

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования Национальный исследовательский ядерный университет
«МИФИ»

УДК 539.17

ОТЧЁТ
О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

**РАСЧЕТ ЭНЕРГЕТИЧЕСКИХ СПЕКТРОВ
НЕЙТРИНО НА УСКОРИТЕЛЕ В РЕЗУЛЬТАТЕ
РАСПАДОВ ПИ-МЕЗОНА(π^-) И К-МЕЗОНА(K^-) С
ИСПОЛЬЗОВАНИЕМ ПРОГРАММЫ GEANT4**

Научный руководитель
д.ф-м.н., проф.

_____ А. А. Соколов

Студент

_____ Т. В. Махамов

Москва 2022

Содержание

Введение	3
1 Описание пакета Geant4	3
1.1 Описание архитектуры	3
1.2 Основные термины Geant4	4
1.3 Описание алгоритмов	4
2 Построение модели	5
2.1 Описание экспериментальной установки	5
2.2 Создание материнского пространства	5
2.3 Добавление источника частиц	6
2.4 Создание тормозного слоя и детектора	7
2.5 Получение и обработка данных	8
3 Результаты моделирования	10
4 Заключение	14
5 Список литературы	15

Введение

В физике высоких энергий предполагается владение информацией об элементарных процессах происходящих при взаимодействии частицы с веществом мишени. В настоящее время практически всегда, эксперимент в физике частиц, представляет собой многократное измерение совокупности случайных процессов — взаимодействия частиц и прохождения их через детектор. Аналитическое решение задачи о прохождении частиц через вещество описывается интегродифференциальными уравнениями, при этом, учет дополнительных процессов приводит к появлению дополнительных интегралов рассеяния, что существенно усложняет решение подобных уравнений. Альтернативой является метод Монте-Карло, который даёт удобный способ расчёта результатов таких измерений. Как раз для таких задач были разработаны инструменты для компьютерного моделирования, чтобы правильно подобрать параметры запуска. Для высокой точности калибровки оборудования, производятся тысячи и даже миллионы тестовых прогонов до реального запуска установки, и для этого требуется мощная вычислительная техника, например, достаточно распространённые сейчас кластеры и grid-системы с установленным на ней программным обеспечением для моделирования экспериментов. И Geant4 является одним из таких инструментов. Комплекс разработан специалистами исследовательской организации ЦЕРН.

Целью данной работы является проведение моделирования переноса нейтринного пучка через вещество средствами программного комплекса Geant4.

1 Описание пакета Geant4

GEANT4("Geometry and Tracking 4") – программа для симуляции методом Монте-Карло, взаимодействий частиц, проходящих через вещество. Программа написана на языке C++. Она используется во многих областях физики высоких энергий, в таких областях, как медицина и астрофизика, где необходим учет ядерных взаимодействий. Это программа с открытым исходным кодом, поэтому каждый пользователь может модифицировать ее для своих целей. Основное направление Geant4 – моделирование задач физики высоких энергий, или физики элементарных частиц, т.е. рассмотрение природы на очень малых расстояниях с целью исследования элементарных составляющих веществ и их взаимодействий. Чем глубже проникновение в материю, тем больше требуется энергии для сталкивающихся частиц, так как по принципу неопределённости существует связь между энергией и расстоянием, при которых происходят процессы в микромире. Принцип неопределённости – это фундаментальный закон в квантовой теории, который гласит, что так называемые дополнительные физические величины вроде энергии и времени, характеризующие физическую систему, не могут одновременно принимать точные значения ($\Delta p \Delta x \sim \hbar$, где \hbar – постоянная Планка).

1.1 Описание архитектуры

Geant4 писался под софтверные нужды современных экспериментов. Как и любая обычная система программного обеспечения, он содержит следующие компоненты: генератор событий, симуляция детектора, реконструкция и анализ. И все эти компоненты могут быть использованы как отдельно, так и в различных комбинациях. Ключевые элементы симуляции прохождения частиц через материю являются:

- геометрия и материалы;
- взаимодействие частиц в материи

- управление трекингом;
- оцифровка и обработка срабатываний;
- управление событиями и треком;
- визуализация и фреймворк для визуализации;

1.2 Основные термины Geant4

Run (сеанс) – самый крупный элемент моделирования, представляющий собой последовательность событий. Во время него условия эксперимента (набор физических процессов и геометрическая модель) остаются неизменными. Представлен классом G4Run, управляется экземпляром класса G4RunManager.

Event (событие) – единичное независимое измерение физического явления детектором. Описывающий событие класс – G4Event – содержит все входные и выходные данные смоделированного события. Экземпляр G4RunManager создаёт объект класса G4Event, который затем передаётся объекту класса G4EventManager для осуществления управления событием. Структура события следующая:

- первичное состояние (вершина и частица);
- траектории;
- коллекция срабатываний;
- коллекция оцифрованных сигналов;

Step (шаг) – минимальное продвижение частицы сквозь вещество в условиях протекания различных физических процессов. Описывается классом G4Step.

Track (трек) – траектория движения частицы. В экземпляре класса G4Track хранится информация о полном продвижении частицы в веществе на момент обращения, а также описание частицы, текущего физического процесса и геометрического объёма.

Hit (срабатывание) – описание одного взаимодействия частицы с веществом в области детектирования – координаты, энергия, импульс в момент взаимодействия, а также время взаимодействия. Моделирование оцифрованного сигнала происходит на основе именно этой информации.

1.3 Описание алгоритмов

В Geant4 реализовано огромное количество алгоритмов, каждый из которых предназначен для своей области, а их в комплексе довольно много. Требуются разные методы моделирования для разных частиц и соответствующих энергий. Но в основе всех методов лежит один общий - метод Монте-Карло. Все методы в Geant4 представляют собой различные композиции и отклонения методов МонтеКарло. Кратко решение задачи данным методом выглядит так. Предположим, нам нужно получить x из интервала $[x_1, x_2]$ на распределении $f(x)$, и нормализованная плотность вероятности может быть записана следующим образом:

$$f(x) = \sum_{i=1}^n N_i f_i(x) g_i(x)$$

где, $N_i > 0$, $f_i(x)$ - нормализованные функции плотности вероятности на $[x_1, x_2]$, и $0 < g_i(x) \leq 1$. Согласно этому методу, x может быть получено следующим путём:

1. Выбирается случайное целое число $i \in \{1, 2, \dots, n\}$ с вероятностью, пропорциональной N_i ;
2. Выбирается значение x_0 на распределении $f_i(x)$
3. Вычисляется $g_i(x_0)$ и принимается $x = x_0$ с вероятностью $g_i(x_0)$
4. Если x_0 отклоняется, вернуться к шагу 1

2 Построение модели

2.1 Описание экспериментальной установки

Для проведения тестового моделирования была написана программа, описывающая следующую установку:

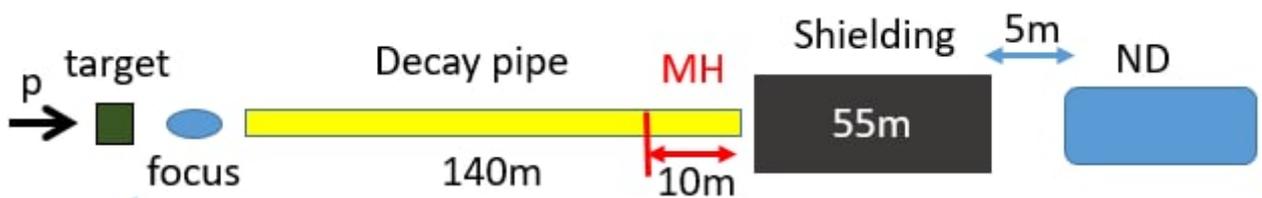


Рисунок 1 — Схема модулируемой установки

Установка состоит из 4 основных частей, параметры которых были изменены для удобства измерений:

- ParticleGun - генератор частиц, находящийся в начале координат
- Decay Pipe - вакуумная трубка в которой летит сгенерированная частица, длиной 100 м, радиусом 1 м
- Shielding - железный слой, длиной 50 м
- ND - детектор, для регистрации нейтрино, радиусом 1 м, и длиной 1 м, находящийся на расстоянии 150 м от центра

Описание экспериментальной установки происходит в файле DetectorConstruction.cc. Сначала определяется конструктор класса World, он принимает значения ширины, высоты, глубины и материала, данный класс создаёт материнский объём в котором будут располагаться элементы установки.

2.2 Создание материнского пространства

```
G4Box* solidWorld = new G4Box("World", 0.5*world_sizeXY,
0.5*world_sizeXY, 0.5*world_sizeZ);

G4Material* world_mat = nist->FindOrBuildMaterial("G4_Galactic");
G4LogicalVolume* logicWorld = new G4LogicalVolume(solidWorld,
world_mat, "World");
G4VPhysicalVolume* physWorld = new G4PVPlacement(0, G4ThreeVector(),
logicWorld, "World", 0, false, 0, true);
```

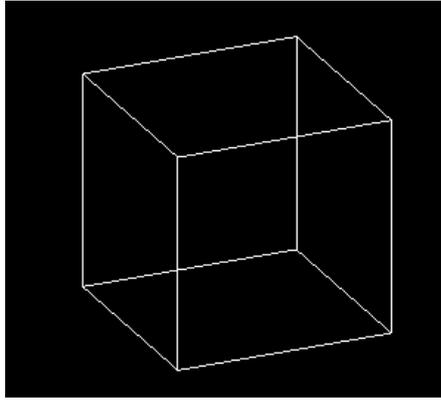


Рисунок 2 — Материнское пространство

Здесь `world_sizeXY` и `world_sizeZ` соответствуют размерам куба. Сам куб заполнен так называемым газом «G4_Galactic», который является приемлемой альтернативой вакууму. Чтобы определить материал (`world_mat`), из которого сделано материнское пространство, использовался встроенный класс «G4LogicalVolume». А с помощью класса «G4VPhysicalVolume» создается куб внутри материнского объема в определенной точке и с заданным вращением. На рисунке 2 показан общий вид материнского объема. Для дальнейшего удобства мы не будем отображать границы куба.

2.3 Добавление источника частиц

Самым оптимальным способом создания пучка частиц в GEANT4 является использования встроенного класса «G4ParticleGun». В этом классе задается положения, тип, энергия, направление вылета и распределение вылетающихся частиц:

```
G4int n_particle = 1;
fParticleGun = new G4ParticleGun(n_particle);
```

где `n_particle` определяет количество частиц, вылетающих из источника за один прогон. Все необходимые частицы берутся из уже созданной таблицы «G4ParticleTable» в GEANT4:

```
G4ParticleTable* particleTable = G4ParticleTable::GetParticleTable();
G4String particleName;
```

```
G4ParticleDefinition* particle
= particleTable->FindParticle(particleName="pi-");
```

```
fParticleGun->SetParticleDefinition(particle);
```

Где класс `G4ParticleDefinition` позволяет задать такие свойства частицы как масса, спин, моды распада и т.д. Установим начальную энергию частицы равной 10 GeV, также добавим генератор случайных чисел от 0 до 5, для генерирования энергий в диапазоне от 10 до 15 GeV, и направление ее импульса вдоль оси Z:

```
fParticleGun->SetParticleMomentumDirection(G4ThreeVector(0.,0.,1.));
fParticleGun->SetParticleEnergy(10 + 5.*G4UniformRand())*GeV;
```

2.4 Создание тормозного слоя и детектора

Теперь, когда наш логический объем и источник частиц созданы, мы можем приступить к созданию цилиндрического слоя, железного слоя и детектора. «G4Tubs» — это класс, который используется для создания цилиндрических форм в GEANT4, и его характеристики (материал, координаты и т.д.) задаются аналогичным образом характеристикам материнского объема. Рассмотрим создание сплошного железного слоя:

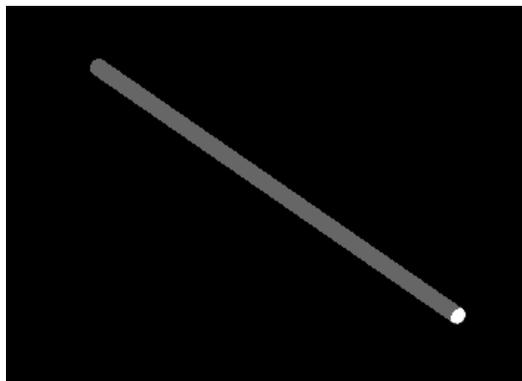


Рисунок 3 — Железный слой

Сначала определим материал детектора. В классе «G4NistManager» имеется метод «FindOrBuildMaterial», который находит predetermined материал в GEANT4 или создает его с нуля в случае его отсутствия. Здесь пользователю предоставлен широкий простор для реализации нужных ему материалов, так как в GEANT4 предусмотрена возможность создавать материалы, начиная с создания элементов. Далее из созданных элементов создается сам материал. Материал можно создавать несколькими способами, например, из элементов создать молекулы или смесь нескольких элементов с определенным весовым коэффициентом. Но в нашем случае полистирол уже указан в списке материалов, как «G4_FE»:

```
G4NistManager* nist = G4NistManager::Instance();
G4Material *tubeMatFe = nist->FindOrBuildMaterial("G4_Fe");
```

```
G4Tubs* solidTargetLayer1 = new G4Tubs( "solidTargetLayer1",
                                          0.*m,
                                          1.*m,
                                          25.*m,
                                          0 *degree,
                                          360* degree );
```

Точно так же мы устанавливаем его логический объем в границах цилиндра и помещаем его во внутрь куба, с помощью класса «G4PVPlacement» (рисунок 3):

```
fLogicTargetLayer1 = new G4LogicalVolume( solidTargetLayer1,
                                           tubeMatFe,
                                           "fLogicTargetLayer1",
                                           0,
                                           0,
                                           0 );
fPhysTargetLayer1 = new G4PVPlacement( 0,
                                       G4ThreeVector(0,0,75*m),
                                       "physTargetLayer1",
```

```
fLogicTargetLayer1 ,
fPhysExperimentalHall ,
false ,
0 ,
checkOverlaps );
```

Аналогичным образом создается вакуумный цилиндр и детектор, с отличием в параметрах, то есть, материал и длина трубки.

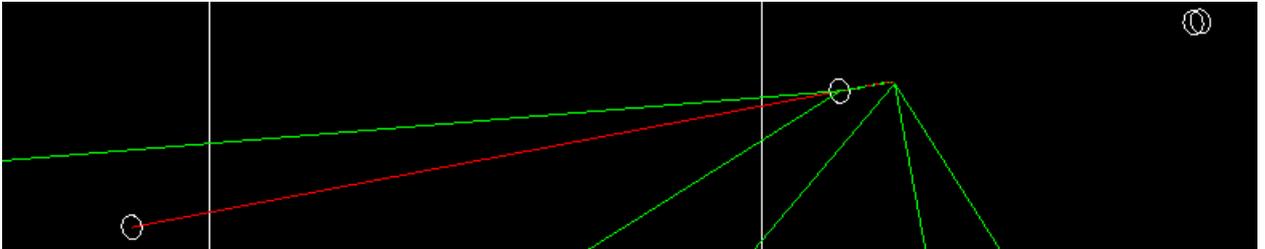


Рисунок 4 — Пи-мезон в отсутствии распада в камере распада

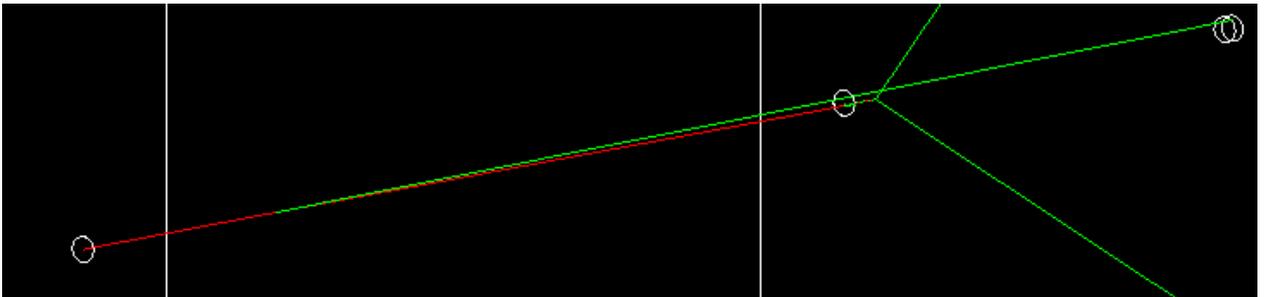


Рисунок 5 — Пи-мезон при распаде в камере распада

2.5 Получение и обработка данных

Для наглядности полученных данных из каждого шага(G4Step), был использован макрофайл из примера B1, под названием run.mac.

```
/run/initialize
#
/control/verbose 2
/run/verbose 2
/event/verbose 0
/tracking/verbose 1
#
/gun/particle pion-
/gun/energy 10 GeV
#
/run/beamOn 100000
```

В результате работы макрофайла получаем, вывод для каждого прогона:

```

*****
* G4Track Information: Particle = pi-, Track ID = 1, Parent ID = 0
*****
Step#   X(mm)   Y(mm)   Z(mm) KinE(MeV)  dE(MeV) StepLeng TrackLeng  NextVolume ProcName
  0      0      0      -5e+04  9.86e+03      0      0      0      expHall initStep
  1      0      0      2.86e+04  9.86e+03  2.64e-21  7.86e+04  7.86e+04  expHall Decay
*****
* G4Track Information: Particle = mu-, Track ID = 3, Parent ID = 1
*****
Step#   X(mm)   Y(mm)   Z(mm) KinE(MeV)  dE(MeV) StepLeng TrackLeng  NextVolume ProcName
  0      0      0      2.86e+04  7.11e+03      0      0      0      expHall initStep
  1     -31.4   78.2    5e+04  7.11e+03  7.19e-22  2.14e+04  2.14e+04  physTargetLayer1 Transportation
  2     -105   261    1e+05  7.11e+03  1.68e-21  5e+04  7.14e+04  physTargetLayer2 Transportation
  3     -106   264   1.01e+05  7.11e+03  3.35e-23  1e+03  7.24e+04  OutOfWorld Transportation
*****
* G4Track Information: Particle = anti_nu_mu, Track ID = 2, Parent ID = 1
*****
Step#   X(mm)   Y(mm)   Z(mm) KinE(MeV)  dE(MeV) StepLeng TrackLeng  NextVolume ProcName
  0      0      0      2.86e+04  2.79e+03      0      0      0      expHall initStep
  1     81.4   -203    5e+04  2.79e+03      0  2.14e+04  2.14e+04  physTargetLayer1 Transportation
  2     271   -675    1e+05  2.79e+03      0  5e+04  7.14e+04  physTargetLayer2 Transportation
  3     275   -685   1.01e+05  2.79e+03      0  1e+03  7.24e+04  OutOfWorld Transportation

```

Рисунок 6 — Выходные значения макрофайла для пи-мезон при распаде

Одним из наиболее удобных способов анализа результатов и построения интерактивных гистограмм является использования объектно-ориентированного программного пакета ROOT. Для хранения данных о потере энергии мы используем набор новых встроенных классов «G4AnalysisManager», «G4Run» и «G4UserRunAction». Теперь давайте создадим новый локальный класс под названием «MyRunAction», который будет наследовать их всех:

```

class MyRunAction : public G4UserRunAction {
public:
    MyRunAction ();
    ~MyRunAction ();
    virtual void BeginOfRunAction ();
    virtual void EndOfRunAction (); };

```

Мы уже объявили новый локальный класс и два метода внутри него (BeginOfRunAction, EndOfRunAction). Чтобы сохранить все необходимые данные, нам нужно создать кортеж, используя только что объявленный класс «MyRunAction»:

```

MyRunAction::MyRunAction ()
{
    G4AnalysisManager *man = G4AnalysisManager::Instance ();
    man->CreateNtuple ("Decay Coordinate ");
    man->CreateNtupleDColumn ("zPos ");
    man->FinishNtuple (0);
}
MyRunAction::~~MyRunAction ()
{}

```

Теперь новый кортеж может быть записан внутри нового файла с расширением **.root**, используя метод «BeginOfRunAction» для открытия этого нового файла и «EndOfRunAction» для его закрытия:

```

void MyRunAction::BeginOfRunAction ()

```

```

{
  G4AnalysisManager *man = G4AnalysisManager::Instance();
  man->OpenFile("output.root");
}

void MyRunAction::EndOfRunAction()
{
  G4AnalysisManager *man = G4AnalysisManager::Instance();

  man->Write();
  man->CloseFile();
}

```

Несмотря на то, что кортеж у нас создан, он все еще пуст и не содержит никаких данных, на основе которых мы могли бы построить гистограмму, следовательно нам нужно создать параметры для заполнения и добавить их в гистограмму(на примере координаты):

```

const G4double mom =
  theStep->GetPreStepPoint()->GetMomentum().mag();
const G4double zPos =
  theStep->GetPostStepPoint()->GetPosition().z();
const G4double mu_coor =
  theStep->GetPreStepPoint()->GetPosition().z();

const G4double angle =
  57.29*std::acos(theStep->GetPostStepPoint()
    ->GetMomentumDirection().z());

G4AnalysisManager *man = G4AnalysisManager::Instance();
man->FillNtupleDColumn(0, zPos);
man->AddNtupleRow(0);

```

3 Результаты моделирования

Исходя из табличных значений, масса и время жизни для пи-мезона(π^-), составляет:

$$\text{Mass } m = 139.57018 \pm 0.00035 \text{ MeV};$$

$$\text{Lifetime } \tau = 2.6033 \pm 0.0005 \times 10^{-8} \text{ s};$$

$$\text{Схема распада для } \pi^-: \pi^- \rightarrow \mu^- + \bar{\nu}_\mu.$$

Для К-мезона(K^-):

$$\text{Mass } m = 493.677 \pm 0.005 \text{ MeV};$$

$$\text{Lifetime } \tau = 1.2380 \pm 0.0020 \times 10^{-8} \text{ s};$$

$$\text{Из всех возможных распадов } K^-, 63\% \text{ составляет: } K^- \rightarrow \mu^- + \bar{\nu}_\mu.$$

В данной работе моделируется прохождение пи-мезона(π^-) и К-мезона(K^-) с энергиями 1 GeV через вакуумную трубку. Движение частиц разбивается программой на шаги(шаги Steps). Каждый шаг это минимальное продвижение частицы с учётом процессов, происходящих с ней. Таких как потеря энергии, столкновение с веществом, отделение новой частицы. В данном случае, исследуется распад на мюон и мюонное анти-нейтрино, которые и представлены на графиках7(a), 7(b). График распределения точек

распада частиц с энергией 1 GeV на оси Z был построен для наглядности представления экспоненциальной зависимости убывания числа распадов от длины.

В случае запуска программы 100000 раз, можно наблюдать на графике для пи-мезона(π^-) распад в 79% случаев, а для К-мезона(K^-) в 100% случаев из которых 63% приходится на распад $K^- \rightarrow \mu^- + \bar{\nu}_\mu$, что соответствует теоретическим значениям. Более того, на графике К-мезона(K^-) наблюдается пик в окрестности точки Z = 10 метров, что можно было оценить аналитически используя среднее время жизни:

$$Z_{\text{рас}} = c \cdot \sqrt{\gamma^2 - 1} \cdot \tau \approx 7\text{м.}$$

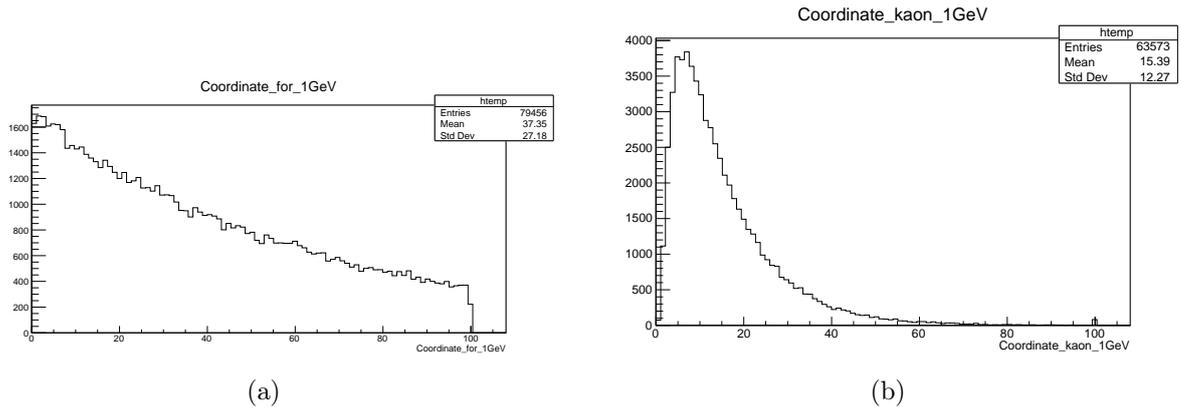


Рисунок 7 — Распределения точек распада частиц с энергией 1 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

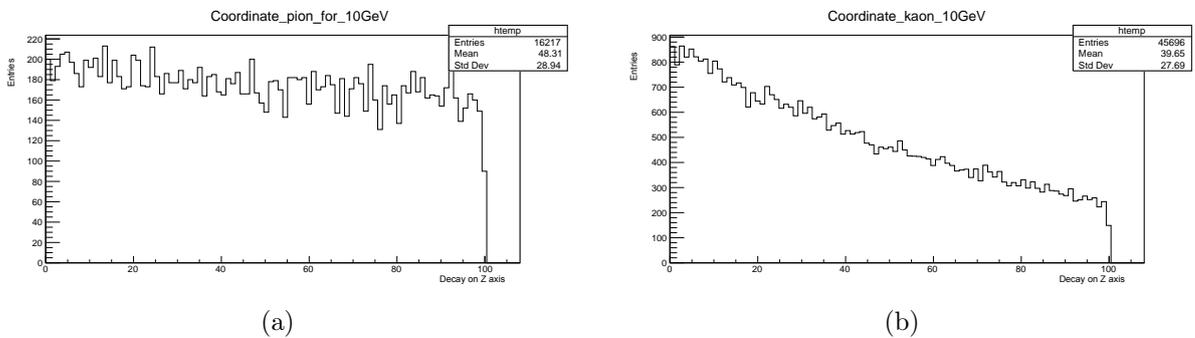
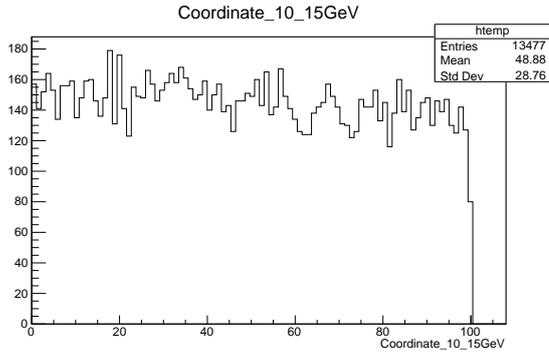
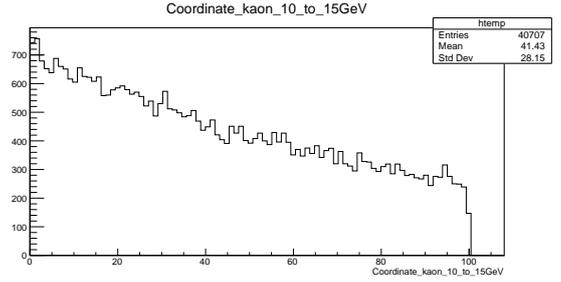


Рисунок 8 — Распределения точек распада частиц с энергией 10 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

На графике8 можно наблюдать уменьшение количества распадов с 79% до 16%, и от 63% до 45% для Пи-мезона(π^-) и К-мезона(K^-), соответственно. Аналогично, были построены графики9(a),9(b) для случая энергий частиц в пределах от 10 до 15 GeV.

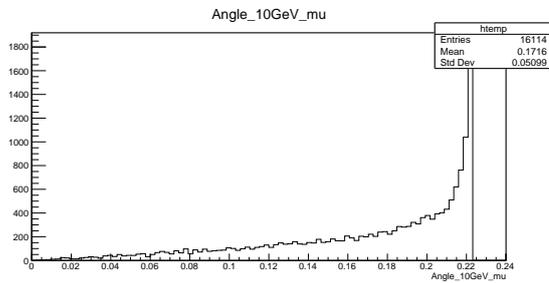


(a)

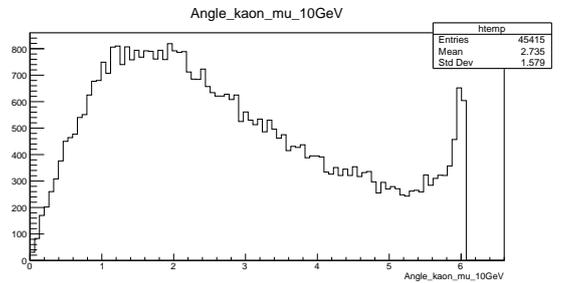


(b)

Рисунок 9 — Распределения точек распада частиц с энергией от 10 до 15 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

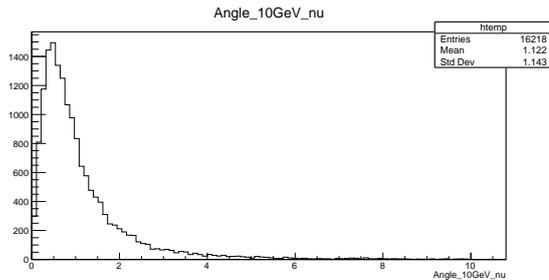


(a)

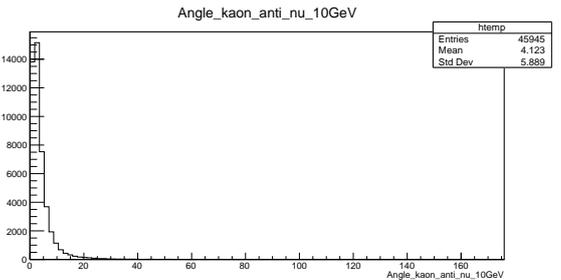


(b)

Рисунок 10 — Распределения углов для мюона после распада частиц с энергией 10 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)



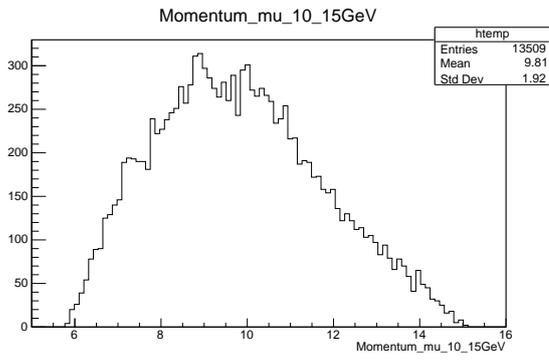
(a)



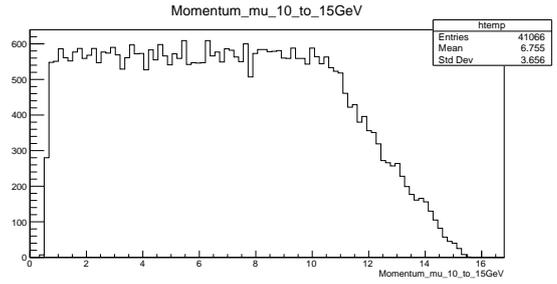
(b)

Рисунок 11 — Распределения углов для мюонного антинейтрино после распада частиц с энергией 10 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

На рисунках 10 11 показаны гистограммы угла мюона и антинейтрино, соответственно, в результате распада пи-мезона и к-мезона с энергиями 10 GeV. Максимальное предельное значение для угла вылета мюона в результате распада Пи-мезона(π^-), составляет 0.22° , а для К-мезона(K^-) в следствии большого различия масс между родительской и вторичной частиц, образуется большое отклонение достигающее в пределе, значения в 6° . Аналогичная ситуация для гистограмм 10(b) 11(b) мюонного антинейтрино.

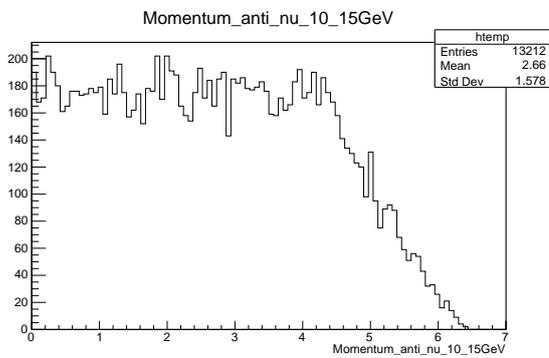


(a)

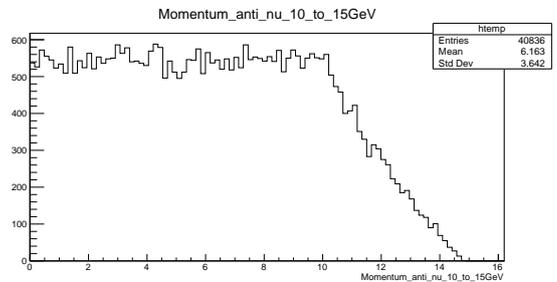


(b)

Рисунок 12 — Распределения импульсов для мюона после распада частиц с энергиями от 10 до 15 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

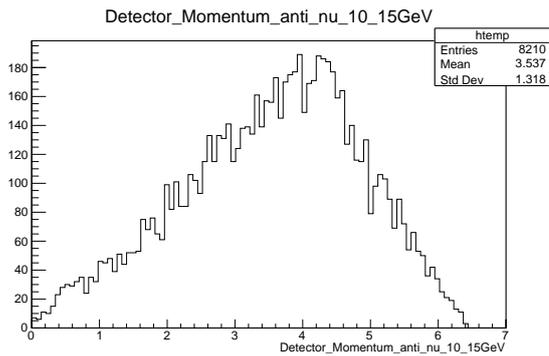


(a)

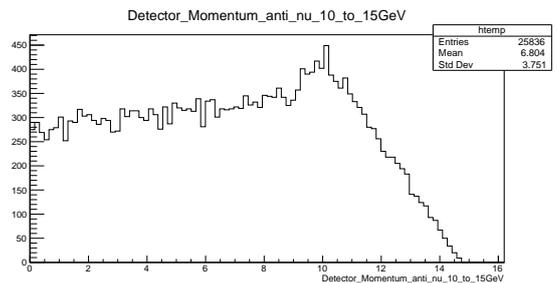


(b)

Рисунок 13 — Распределения импульсов для мюонного антинейтрино после распада частиц с энергиями от 10 до 15 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)



(a)



(b)

Рисунок 14 — Импульсные спектры для мюонного антинейтрино попавшего в область детектора после распада частиц с энергиями от 10 до 15 GeV на оси Z для Пи-мезона(π^-)(a), К-мезона(K^-)(b)

На рисунках 12(a), 13(a) представлены гистограммы для импульсных спектров мюона и мюонного антинейтрино в результате распада пи-мезона с исходной энергией от 10 до 15 GeV. Для гистограммы мюонного антинейтрино можно наблюдать, сплошной спектр импульсный в диапазоне от 0 до 4.5 GeV, с последующим спуском. Аналогичные

гистограммы(12(b),13(b)) были построены в результате распада К-мезона, в диапазоне энергий от 10 до 15 GeV. Имеет место, более жесткий спектр импульсов, как для мюона, так и для антинейтрино, объясняемый кинематикой процесса.

4 Заключение

В данной работе были изучены:

- Формирование нейтринных пучков на ускорителе
- Принципы работы с программным пакетом Geant4
- Принципы построения гистограмм в библиотеке Root
- Построенна модель, аналог ускорителя У-70
- Произведен импульсный расчет нейтринных пучков в ускорителе

В ходе тестового моделирования было рассмотрено формирование пучка нейтрино на ускорителе. На Рисунке14 был построен спектр импульсов для мюонного антинейтрино попавшего в чувствительную область детектора в результате распада Пи-мезона(π^-) и К-мезона(K^-) с энергиями в диапазоне от 10 до 15 GeV. Импульсный спектр14(b) для К-мезона(K^-) является более жёстким. Средние значения для импульсов мюонного антинейтрино в результате распада составляет 6.163 GeV, для распада К-мезона, и 2.66 GeV Пи-мезона. В прогоне 100000 частиц, доля распавшихся родительских частиц составляет 41% и 13%, а доля мюонного антинейтрино попавших в область детектора в результате распада, составляет 26% и 8%, для К и Пи мезонов, соответственно.

5 Список литературы

1. Официальный сайт GEANT4: <https://geant4.web.cern.ch/>
2. Официальный сайт ROOT: <https://root.cern/>
3. Документация GEANT4: <https://apc.u-paris.fr/~franco/g4doxy/html/>
4. Примеры GEANT4: <https://gitlab.cern.ch/geant4/geant4/-/tree/master/examples>
5. Mathieu Perrin-Terrin. Neutrino Tagging: a new tool for accelerator based neutrino. - Eur. Phys. J. C manuscript No 23.12.2021
6. P.A. Zyla et al. (Particle Data Group), Prog. Theor. Exp. Phys. 2020, 083C01 (2020) <https://pdg.lbl.gov/2020/listings/rpp2020-list-K-plus-minus.pdf>
7. K.A.Olive et al. (Particle Data Group), Chin. Phys. C, 38, 090001 (2014) and 2015 update <https://pdg.lbl.gov/2015/tables/rpp2015-tab-mesons-light.pdf>