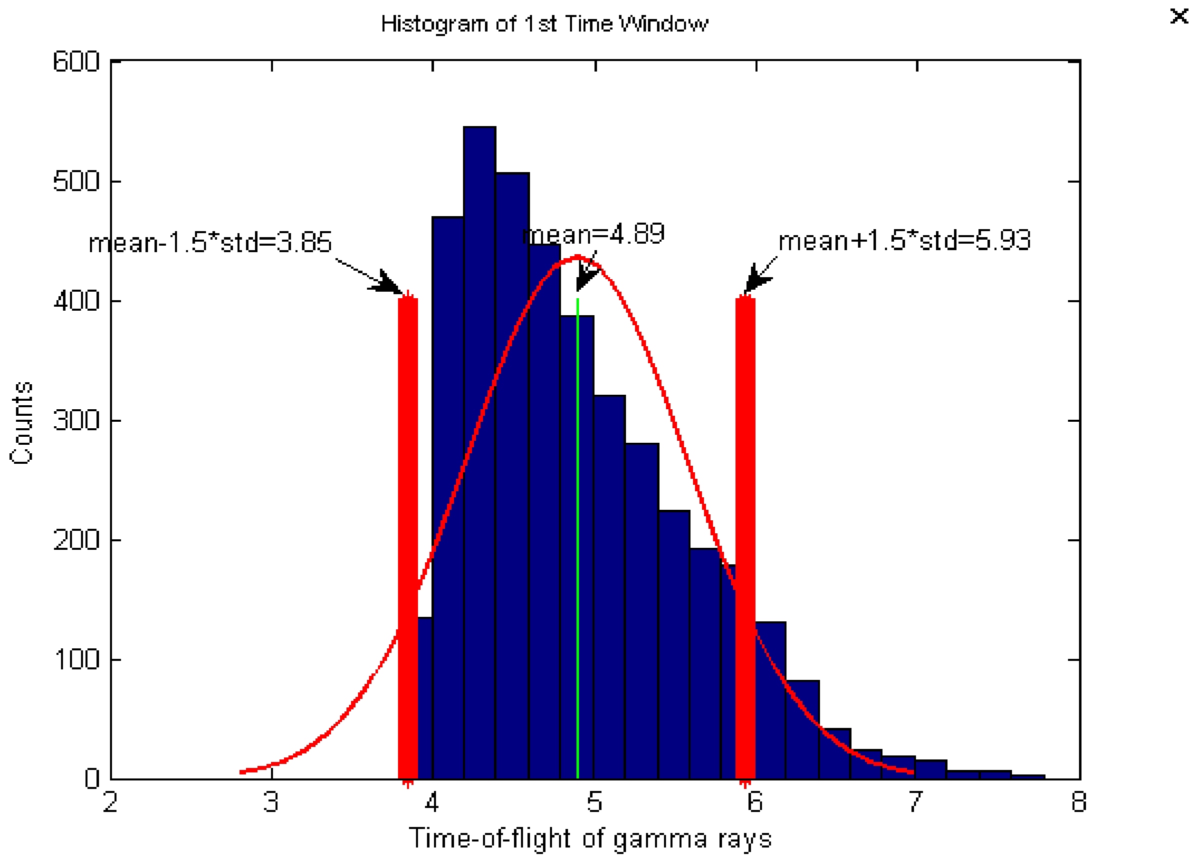


Figure 5.12: Histogram of the first time window

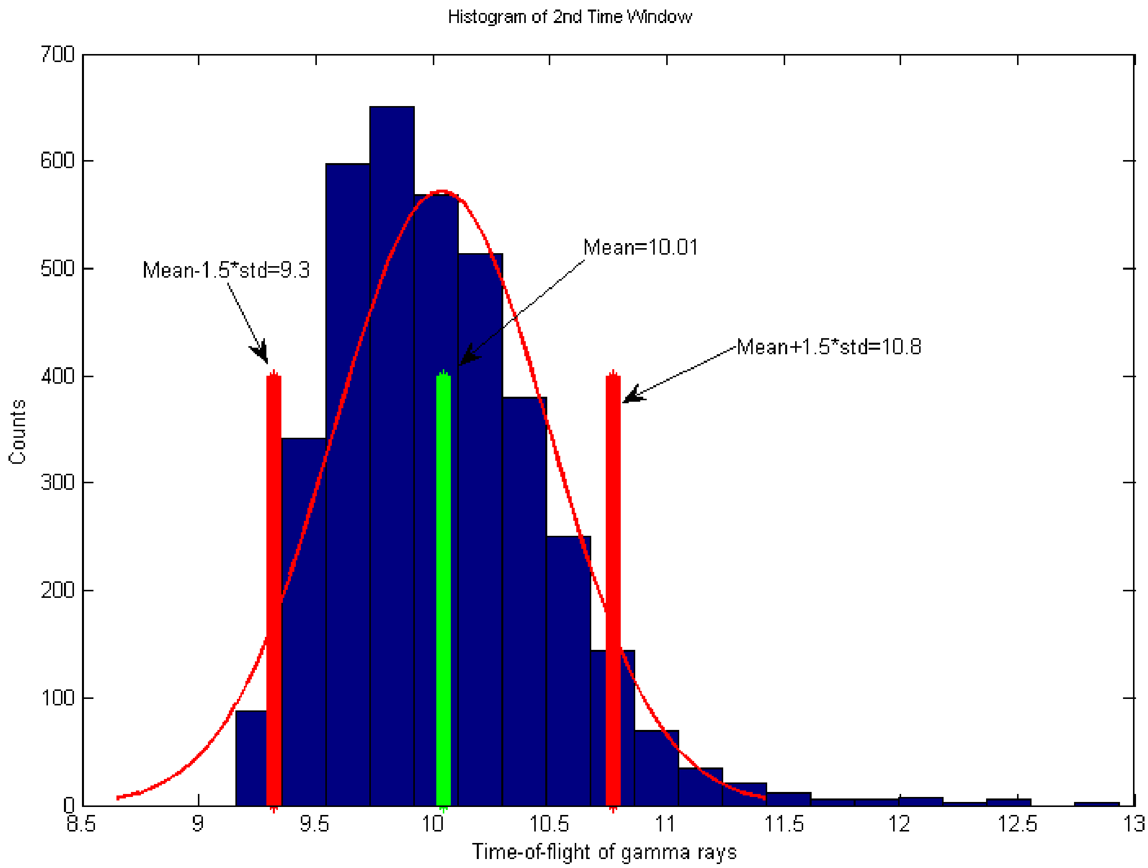


Figure 5.13: Histogram of the second time window

The upper boundary of the first histogram and the lower boundary of the second histogram complete the overall time window-6.0ns to 9.3ns. To testify the accuracy of the time window, we have tested it again with the third simulation, hexagen in various depths of soil. The black line shows the spectrum without time window whereas the colorful ones are the results with time window. The comparisons are as follow, see Fig. 5.14-5.17:

(i) Hexagen in 5cm soil

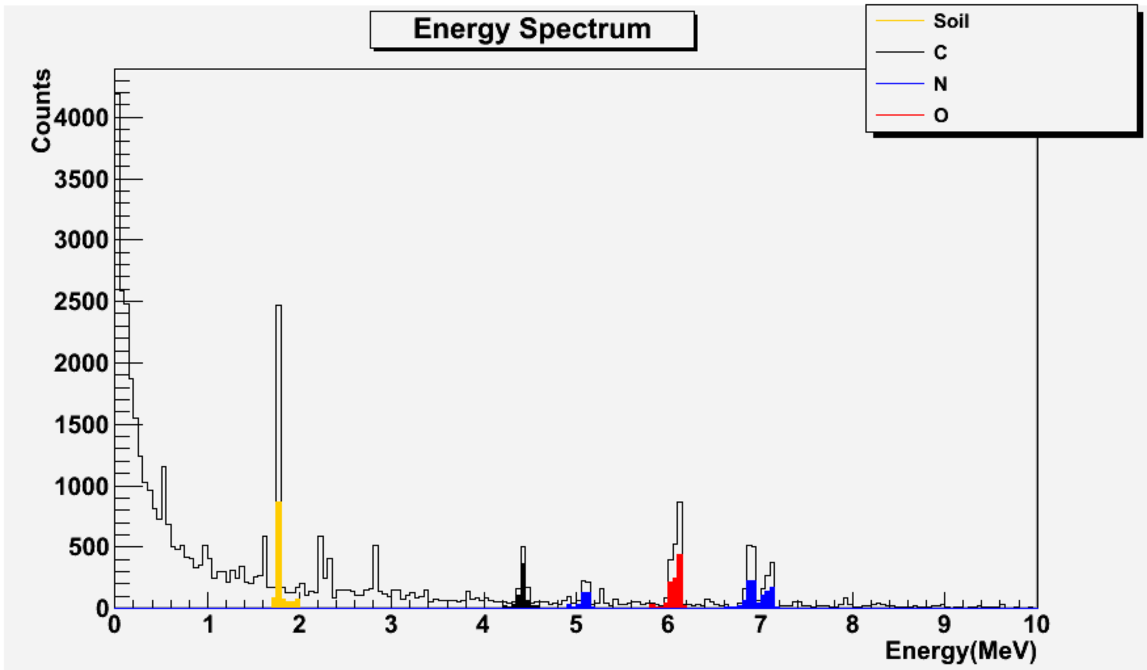


Figure 5.14: Comparison in spectrum between with-time-window and without-time-window for Hexagen 5cm in soil

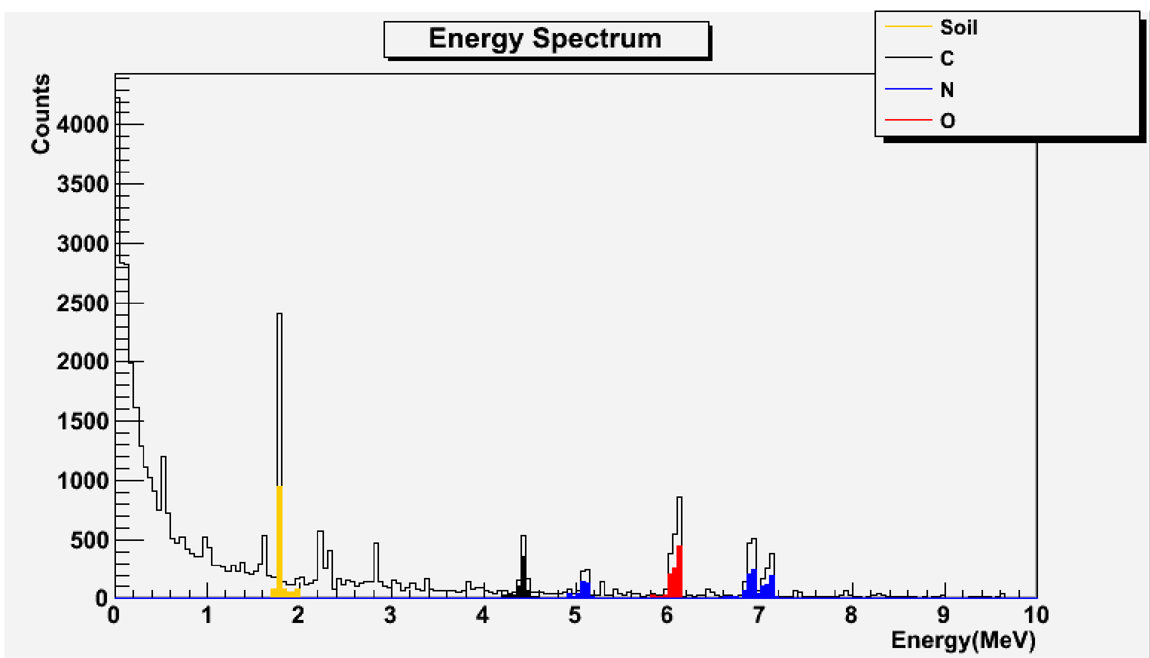


Figure 5.15: Comparison in spectrum between with-time-window and without-time-window for Hexagen 10cm in soil

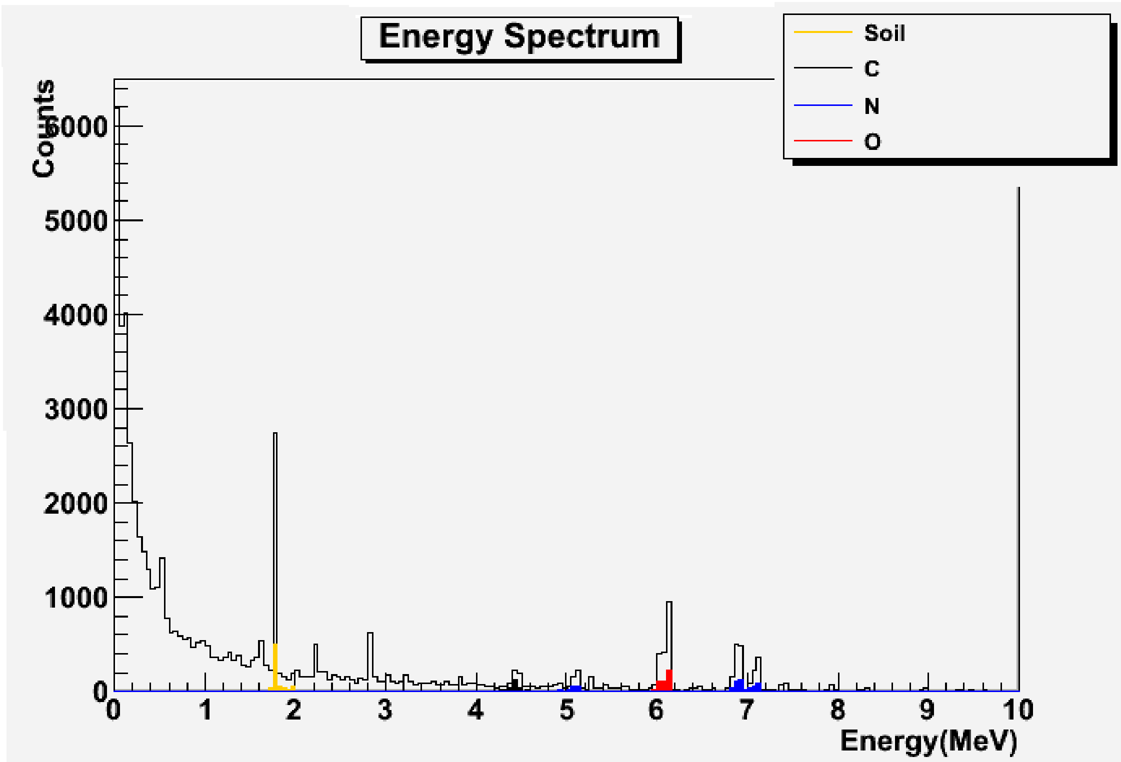


Figure 5.16: Comparison in spectrum between with-time-window and without-time-window for Hexagen 15cm in soil.

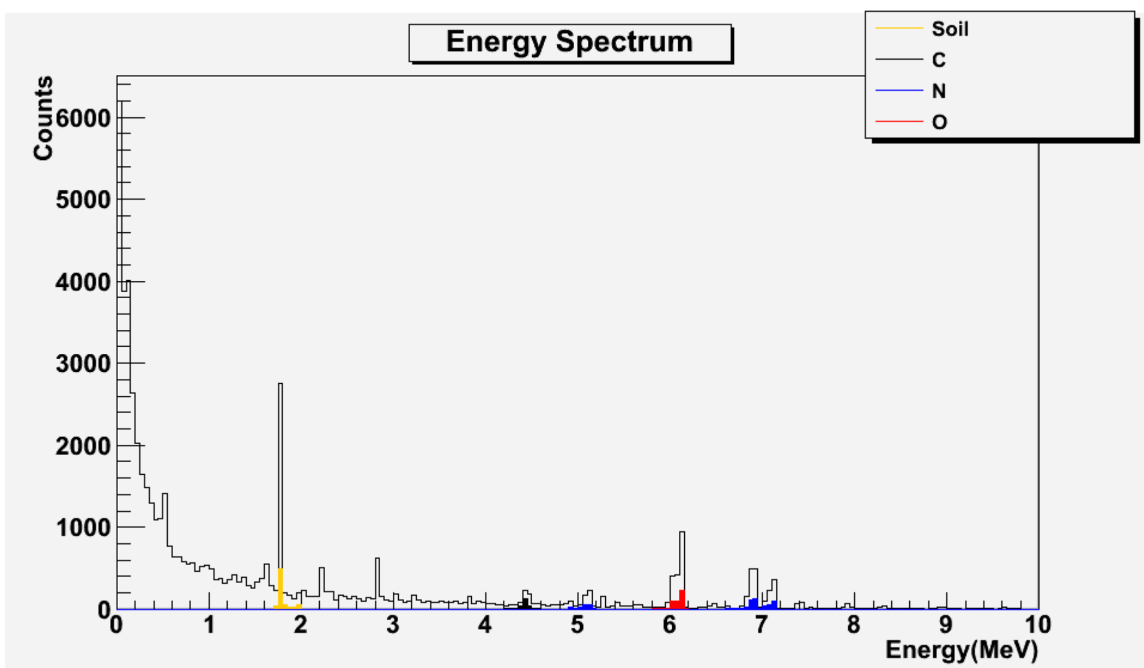


Figure 5.17: Comparison in spectrum between with-time-window and without-time-window for Hexagen 20cm in soil

From the comparisons above, the soil signal was largely reduced (>70%), while although other signals were comparatively weaken at the same time, they stayed strong elemental-specific.

Chapter 6

6. Conclusion and Future Work

The D+T fusion reaction has been successfully simulated. Neutron source is fast and accurately produced. More importantly, the UXO sensing system model has been built in GEANT4. It is easily manipulated and time-efficient. Satisfactory imaging reconstruction of the UXO and elemental-specific spectrum is generated once the simulation run ends.

The future work lies in advanced elemental analysis implementation, changes to the initial energy of Deuterium as well as real UXO sensing system building.

(i) In the case of elemental composition analysis, we need another separate program to analyze the energy information, so that it can tell us directly what material the UXO is made of[20]. Presently, we are capable of knowing each single element in a composite material but the analysis of unknown material whether high explosive or benign remains. Also, one of the elements-Iron did not give us correct information as it should have. Libraries and cross section files need to be verified for Iron.

(ii) Real experiments will be set up in the near future to benchmark the experimental results with GEANT4 simulation results.

I see my work as a fundamental first-step success to the development of the UXO sensing system. It is hoped that this model aids researchers, students, professors as well as other professionals working in GEANT4 simulation, neutron source modeling, detector developing, associated particle imaging technique and other related field.

Bibliography

[1] Moyes, R., 2005. Landmine Action. “Explosive remnants of war and mines other than anti-personnel mines”. Report of Landmine Action, Landmine Action, London, UK.

[2] United States. Office of the Under Secretary of Defense. “Unexploded Ordnance”. Report of the Defense Science Board Task Force. Washington, D.C, 2003.

[3] Abdu, H., and Robinson, D.A., & Jones, S.B., 2005. “Comparing Bulk Soil Electrical Conductivity Determination Using the DUALEM-1S and EM38-DD Electromagnetic Induction Instruments”. *Soil Science Society of America Journal*, 71(189-196).

[4] Rodger C. M., 2000. “Applications and Availability of Californium-252 Neutron Sources for Waste Characterization”. In Spectrum 2000 International Conference on Nuclear and Hazardous Waste Management.

[5] Vourvopoulos, G., Womble, P. C., & Paschal, J., 2000. “PELAN: A pulsed neutron portable probe for UXO and landmine identification”. Proceedings

of SPIE, the International Society for Optical Engineering, vol. 4142(142 149).

[6] Meo, S. L., Bennati, P., et al. 2009. “A Geant4 simulation code for simulating optical photons in SPECT scintillation detectors. *Journal of Instrumentation*, **4**, 07002.

[7] Chichester., D.L., et al, 2005. “The API 120: A portable neutron generator for the associated particle technique”. Nuclear Instruments and Methods in Physics Research, B 241(753-758)

[8] Oracle. http://library.thinkquest.org/17940/texts/fusion_dt/fusion_dt.html

[9] Wikipedia. http://en.wikipedia.org/wiki/Nuclear_fusion.

[10] Benveniste, J., & Zenger, J., 1954. “Information on the Neutrons Produced in the H^3(d,n)He^4 Reaction”. Report of Atomic Energy Commission, United States.

[11] Benveniste, J., Mitchell, A. C., Schorader, C. D., & Zenger, J. H., 1960. “The Problem of Measuring the Absolute Yield of 14-Mev Neutrons by Means of an Alpha”. *Nuclear Instruments & Methods*, 7(3), 306-314.

[12] Geant4 Home Page, 2000. http://geant4.web.cern.ch/geant4/S.

[13] Agostinelli, S., Allison, J., & Amako, K., et al., 2003. "Geant4-a simulation toolkit," *Nuclear Instruments & Methods.* Res. A, vol. A506, pp. 250–303.

[14] GEANT4: Physics Reference Manual, Dec. 2006 [Online]. Available: http://geant4.web.cern.ch/geant4/support/userdocuments.shtml

[15] Carrier, J.F., Archambault, L., & Beaulieu, L., 2004. "Validation of GEANT4, an object-oriented Monte Carlo toolkit, for simulations in medical physics," *Med. Phys*., vol. **31**, **NO**.3, pp. 484–492

[16] Schubert, H., Kuznetsov, A., 2006. "Associated Particle Imaging: An Enabling Technology of Detection of Improvised Explosive". Detection and Disposal of Improvised Explosives, 123-125. Springer, Printed in the Netherlands.

[17] Shepp, L.A., Vardi,Y., 1982. "Maximum Likelihood Reconstruction for Emission Tomography", *IEEE Transactions on medical imaging*, vol. **MI-1, NO**. 2, pp. 113-122,

[18] Dickerman, E. R., 1996. "Associated-Particle Sealed-Tube Neutron Probe for Non-intrusive Inspection", Proceedings of the 14th International

Conference on the App. of Accel. In Res. and Indust., Denton, TX.

[19] Associated Particle Imaging (API), 1998. DOE/NV11718-223, Bechtel Nevada, Special Technologies laboratory, Santa Barbara, CA.

[20] Bruyant, P. P., 2002. "Analytic and Iterative Reconstruction Algorithms in SPECT". *Nuclear Instruments & Methods*, **43**, pp. 1343-1358.