

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет ФИЗИЧЕСКИЙ

Кафедра ФИЗИКО-ТЕХНИЧЕСКОЙ ИНФОРМАТИКИ

Магистерская программа ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ И СИСТЕМЫ

Направление подготовки 03.04.02 ФИЗИКА

Образовательная программа МАГИСТРАТУРА

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

научно-исследовательский формат

Белозерова Мария Сергеевна

(фамилия, имя, отчество автора)

Тема работы Разработка параметрического моделирования и интерфейсов первичных

генераторов событий для эксперимента на Супер С-Тау фабрике

«К защите допущена»

Декан ФФ НГУ

док. физ.-мат. наук

Блинов В. Е. /

(фамилия, И. О.)

(подпись)

« »

2022 г.

Научные руководители

канд. физ.-мат. наук

с.н.с. ИЯФ СО РАН

Воробьев В. С. /

(фамилия, И. О.)

(подпись)

« »

2022 г.

н.с. ИЯФ СО РАН

Разуваев Г. П. /

(фамилия, И. О.)

(подпись)

« »

2022 г.

Дата защиты: « » 2022 г.

Новосибирск, 2022

Содержание

1	Введение	4
2	Процесс анализа данных эксперимента	7
2.1	Эксперимент Супер s - τ фабрика	8
2.2	Фреймворк AuToGa	10
3	Первичные генераторы частиц	14
3.1	Генераторы в фреймворке AuToGa	15
3.2	Генератор ККМС	16
3.2.1	Устройство генератора	17
3.2.2	Тип выходных данных	18
3.2.3	Реализованный интерфейс	19
4	Пакет для параметрического моделирования	21
4.1	Виды моделирования детектора	21
4.2	Схема работы программного пакета SctParSim	24
4.3	Реализация инструментов	27
4.3.1	Дрейфовая камера	27
4.3.2	ФАРИЧ	30
4.3.3	Калориметр	31
4.3.4	Мюонная система	33
5	Обзор пакета для отбора событий	34
5.1	Инструмент для отбора событий	34
6	Результаты работы	36

6.1	Пример использования	36
6.2	Полученные результаты	39
7	Заключение	45
8	Список литературы	47
9	Приложение	52

1 Введение

Супер $c\text{-}\tau$ фабрика — проект симметричного электрон-позитронного коллайдера сверхвысокой светимости $10^{35} \text{ см}^{-2}\text{с}^{-1}$ с энергией в центре масс от 3 до 7 ГэВ, предложенный в институте ядерной физики им. Г. И. Будкера СО РАН (г. Новосибирск). Физическая программа проекта включает прецизионные эксперименты с τ -лептонами и очарованными адронами [1, 2], а также эксперименты по нахождению физики, не описываемой в рамках Стандартной Модели. К основным направлениям физической программы относятся:

- прецизионное измерение вероятностей переходов между уровнями чармония, их масс, полных и лептонных или двухфотонных ширин;
- систематическое измерение вероятностей распадов для всех низколежащих состояний чармония;
- изучение экзотических состояний чармония;
- исследование физики D -мезонов;
- прецизионное измерение характеристик τ лептона (массы, времени жизни, электрический и магнитный дипольные моменты);
- изучение лептонных распадов τ ;
- поиск редких явлений за рамками Стандартной Модели, таких как несохранение лептонного числа и CP -четности.

Место встречи пучков электронов и позитронов будет оборудовано универсальным детектором частиц, задача которого измерять параметры рожденных

в ходе столкновения частиц. Детектор с аксиальным магнитным полем состоит из нескольких подсистем: трековой системы, калориметра и систем идентификации.

Для физического анализа необходимы данные не только, полученные непосредственно с детектора, но и из моделирования. Моделирование позволяет получить данные близкие к данным реального эксперимента, основываясь на теоретических моделях. Оно используется для проведения предварительных физических анализов, определения чувствительности детектора к различным физическим событиям, разработки процедур калибровки систем детектора и методик реконструкции данных, отладки программного обеспечения.

Моделирование по степени детализации взаимодействия частиц и вещества подразделяется на полное, быстрое и параметрическое. Полное моделирование позволяет получить данные, максимально приближенные к реальному эксперименту. Отклик параметрического моделирования задается аналитическими функциями с набором параметров без привязки к конкретной реализации подсистемы. Быстрое моделирование, в отличие от параметрического, основано на знаниях о конкретной реализации детектора.

Цель выполненной квалификационной работы заключается в разработке программного обеспечения, необходимого для проведения физического анализа на Супер s - τ фабрике. В рамках чего были поставлены и решены задачи: реализация интерфейса первичного генератора событий рождения частиц в e^+e^- -столкновениях и разработка пакета параметрического моделирования детектора частиц. Под событием подразумевается группа частиц, рожденных в ходе одного столкновения.

Работа выполнялась с использованием фреймворка AuToG, который является стандартом для программного обеспечения проекта Супер s - τ фабрики.

В ходе данной работы был предложен алгоритм подключения генераторов

к программному окружению Супер c - τ фабрики и реализовано подключение генератора ККМС [3, 4]. Также было разработано параметрическое моделирование. Так как работа выполнена в рамках проекта Супер c - τ фабрики, и работа таких подсистем как ФАРИЧ (FARICH — Focusing Aerogel Ring Imaging Cherenkov) и мюонной пробежной системы известна, то моделирование было реализовано с элементами быстрого моделирования.

Выпускная квалификационная работа имеет следующую структуру изложения. Во втором разделе рассмотрены существующие программные пакеты, которые используются для моделирования детектора, а также фреймворк Aurora. В третьем разделе описана архитектура подключения генераторов в фреймворк Aurora и генератор ККМС. В разделе «Пакет для параметрического моделирования» описана реализация разрабатываемого пакета: его структура и параметризация, заложенная в подсистемы. В следующем разделе представлено описание пакета для отбора событий. Пример запуска программного обеспечения и полученные результаты продемонстрированы в шестом разделе. В седьмом разделе подведены итоги работы.

2 Процесс анализа данных эксперимента

Одним из основных методов изучения физики элементарных частиц являются эксперименты на ускорительных комплексах со встречными пучками. Место встречи пучков оборудовано детектором, который измеряет основные характеристики рождённых в ходе столкновения частиц, такие как импульс, энергия, координаты.

После получения откликов отдельных подсистем детектора происходит этап физической реконструкции данных. В ходе данного этапа данные группируются в треки, кластеры, в зависимости от подсистемы. После этого данные, полученные различными подсистемами детектора, объединяются. Это необходимо для определения типа частицы и её основных характеристик. Данные группируются по событиям, которые представляют собой набор частиц, рождённых в ходе одного столкновения. Далее информация о событиях поступает в систему хранения и используется для физического анализа.

Однако для конечного анализа недостаточно данных, полученных в детекторе, также необходимы данные математического моделирования эксперимента. Данные эксперимента и данные, полученные с помощью математического моделирования, обрабатываются одинаковым образом и затем сравниваются (Рис. 1). Подробнее виды и задачи моделирования детектора описаны в главе «Пакет для параметрического моделирования».

Далее описан эксперимент Супер s - τ фабрики и фреймворк Aurora, который является стандартом программного окружения данного проекта.

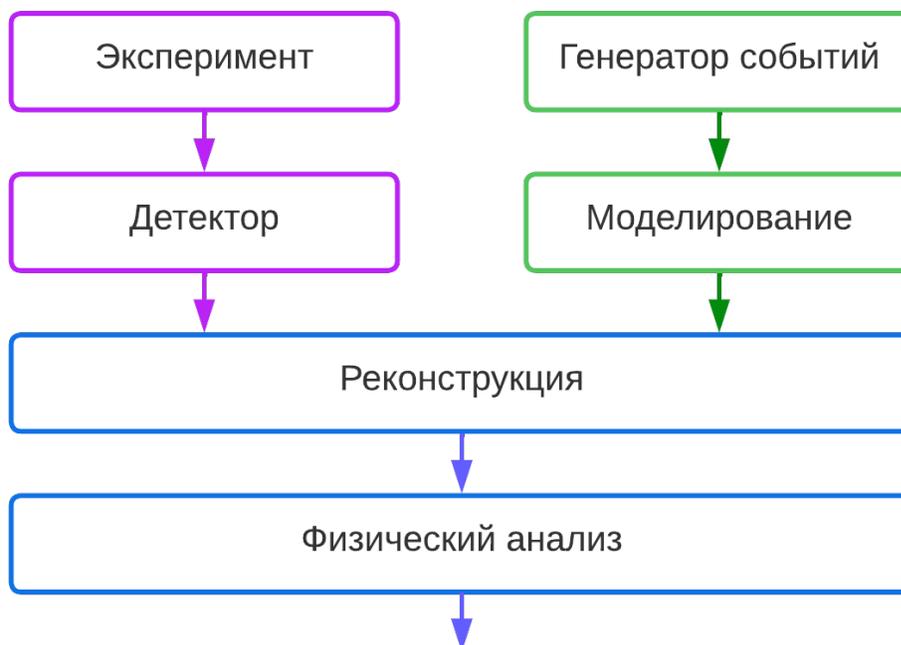


Рис. 1: Структура физического эксперимента на ускорительном комплексе

2.1 Эксперимент Супер $c\text{-}\tau$ фабрика

Супер $c\text{-}\tau$ фабрика — проект симметричного электрон-позитронного коллайдера Института ядерной физики им. Будкера СО РАН с энергией в центре масс от 3 до 7 ГэВ с проектной светимостью до $10^{35} \text{ см}^{-2}\text{с}^{-1}$.

Место столкновения пучков будет оборудовано универсальным детектором частиц. Измерением характеристик частиц, таких как импульс, энергия, траектория занимаются различные подсистемы детектора (Рис. 2).

Трековая система состоит из двух частей: внутреннего и внешнего трекера.

Внутренний трекер служит для увеличения эффективности регистрации треков частиц с импульсом порядка 50 МэВ/с и менее и определения траекторий короткоживущих частиц, например, K_S^0 -мезонов и Λ -барионов. В рамках проекта Супер $c\text{-}\tau$ фабрика рассматривается несколько опций реализа-

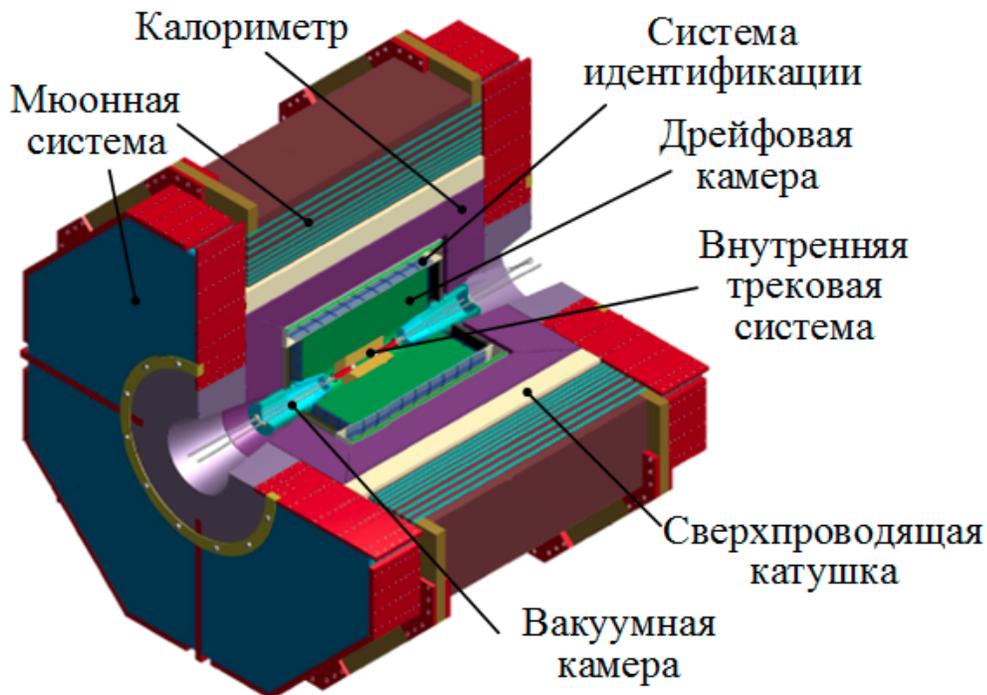


Рис. 2: Устройство детектора

ции внутреннего трекера: время-проекционная камера [5] и цилиндрический μ -RWELL [6].

Внешний трекер, который представлен дрейфовой камерой, занимается определением траектории частицы и её ионизационных потерь для идентификации типа частицы. В проекте Супер e - τ фабрики развиваются два направления: “классическая” дрейфовая камера, которая имеет аксиальные и стереослои (Рис. 3), а также дрейфовая камера, состоящая только из стереослоёв.

Калориметр служит для измерения энергии частицы и является главной системой регистрации гамма-квантов. В проекте Супер e - τ фабрики планируется использовать гомогенный электромагнитный калориметр, состоящий

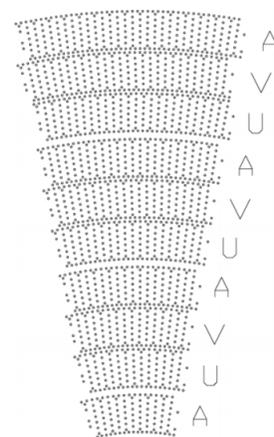


Рис. 3: Проволочная структура дрейфовой камеры

из кристаллов чистого йодистого цезия [7, 8].

Мюонная система позволяет отделить мюоны и пионы от других частиц, основываясь на их взаимодействии с веществом. Планируется построить мюонную систему как многослойную систему с детекторными слоями из полистирола с сцинтилляционной примесью, перемежающимися с поглощающим материалом, таким как сталь [9]. Предполагается девять слоёв, в которых толщина поглотителя 30 мм в первых трёх слоях, затем два слоя — 40 мм, оставшиеся слои — 80 мм.

Данные собранные системой идентификации позволяют сделать вывод о типе рожденных частиц. В детекторе на Супер $c\text{-}\tau$ фабрике будет использоваться система идентификации ФАРИЧ, которая представляет собой детектор черенковского излучения на основе “фокусирующего” аэрогеля (Рис. 4). При пролете через аэрогель частица порождает черенковское излучение, которое распространяется

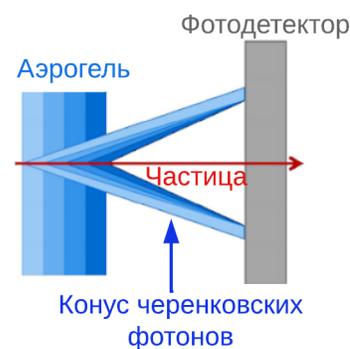


Рис. 4: Концептуальная схема ФАРИЧ

в виде конуса. Излучение образует на поверхности фотодетектора изображение в виде кольца, по радиусу которого определяется скорость частицы.

2.2 Фреймворк Aurora

Основой программного окружения в проекте Супер $c\text{-}\tau$ фабрика является фреймворк Aurora. Фреймворк представляет собой шаблон для реализации программной системы. Aurora является надстройкой над фреймворком Gaudi, при этом сохраняет всю архитектуру предоставляемую им [10].

Существуют аналогичные надстройки над Gaudi для других экспериментов в физике частиц: Athena (LHCb) [11], basf2 (Belle2) [12].

Фреймворк Gaudi ставит своей целью предоставление физикам программного обеспечения, которое является простым в использовании. Он предоставляет разработчикам и физикам архитектуру, которая состоит из компонентов, взаимодействующих друг с другом. Компоненты представляют собой блоки кода с четко определённым интерфейсом и функциональностью, т. е. реализация основана на объектно-ориентированном подходе.

Существует 3 основных вида компонентов (Рис. 5):

- алгоритм. Он принимает на вход данные, обрабатывает их и возвращает новые данные. В данном компоненте заключена основная логика работы программы. Обычно, алгоритм делегирует задачи на другие компоненты (инструменты, сервисы).
- инструмент. Инструменты представляют собой классы, которые помогают другим компонентам выполнять свою работу.
- сервис. Данный компонент инициализируется один раз и затем используется несколькими алгоритмами. Примерами сервисов являются сервис передачи сообщений, сервис генерации случайных чисел, job option сервис, который позволяет использовать уже готовый скрипт для запуска без перекомпиляции.

Программный код реализовывается на языке C++.

Для компонентов могут быть определены свойства — параметры, определяющие поведение компонента (Листинг 1). Они объявляются как поля класса алгоритма или инструмента. В приведенном примере первый аргумент — указатель на рассматриваемый объект класса, второй — имя свойства, по которому

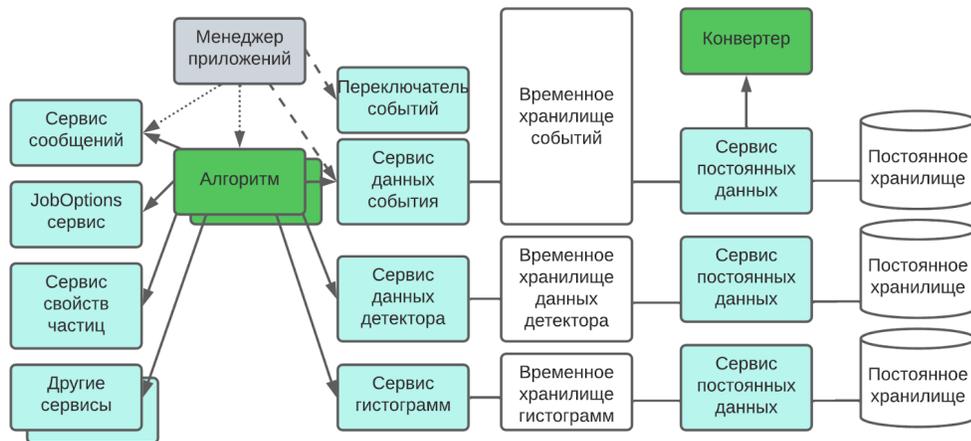


Рис. 5: Диаграмма взаимодействия компонентов Gaudi. Зеленым цветом выделены алгоритмы, голубым — сервисы, белым — хранилища данных

к нему может обращаться пользователь, третий — значение по-умолчанию, последний аргумент — текстовое описание свойства.

Листинг 1: Инициализация свойства инструмента Gaudi

```
1 Gaudi::Property<double> m_magneticField{this, "B", 1.5, "↵
  Detector magnetic field"};
```

Данные, подвергающиеся обработке компонентами Gaudi, хранятся в постоянных и временных хранилищах. Постоянные хранилища локализованы на диске и не имеют ограничений по времени хранения. Временные данные после завершения выполнения программы удаляются, но при необходимости они могут быть записаны в постоянное хранилище.

Созданный разработчиком алгоритм или инструмент (они имеют одинаковую конструкцию), наследуется от класса, предоставляемого фреймворком. Для алгоритмов родительским классом является `AuroraAlgorithm`, для инструментов — `AuroraTool`. Определяемый класс имеет один конструктор, принимающий два аргумента: уникальное во фреймворке имя алгоритма и указатель на его интерфейс. Родительский класс определяет несколько стандартных ме-

тодов. Методы `initialize()`, `execute()` и `finalize()` являются полностью виртуальными и нуждаются в реализации. Эти три метода возвращают `StatusCode`, который принимает значение `FAILURE` или `SUCCESS`, в зависимости от успеха выполнения. Если какой-либо из методов возвращает `FAILURE`, выполнение программы прекращается.

В начале работы программы менеджер приложений, прежде чем считать данные о событиях, инициализирует используемые алгоритмы и инструменты, с помощью метода `initialize()`. Также на данном этапе происходит установка свойств.

Основная логика алгоритма или инструмента реализована в методе `execute()`. Этот метод вызывается единожды для каждого события в алгоритме. Для инструментов данный метод может быть вызван несколько раз в алгоритмах, которые используют данный компонент.

Метод `finalize()` вызывается при завершении работы алгоритма или инструмента. В нём могут быть закрыты используемые конфигурационные файлы.

Фреймворк `Auroga` сохраняет архитектуру, предоставляемую `Gaudi`. Создание и настройка компонент, обмен данными между ними, планирование запуска алгоритмов происходит методами `Gaudi`. Но также существуют отличия:

- используются модель данных и единицы измерения, характерные для специфики эксперимента Супер s - τ фабрики;
- объединение нескольких событий происходит через `PileUpTool`;
- подключены внешние библиотеки, такие как `ROOT`, `Key4hep`, `Eigen`, `EvtGen`, `GSL`, `HepMC3` и т. д.;
- запуск алгоритмов реализован через скрипт на языке `Python`.

3 Первичные генераторы частиц

Первичные генераторы рождения и распадов частиц являются важным элементом программного обеспечения для экспериментов по физике элементарных частиц. Они случайным образом, исходя из вероятностей процессов, генерируют набор частиц, получаемых в результате распада. Параметры и распределения частиц задаются согласно заложенным теоретическим моделям. В процессе генерации распадов частиц учитывается кинематика процесса, для некоторых генераторов — производится адронизация и генерация распадов нестабильных частиц.

Генераторы классифицируются по физической направленности:

- генераторы распадов адронов (EvtGen [13], Pythia [14], HERWIG [15], ISAJET [16], SHERPA [17]);
- генераторы распадов лептонов (Tauola [18]);
- специализированные генераторы (KKMC [3, 4], Ariadne [19], ALPGEN [20]).
Данные, полученные с помощью их, отличаются большой точностью (доли процента) для определенных процессов.
- генератор одиночных частиц. “Стреляет” определенным типом частиц с заданными параметрами. Обычно используется для служебных целей, например, тестирования программного обеспечения.

3.1 Генераторы в фреймворке Aurora

В фреймворке Aurora уже реализованы интерфейсы для нескольких генераторов частиц: генератор одиночных частиц (Particle gun), EvtGen, Pythia, Tauola.

Particle gun генерирует одиночные частицы с заданными характеристиками. Пользователь имеет возможность задать тип частицы через её PID (Particle Identification — идентификационный номер частицы, согласно PDG ID [21]), импульс и направление, в котором вылетела частица из места столкновения (центра детектора).

EvtGen — это генератор, который имитирует распады тяжелых частиц, в первую очередь B - и D -мезонов. Он содержит ряд моделей распада для промежуточных и конечных частиц, но пользователь может самостоятельно задать распад с помощью специального конфигурационного файла.

Pythia предназначена для генерации событий физики высоких энергий. Программа способна моделировать столкновения e^+ , e^- , протон и анти-протон в различных комбинациях в диапазоне энергий от 10 ГэВ до 100 ТэВ.

Tauola — генератор, моделирующий распады τ -лептонов.

Исторически сложилось, что большинство генераторов частиц реализовано на языке Fortran77. Однако они не имеют единого стандарта реализации, т. е. процесс инициализации и запуска, а также формат возвращаемых данных у них различен. Подключать каждый генератор уникальным образом для пользователя неудобно, поэтому предложено использовать для этого паттерн проектирования абстрактной фабрики (Рис. 6). Благодаря данному паттерну, пользователь не зависит от способа инициализации и запуска каждого конкретного генератора событий, а также получает данные в едином формате.

Подробнее формат данных будет описан в разделе 3.2.2.

Интерфейсы генераторов имеют общую оболочку `INepMCProviderTool`, что позволяет использовать их единым образом. Пользователь в качестве свойства алгоритма `GenAlg` задает желаемый генератор с необходимыми ему параметрами. `GenAlg`, используя виртуальный метод `getNextEvent()`, реализованный в абстрактном классе `INepMCProviderTool`, собирает необходимое количество событий. Для каждого генератора реализуется оболочка, как инструмент `Gaudi`, которая реализует данный абстрактный класс. Все генераторы возвращают данные в формате `NepMC3`.

В качестве примера был разработан интерфейс для специализированного генератора `ККМС`.

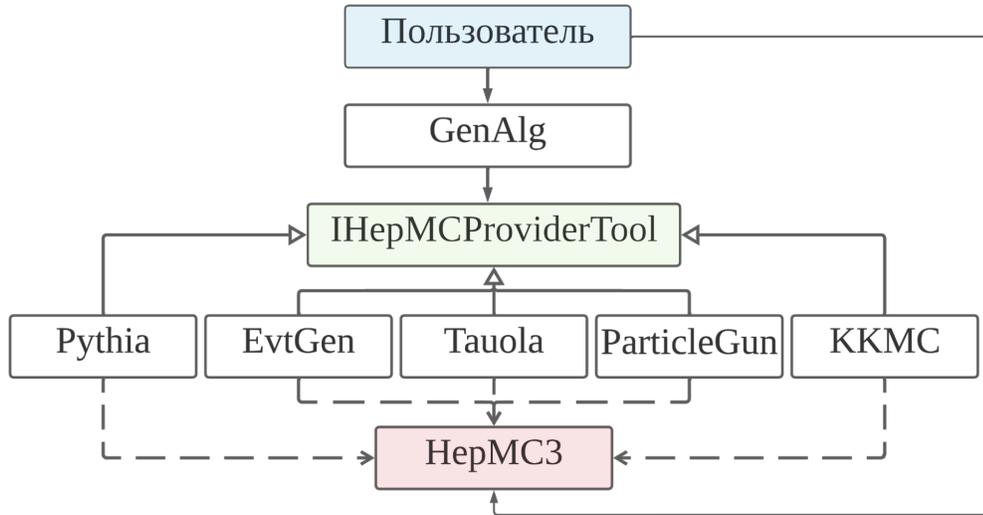


Рис. 6: Схема подключения генераторов в фрейворке

3.2 Генератор ККМС

ККМС — генератор частиц, моделирующий процессы $e^+e^- \rightarrow f\bar{f} + n\gamma$, где n — количество фотонов γ , f — фермионы $\mu, \tau, \nu, u, d, s, c, b$ [3, 4].

Генератор рассчитан на диапазон энергий в центре масс от рождения τ -лептона (3,55 ГэВ) до 1 ТэВ. Существует возможность задать продольную и поперечную поляризацию пучков. Для кварков и τ -лептонов включено соответствующее моделирование адронизации или распада.

Генератор реализован на языке Fortran77.

В 2022 году вышла новая версия генератора 5.00.01. Генератор, начиная с версии 4.17, имеет интерфейс на языке C++ (в отличии от предыдущих версий), однако для его интеграции в программное окружение Супер s - τ фабрики необходима дополнительная оболочка. Связано это с несоответствием формата данных, необходимости реализации ККМС как инструмента Aurora (подробнее об этом было сказано в разделе 3.1).

3.2.1 Устройство генератора

Код генератора разделен на модули:

- KK2F — реализация генератора ККМС. KK2F предоставляет пользовательский интерфейс, с помощью которого производится инициализация генератора и запуск генерации событий. В ходе инициализации задаются начальные значения для других используемых модулей и считываются конфигурационные файлы, содержащие описания распадов.
- bornv — модуль представляет собой базу данных со свойствами фермионов, таких как масса, заряд, цвет и другие параметры.
- glibk — утилиты (GLK (утилита для работы с гистограммами), математическая библиотека);
- DIZET — библиотека для моделирования электрослабых взаимодействий;

- TAUOLA — библиотека, генерирующая распады τ лептонов;
- PHOTOS — библиотека для распадов W и Z бозонов [22];
- JATSET — пакет для адронизации. В качестве инструмента для адронизации используется Pythia6.

В ходе инициализации происходит печать в файл и вывод на консоль параметров компонентов.

3.2.2 Тип выходных данных

Генератор по умолчанию выдает данные в формате HepEvt [23]. Класс, описывающий частицу, содержит следующие поля:

- количество частиц в событии;
- статусный код, показывающий это конечная частица или в дальнейшем она распадается. В зависимости от значения код имеет следующий смысл:
 - 0 — нулевая запись;
 - 1 — финальная частица, которая не распалась;
 - 2 — промежуточная частица, которая в дальнейшем распалась;
 - остальные значения зарезервированы для будущих стандартов и служебных записей.
- идентификатор частицы, согласно стандарту Pythia;
- индекс родительской частицы. Для начальных частиц равен 0, индексация по частицам начинается с 1;

- индексы дочерних частиц. Если частицы не распадаются индекс равен 0;
- 4-импульс частицы;
- масса частицы;
- координата вершины;
- время рождения частицы в единицах измерения мм/с ($\sim 3,33 \times 10^{12}$ с).

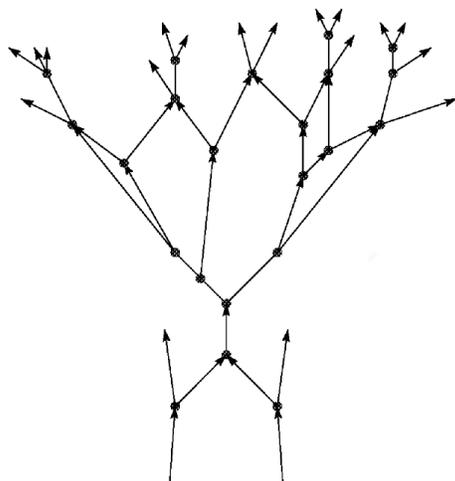


Рис. 7: Структура хранения события в HepMC3

Во фреймворке Аугога используется единый формат данных HepMC3 [23]. HepMC3 в настоящее время является стандартом выходных данных генераторов частиц.

Событие в данном формате представляется в виде графа, где вершины — точки столкновения частиц, дуги — частицы (Рис. 7). Основная информация, связанная с вершинами — это список входящих и исходящих частиц, их положение. Для частицы записывается четырех-вектор и идентификатор частицы, согласно стандарту PDG ID. Кроме того, опционально имеется возможность записывать информацию о поляризации, имя частицы, спин, массу и время жизни.

тор частицы, согласно стандарту PDG ID. Кроме того, опционально имеется возможность записывать информацию о поляризации, имя частицы, спин, массу и время жизни.

3.2.3 Реализованный интерфейс

Для подключения данного генератора к фреймворку Аугога разработан интерфейс на языке C++.

В качестве свойств Аулога-инструмента заданы следующие поля:

- два 4-х вектора пучков электронов и позитронов,
- разброс энергии центра масс,
- флаг, позволяющий записывать в выходной файл все частицы распада или только конечные,
- три конфигурационных файла,
- файл для записи конфигурации. По умолчанию конфигурация не будет доступна после завершения работы алгоритмов.

Полный список параметров, доступных для конфигурации, представлен в Приложении 1.

Конфигурационные файлы `kkdefaultFileName` и `taudecaytableFileName` содержат информацию о частицах, распадах и их вероятностях. Данные файлы не рекомендуется редактировать. Пользователь имеет возможность изменять файл `inputFileName`, задавая в нём интересующие его процессы.

В ходе инициализации ККМС происходит чтение конфигурационных файлов, устанавливаются параметры пучка, инициализация пакетов ККМС. В переопределяемом виртуальном методе `getNextEvent` генерируется событие и данные форматируются из формата `HerEvt` в `HerMC3`.

Подробнее процесс использования реализованного интерфейса описан в разделе [6.1](#) Пример использования.

4 Пакет для параметрического моделирования

Моделирование детектора необходимо для успешного выполнения физической программы, как на этапе разработки детектора, так и во время проведения экспериментов.

Можно выделить следующие основные задачи моделирования:

- расчёт ожидаемого отклика детектора на сигнальные и фоновые процессы;
- отладка методов анализа данных, определение наилучших критериев отбора событий;
- оценка ожидаемой точности проводимых физических анализов;
- оптимизация систем детектора;
- отладка программного обеспечения: реконструкции событий, пакетов для анализа.

Ниже изложена классификация моделирования, структура разработанного параметрического моделирования и приведены параметризации реализованных в моделировании подсистем детектора.

4.1 Виды моделирования детектора

По степени детализации взаимодействия частиц с веществом и формирования отклика детектора выделяется три основных вида моделирования: параметрическое, быстрое и полное. Возможны также их комбинации. Разра-

батываемый программный пакет `SctParSim` содержит в себе элементы параметрического и быстрого моделирования.

В параметрическом моделировании преобразование характеристик частиц задается набором аналитических функций. При этом происходит абстрагирование от конкретной реализации подсистемы. В одной из реализаций отклик задаётся с использованием случайных чисел, распределённых по нормальному распределению. Параметрами в данном случае будут являться ширина пика нормального распределения и пороговые значения срабатывания подсистемы.

В программном окружении проекта Супер c - τ фабрики такой вид моделирования представлен пакетом `SimpleSctParSim`.

Отклик детектора в быстром моделировании в большинстве случаев также задается функциями с параметрами, но они имеют более сложный вид и больший набор параметров. Быстрое моделирование связано уже с конкретной реализацией системы, в отличие от параметрического моделирования.

Существует несколько внешних программных продуктов для быстрого моделирования экспериментов на высокой энергии (≥ 100 ГэВ), например, PAPAS (The Parameterized Particle Simulation) [24] и Delphes [25]. PAPAS используется для моделирования столкновений электрон-позитронных пучков, а Delphes — адронных.

Полное моделирование позволяет получить отклик детектора, близкий к реальному. Основным инструментом полного моделирования детектора является пакет GEANT4 [26]. В ходе моделирования каждая частица, смоделированная генератором событий, итерационно проводится через вещество. На каждом шаге отклик разыгрывается случайным образом с учётом вероятности процессов. В пакете GEANT4 содержится большое количество физических моделей взаимодействия частиц с веществом, например, электромагнитные и адронные процессы, моделирование распадов, параметризация ливней.

Однако использование GEANT4 сопряжено с некоторыми трудностями. Такое моделирование имеет высокий порог входа и занимает много человеко-часов на написание программы, из-за необходимости задавать подробное описание детектора: геометрические размеры, описание материала каждой подсистемы, разрешение. GEANT4 выдаёт отклики детектора, для которых необходимо моделировать оцифровку и реконструкцию. Кроме того, само моделирование занимает более длительное время, чем параметрическое моделирование. Для сравнения было проведено измерение времени полного и параметрического моделирования 10 000 событий процесса инклюзивного распада J/ψ . Полное (параметрическое) моделирование в программном окружении Супер $c\text{-}\tau$ фабрики заняло в среднем 0,166 с (0,795 мс) со среднеквадратичным отклонением 0,148 с (0,441 мс) на событие, инициализация длилась 22,5 с (110 μ с) со среднеквадратичным отклонением 5,3 с (28 μ с).

Полное моделирование может выполняться не только для всего детектора, но и отдельно для каждой подсистемы. Также существуют программные пакеты для решения конкретных задач. Например, для расчетов взаимодействия ионов с веществом на низких энергиях (≤ 1 ГэВ) может быть использован SRIM (Stopping and Range of Ions in Matter) [27].

Использование различных типов моделирования можно представить в виде пирамиды (Рис. 8), где чем ниже расположен элемент, тем выше порог входа, сложнее использование, больше время, затрачиваемое на моделирование события, но и данные ближе к реальному эксперименту. В зависимости от задачи выбирается оптимальное соотношение между точностью и сложностью.

Для задач отладки алгоритмов реконструкции, проведения предварительных анализов, а также оценки эффективности использования различных подсистем детектора было решено разработать пакет для параметрического моделирования, который описан далее.

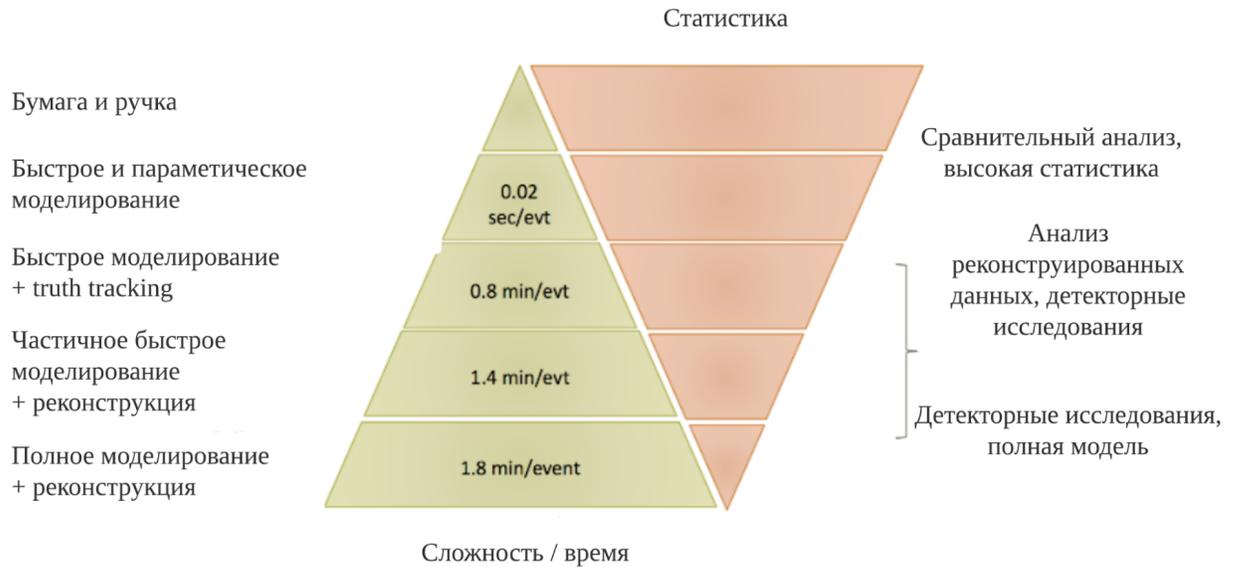


Рис. 8: Уровни моделирования детектора [28]

4.2 Схема работы программного пакета SctParSim

Следуя архитектуре, предоставляемой фреймворком Gaudi, и принципу разделения ответственности, разработанное параметрическое моделирование состоит из четырёх пакетов:

- **SctParSimAlg** — основной алгоритм параметрического моделирования. Представляет собой компонент-алгоритм в терминах фреймворка.
- **SctParSimCxx** — в данном пакете содержатся вспомогательные классы, в частности для работы с геометрией. Классы в данном пакете не являются компонентами Gaudi.
- **SctParSimTools** — содержит реализации подсистем детектора. На данный момент реализованы такие подсистемы как: дрейфовая камера, система идентификации частиц ФАРИЧ, калориметр и мюонная система. Их

параметризации описаны в следующем разделе. Каждая подсистема реализована как инструмент в терминах Gaudi.

- `SctParSimInterfaces`. Данный пакет содержит интерфейсы подсистем, реализованных в `SctParSimTools`.

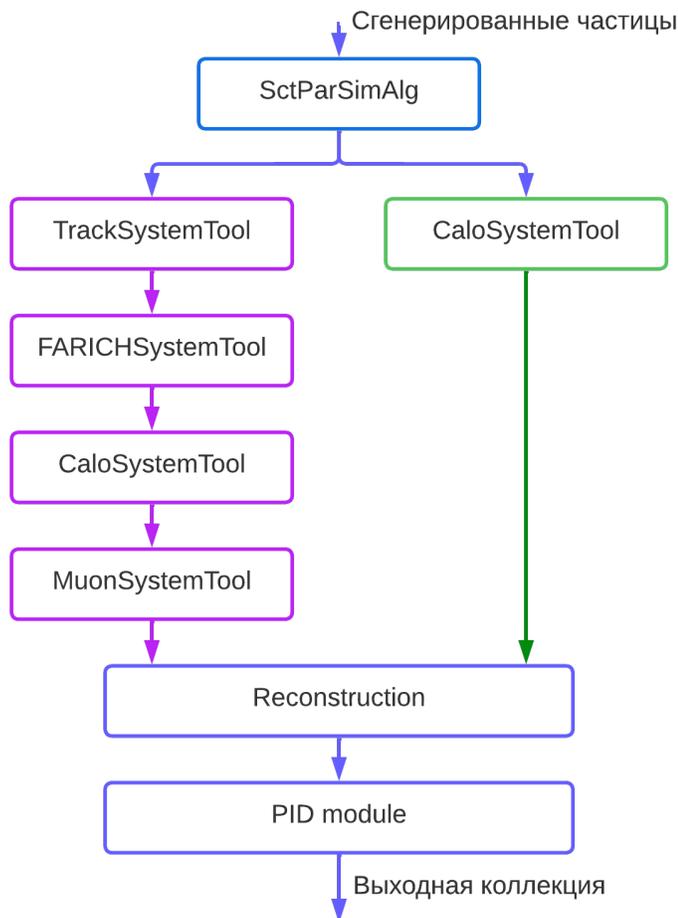


Рис. 9: Схема вызова инструментов в параметрическом моделировании

Подробная диаграмма классов представлена в Приложении 2.

Пакет `SctParSimAlg` содержит единственный класс, имеющий аналогичное имя и реализованный как Gaudi-алгоритм. Он проходит по всем частицам из входной коллекции, которая представляет собой n -мерный кортеж в формате

`ROOT::TTree`, вызывает для них подсистемы детектора в зависимости от электрического заряда частицы, затем собирает все отклики в выходную коллекцию (Рис. 9).

Также в `SctParSimAlg` находятся примеры работы с пакетом, с использованием других компонентов: генератора, средства записи в выходной файл и чтения из входного файла, пакета для анализа.

Подсистемы детектора в пакете `SctParSimTools` реализованы как Gaudi-инструменты.

Параметры основного алгоритма и подсистем детектора задаются в качестве свойств, что позволяет пользователю их конфигурировать из скрипта, запускающего алгоритмы.

В пакете `SctParSimCxx` на данный момент реализовано два класса для описания треков частиц: `SctHelix` для заряженных частиц и `SctLine` для нейтральных, а также реализовано пространство имён `Crosser`. В данном пространстве имён реализованы функции, которые рассчитывают координаты геометрического пересечения объектов (треков, цилиндров и цилиндрических слоёв). Класс `SctHelix` является наследником класса уже реализованного в программном окружении Супер c - τ фабрики `Helix`. Но в нём добавлен функционал для работы со скоростью частицы и временами достижения определенных точек.

Все классы из данного пакета покрыты тестами. Для каждого вида трека (винтовая линия, прямая) проверяются всевозможные пересечения их с цилиндрами и цилиндрическими слоями. Также для треков протестированы функции вычисления времени достижения определенной координаты, местоположения частицы по времени движения и длины. Данные тесты могут быть запущены как отдельно для пакета `SctParSimCxx`, так и совместно с тестами всего программного окружения.

4.3 Реализация инструментов

Были разработаны следующие подсистемы:

- дрейфовая камера,
- ФАРИЧ,
- калориметр,
- мюонная система.

Ниже приведена параметризация, используемая в разработанном программном обеспечении.

4.3.1 Дрейфовая камера

Параметризация дрейфовой камеры моделирует проволочную структуру. Моделируются 10 суперслоёв, состоящие из 4-х слоёв каждый, из которых 4 суперслоя являются аксиальными, а остальные — находятся под углом к оси детектора. Слои моделируются как цилиндрические, разбиения на отдельные ячейки нет.

Разрешение дрейфовой камеры по координате перпендикулярной оси детектора рассчитывается по следующей формуле [29]:

$$\sigma(p_t) = p_t \sqrt{(\xi p_t)^2 + \zeta^2}, \quad (1)$$

где p_t — импульс частицы в плоскости, перпендикулярной оси детектора,

коэффициенты:

$$\xi = \xi_0 \frac{L_0^2}{L^2} \sqrt{\frac{42}{N+1}} \quad (2)$$

$$\zeta = \frac{53,9 \times 10^{-3}}{\beta B} \times (1 + \cos^2 \theta)^{\frac{3}{4}} \times \frac{(1 + 0,038 \ln \frac{L_{tr}}{X_0})}{\sqrt{L X_0 \sin \theta}}, \quad (3)$$

где N — число измерений координаты, $\xi_0 = 2,12 \times 10^3$ рассчитано по результатам полного моделирования, L_0 — толщина дрейфовой камеры, L — расстояние, пройденное частицей, β — скорость частицы в единицах скорости света, B — величина магнитного поля, θ — угол отклонения от оси дрейфовой камеры, L_{tr} — длина траектории частицы в дрейфовой камере, X_0 — радиационная длина (Рис. 10). Число измерений координаты рассчитывается как число слоёв проволоочек, пересеченных частицей. Частицы с мягкими треками, имеющие менее 6 измерений координаты, не имеют отклика дрейфовой камеры, так как 5 и менее измерений недостаточно для восстановления спирали.

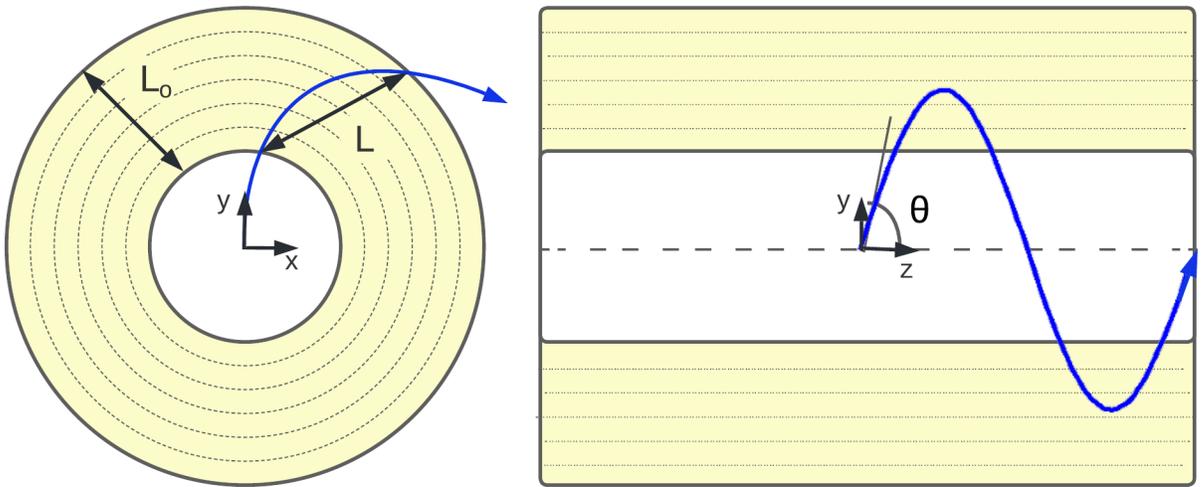


Рис. 10: Иллюстрация к формулам 1, 2, 3

Угловое разрешение дрейфовой камеры σ_θ складывается из разрешения, связанного с многократным рассеянием, σ_θ^{MS} и пространственного разре-

ния σ_θ^0 :

$$\sigma_\theta^{MS} = \frac{0,014z}{\beta p} \sqrt{\frac{L_{tr}}{X_0}} \left(1 + 0,038 \ln \frac{L_{tr}}{X_0} \right) \quad (4)$$

$$\sigma_\theta^0 = \begin{cases} 1,281 \times 10^{-3}/N_z, & \theta \leq 40,16^\circ \\ 3,08 \times 10^{-3} \sin^2 \theta / \sqrt{N_z}, & \theta > 40,16^\circ \end{cases} \quad (5)$$

$$\sigma_\theta = \sqrt{\sigma_\theta^{02} + \sigma_\theta^{MS2}} \quad (6)$$

$$\sigma_p = p \sqrt{\xi^2 p_t^2 + \zeta^2 + \text{ctg}^2 \theta \times \sigma_\theta^2}, \quad (7)$$

где N_z — число измерений z -координаты (количество пересечений стереослоёв). Угол $40,16^\circ$ соответствует максимальному углу вылета частицы, при котором она попадает в торец дрейфовой камеры.

Помимо расчёта измеренного импульса частицы, дрейфовая камера содержит две опции идентификации частиц: по потерям энергии на ионизацию и с помощью подсчёта кластеров ионизации.

Средние ионизационные потери рассчитываются согласно модифицированной формуле Бете-Блоха, которая даёт лучшее значение χ^2 для подгонки экспериментальных данных работы дрейфовой камеры ВаВаг, чем классическая формула [30]:

$$BB(\beta\gamma) = \frac{p_0}{\beta^{p_4}} \left(p_1 - \beta^{p_4} - p_6 \log \frac{1 + p_2(\beta\gamma)^{p_3}}{1 + p_5(\beta\gamma)^{p_3}} \right). \quad (8)$$

Параметры имеют следующие значения $p_0 = 22,88$, $p_1 = 17,99$, $p_2 = 3,655 \times 10^{-19}$, $p_3 = 8,513$, $p_4 = 2,012$, $p_5 = 0,2848$, $p_6 = 0,2906$.

Разрешение по ионизационным потерям рассчитывается по формуле [30]:

$$\frac{\sigma \frac{dE}{dx}}{\frac{dE}{dx}} = p_0 \left(\frac{N}{N_0} \right)^{p_1} \left(\frac{L_{tr}}{L_1} \right)^{p_2} (1 + p_3(x - p_4)^2) (1 + p_5(p_t - p_6)^2), \quad (9)$$

где $N_0 = 40$, $L_1 = 100$, $x = \frac{BB(\beta\gamma) - BB_{min}(\beta\gamma)}{BB_{min}(\beta\gamma)}$.

Число кластеров ионизации разыгрывается по распределению Пуассона с математическим ожиданием:

$$\lambda = \frac{dE}{dx} / \frac{dE}{dx_{min}} \delta_{cl} L_{tr}, \quad (10)$$

где $\delta_{cl} = 12.5/\text{см}$ для газовой смеси $He : iC_4H_{10} = 90 : 10$. В отклик дрейфовой камеры записывается значение, разыгранное по распределению Пуассона, нормализованное на длину трека.

Отклик дрейфовой камеры содержит импульс частицы, потери энергии на ионизацию, количество кластеров ионизации, время нахождения частицы в дрейфовой камере. Т. к. параметризация внутреннего трека отсутствует, то точкой рождения частицы принимается центр детектора, т. е. место столкновения пучков.

4.3.2 ФАРИЧ

ФАРИЧ — это детектор черенковского излучения на основе «фокусирующего» аэрогеля.

Работа подсистемы ФАРИЧ хорошо известна, поэтому в параметрическом моделировании используются результаты полного моделирования с использованием GEANT4.

Для расчёта отклика сначала определяется ожидаемое число фотоэлектронов N_0 по углу, под которым частица вылетела, и скорости частицы (Рис. 11). Зарегистрированное число фотоэлектронов N_{ph} разыгрывается по распределению Пуассона с математическим ожиданием N_0 . Если N_{ph} меньше 4, то считается, что эллипс восстановлен не был, и частица не имеет значения параметра β , измеренного в системе ФАРИЧ. Для зарегистрированных частиц

параметр β размазывается согласно данным из файла, полученного полным моделированием.

На выход подсистема отдает β параметр частицы, количество зарегистрированных и ожидаемое количество фотоэлектронов.

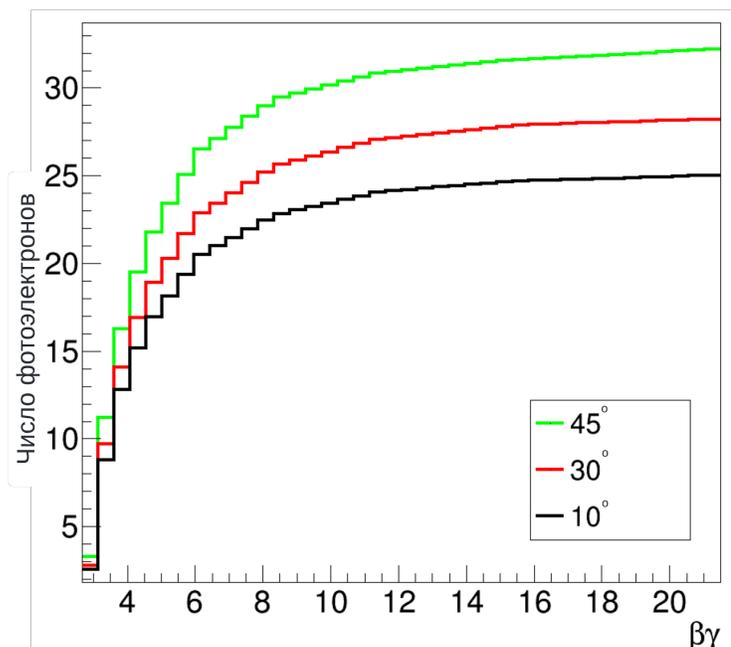


Рис. 11: Ожидаемое число фотоэлектронов для разных углов

4.3.3 Калориметр

В программном обеспечении моделируется электромагнитный калориметр на основе кристаллов чистого йодистого цезия.

Разрешение калориметра рассчитывается по формуле [8]:

$$\sigma_E = E \sqrt{0.000134^2 + \left(\frac{0.00066}{\sqrt{E}}\right)^2 + \left(\frac{0.00081}{\sqrt[4]{E}}\right)^2}, \quad (11)$$

где первое слагаемое отвечает за неоднородности отклика детектора, второе —

за шумы, третье — за флуктуации частиц в ливне.

Размер кластера для ливнеобразующих частиц (электронов и фотонов) составляет 0,15 м, для неливнеобразующих — 0,045 м. Если энергия частицы ниже пороговой, то частица не будет зарегистрирована.

После работы калориметра срабатывает алгоритм реконструкции, т.е. сшивки данных, полученных трековой системой, и калориметрических кластеров. Объединение происходит по следующему алгоритму:

- кластеры, которые пересекаются геометрически, объединяются в один кластер;
- для каждой частицы с учётом точки входа её в калориметр подбором определяется соответствующий кластер. Занятый кластер исключается из рассмотрения для следующих частиц;
- перед возвратом происходит перерасчет параметров кластера:
 - энергия — сумма энергий, выделенных в составляющих кластера;
 - время — минимальное время срабатывания калориметра;
 - размер — объединение площадей кластеров;
 - положение кластера — среднее взвешенное с учетом выделенных энергий в кластерах.

В выходном файле записывается размер выделившейся энергии в калориметре, положение и размер кластера.

4.3.4 Мюонная система

Мюонная система работает, используя результаты stand-alone моделирования, которое было проведено с помощью G4Beamline [31], основанном на GEANT4.

Система моделировалась восемью цилиндрическими слоями поглотителя, чередующихся с восемью чувствительными слоями из полистерола. Поглотителем являлось железо. Торцы цилиндра были представлены аналогично слоями. Также были смоделированы калориметр и магнитная катушка, другие подсистемы не рассматривались. Моделировались мюоны и пионы с импульсом до 4 ГэВ/с.

На выход подсистема отдает номер слоя, который последним зарегистрировал частицу.

5 Обзор пакета для отбора событий

Физический анализ событий требует применения множества алгоритмов и сложных методов статистического анализа. Но физический анализ проводится не на всех “сырых” данных, полученных с подсистем детектора. Перед этим отбираются частицы, которые являются кандидатами на участие в конкретном распаде. Затем происходит реконструкция распада, т. е. отбираются частицы, которые действительно могли участвовать в распаде и удовлетворяющие определённым критериям. Какой распад будет изучаться и критерии определяются экспериментом и теорией, которую необходимо проверить. Такому отбору подвергаются не только данные с детектора, но и данные моделирования. Их сравнение и позволяет делать заключения о соответствии эксперимента и теории.

В программном окружении проекта Супер $c\text{-}\tau$ существует инструмент для отбора событий, который и будет описан далее.

5.1 Инструмент для отбора событий

Инструмент для отбора событий `ParticleCombinerAlg` [32] реализован в рамках фреймворка `Aurora`, что позволяет использовать его совместно с разработанным интерфейсом для генератора ККМС и параметрическим моделированием. Данный алгоритм перебирает пары частиц, которые являются кандидатами в промежуточные частицы распада, и объединяет подходящие. Последовательное применение алгоритма объединения частиц используется для создания произвольных деревьев распада.

Описание распада и критериев отбора задаётся в строковом формате. Правила записей частиц зафиксированы в генераторе EvtGen и являются стандартными. Примеры описания распада нейтрального D -мезона представлены в Листинге 2. Слева от стрелки указана родительская частица, справа — дочерние. Также возможна запись каскадных распадов, где дочерние частицы не являются конечными.

Листинг 2: Пример описания распада в ParticleCombinerAlg

```
1 "D0 -> K- pi+"
2 "D0 -> [rho0 -> pi+ pi-] pi0"
3 "D0 -> [rho0 -> pi+ pi-] [rho+ -> pi+ [pi0 -> gamma gamma]]"
```

Критерии отбора конечных частиц также записываются в строковом формате (Листинг 3).

Листинг 3: Пример описания критериев отбора в ParticleCombinerAlg

```
1 "E < 1 and pt > 0.3"
```

После отбора частиц для распада, с помощью алгоритма NtupleAlg пользователь может задать интересующие его кинематические характеристики, которые будут записаны в выходной файл. Кинематические характеристики, доступные пользователю для записи, определяются пакетом AnaVarManager: импульс, энергия, инвариантная масса, полярный и азимутальный угол вылета частицы, заряд, вероятности разделения K/π , K/p и π/η .

На выход алгоритм выдаёт кинематические характеристики отобранных частиц в формате N-tuple. Данный формат представляет собой кортежи, состоящие из чисел.

6 Результаты работы

6.1 Пример использования

Запуск параметрического моделирования происходит с помощью пользовательского скрипта, написанного на языке Python.

В данном разделе будет представлен пример использования генератора событий ККМС и разработанного параметрического моделирования. Полная версия скрипта представлена в Приложении 3.

Первым этапом происходит запуск генератора событий (Листинг 4).

Листинг 4: Инициализация генератора событий ККМС

```

1 kkmc = ККМС('SignalProvider')
2 #kkmc.ККМСOutputFileName = 'output'
3 gen = GenAlg('EvtGenAlg', SignalProvider=kkmc)

```

Первая строка создает экземпляр генератора. Вторая строка иллюстрирует процесс конфигурирования генератора. Обращение к конкретному свойству инструмента или алгоритма происходит, как к атрибуту класса, через точку. В третьей строке создается экземпляр алгоритма генерации частиц `GenAlg`, которому в качестве параметра `SignalProvider` передается генератор.

Следующий этап — моделирование детектора (Листинг 5). Для этого необходимо создать экземпляр алгоритма `SctParSimAlg` и опционально изменить параметры подсистем детектора. Полный набор параметров представлен в Приложении 4. Для изменения параметра конкретной подсистемы, сначала происходит обращение к подсистеме детектора, а затем к параметру.

 Листинг 5: Инициализация и конфигурация параметрического моделирования

```

1 sct_alg = SctParSimAlg('SctAlg')
2 sct_alg.CaloSystemTool.caloClSize = 0.015

```

Последним смысловым блоком является настройка инструмента для отбора событий. Для этого используется несколько алгоритмов.

Сначала с помощью алгоритма `EventLoader` происходит чтение и отбор конечных частиц из временного хранилища данных (Листинг 6). В первой строке происходит создание экземпляра алгоритма. Во второй строке задается имя коллекции для чтения. Сгенерированные частицы находятся в коллекции `'MCParticles'`, реконструированные, т. е. прошедшие через детектор — в коллекции `'Particles'`. Третья строка задает имя коллекции, под которым данные будут записаны во временное хранилище. В последней строке задаётся список конечных частиц, необходимых для отбора. Для того, чтобы отбирать частицу и ее античастицу в строке после имени указывается подстрока `'cc'`. Также имеется возможность сразу задать ограничения на характеристики отобранных частиц в формате `['gamma', 'p > 0.4']`.

 Листинг 6: Подключение алгоритма `EventLoader`

```

1 evlo = EventLoader('EvtLoader')
2 evlo.pcl.Path = 'Particles'
3 evlo.pListMap.Path = 'Lists1'
4 evlo.plists = [['gamma']]

```

Далее происходит реконструкция распадов на основе отобранных частиц с помощью алгоритма `ParticleCombinerAlg` (Листинг 7). Во второй строке задаётся распад в строковом формате. Параметр `cutStr` задаёт критерии отбора на кинематические характеристики. Параметр `selfConj` имеет значение `True`, если реконструируемая частица нейтральная, иначе — `False`. Седьмая и восьмая строки — имена входной и выходной коллекции во временном хранилище.

Листинг 7: Подключение алгоритма ParticleCombinerAlg

```

1 cmbr = ParticleCombinerAlg('Cmbr',
2     decStr = 'pi0 -> gamma gamma',
3     cutStr = 'E > 0.5',
4     selfConj = True
5 )
6
7 cmbr.pListMapI = evlo.pListMap.Path
8 cmbr.pListMapO.Path = 'Lists2'

```

Последний этап — запись кинематических характеристик частиц, участвующих в распаде, в формате N -tuple (Листинг 8). Во второй строке задано имя списка материнских частиц. В третьей строке указано имя выходного файла. В строках 4–6 перечислены характеристики частиц в строковом формате для записи в выходной файл. Последнее строковое значение в наборе показывает, параметры каких частиц должны быть записаны, они помечены знаком '^'. Пустая строка указывает на то, что параметры записываются для родительской частицы.

Листинг 8: Подключение алгоритма NtupleAlg

```

1 tupl = NtupleAlg('piTuple')
2 tupl.listName = 'pi0'
3 tupl.fileName = 'scttuple/tup'
4 tupl.vars = [['px_mc', 'py_mc', 'pz_mc', 'E', 'pi0 -> ^gamma ^←
5     gamma'],
6     ['M', ' '],

```

Для запуска всей цепочки алгоритмов используется менеджер приложений ApplicationMgr (Листинг 9). Он последовательно запускает алгоритмы с заданными свойствами. Свойство 'TopAlg' представляет собой перечисление запускаемых алгоритмов. Свойство 'EvtMax' определяет количество генерируемых событий. Пятая строка запускает менеджер приложений.

Листинг 9: Запуск алгоритмов

```

1 options= {
2     'TopAlg' : [gen, edm, sct_alg, evlo, cmbr, tupl],
3     'EvtMax' : 100,
4 }
5 ApplicationMgr(**options)

```

6.2 Полученные результаты

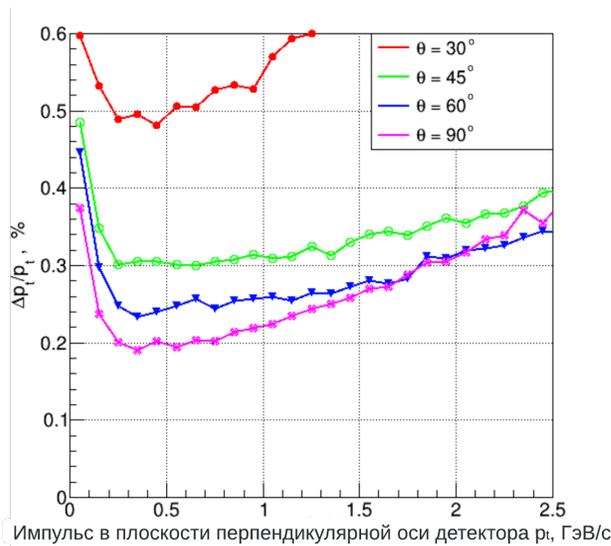


Рис. 12: Зависимость разрешения дрейфовой камеры от импульса

Для демонстрации корректности работы разработанных подсистем в параметрическом моделировании были построены характерные распределения.

Разрешение дрейфовой камеры (Рис. 12) соответствует результатам полного моделирования, представленным в статье “The drift chamber project for the Super Charm-Tau Factory detector” [33].

Также для дрейфовой камеры были построены зависимости ионизационных потерь (Рис. 14) и количество кластеров ионизации на единицу длины (Рис. 13) от импульса.

Зависимость β -фактора частицы от импульса изображена на Рис. 15. Данные, полученные с помощью параметрического моделирования, соответствуют постеру, посвященному системам идентификации на Супер c - τ фабрике на VCI2019 [34].

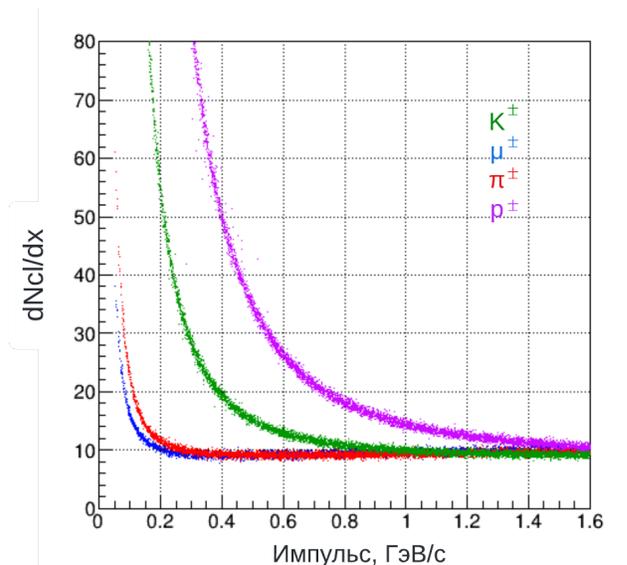


Рис. 13: Зависимость количества кластеров ионизации в дрейфовой камере от импульса

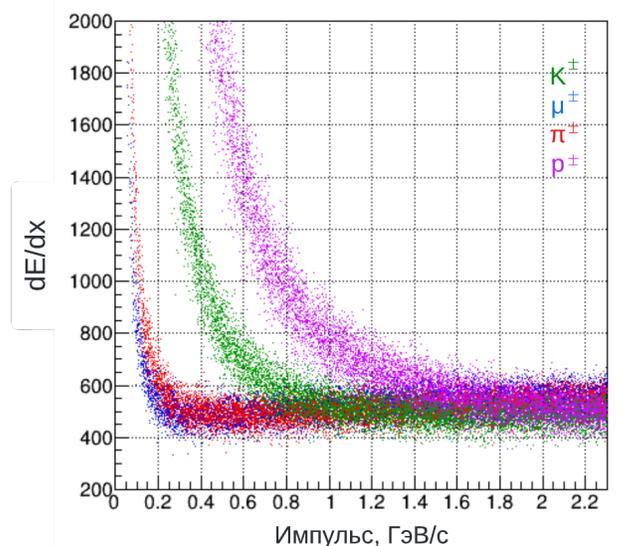


Рис. 14: Зависимость ионизационных потерь частицы в дрейфовой камере от импульса

Для калориметра реализована реконструкция с учётом геометрического пересечения кластеров. С помощью генератора EvtGen был смоделирован распад одиночного π^0 в два фотона. С увеличением импульса π^0 , угол между фотонами уменьшается (Рис. 16), калориметрические кластеры начинают пересекаться. Из-за пересечений кластеров, эффективность регистрации падает (Рис. 17). Резкое падение эффективности регистрации происходит при меньшем импульсе с увеличением размера кластера.

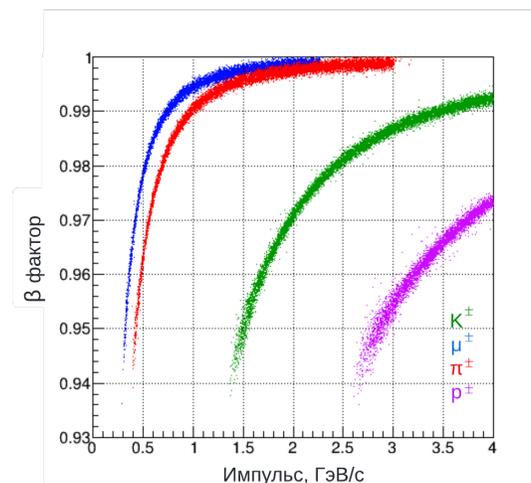


Рис. 15: Зависимость скорости частицы, регистрируемой в ФАРИЧ, от импульса

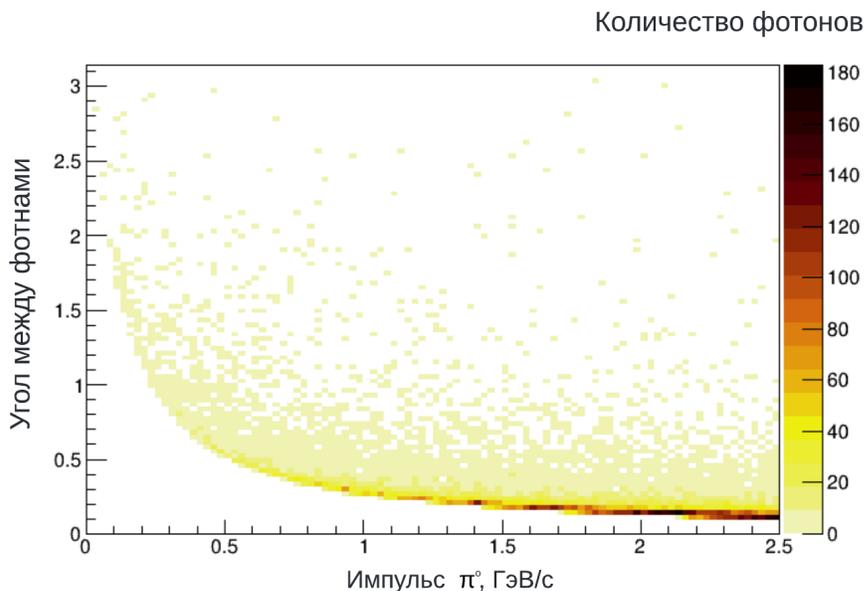


Рис. 16: Угол между сгенерированными фотонами в лабораторной системе отсчёта в зависимости от импульса π^0

Мюоны обладают большей проникающей способностью, поэтому они долетают до дальних слоёв мюонной системы от центра детектора. Данный эффект наблюдается на построенных распределениях (Рис. 18). Пионы также регистрируются в мюонной системе, однако они застревают в первых слоях поглотителя.

Разработанный пакет для параметрического моделирования может быть использован для физического анализа. Пример полного скрипта, производящего генерацию, моделирование детектора и отбор событий, представлен в Приложении 3.

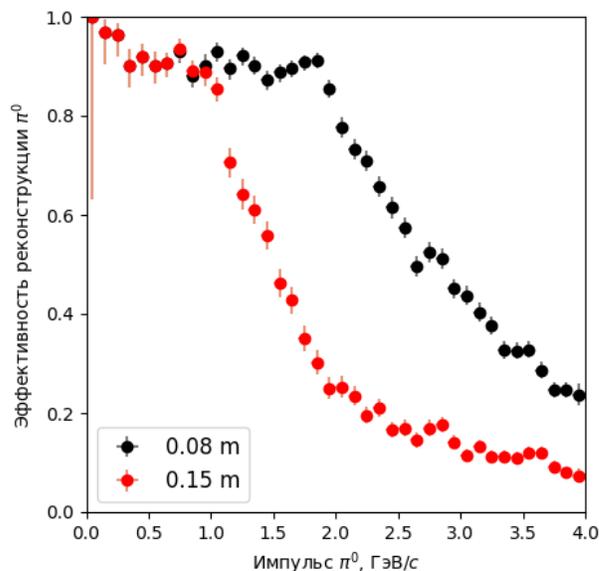


Рис. 17: Эффективность регистрации π^0 в зависимости от энергии для разных размеров кластера

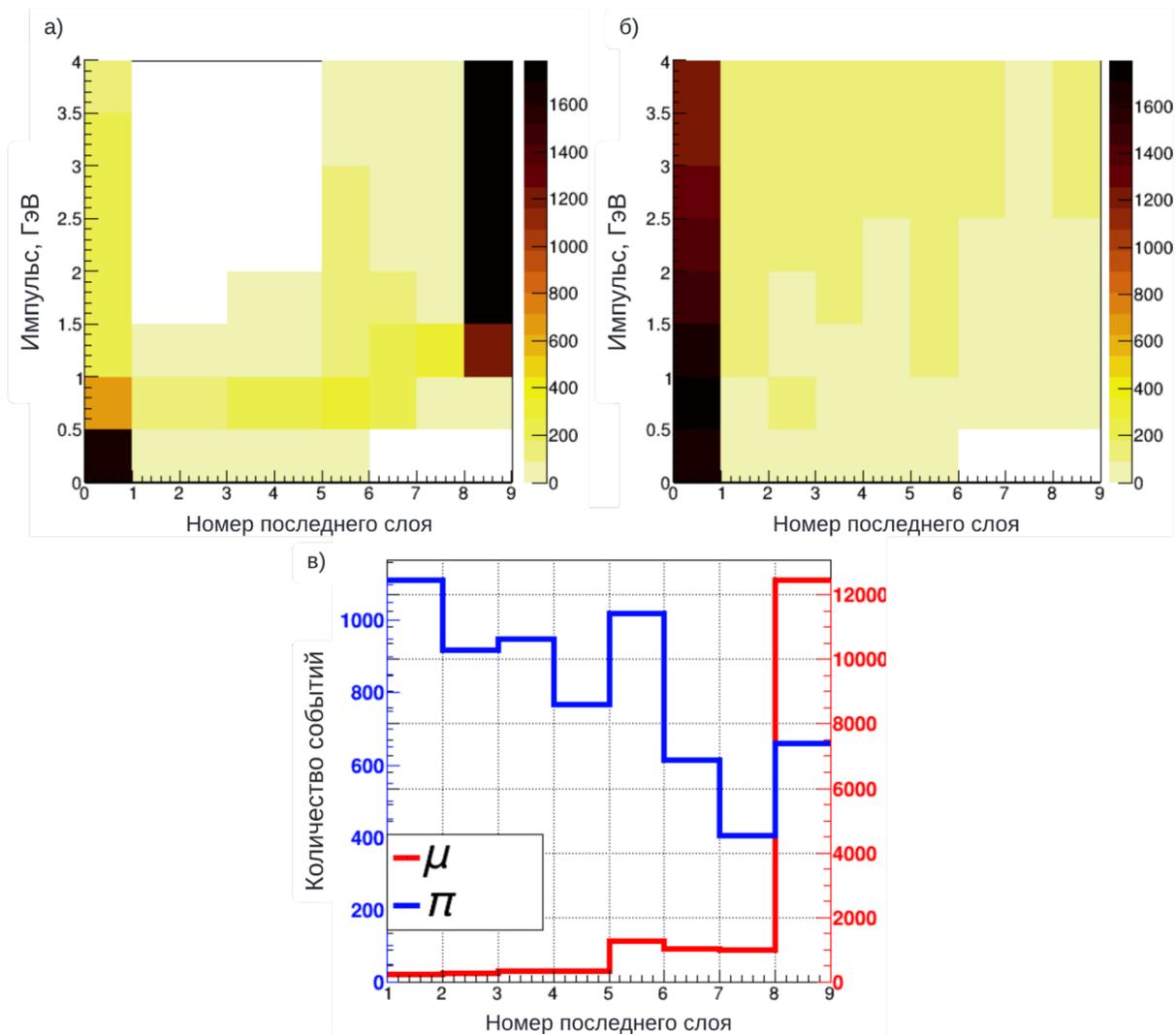
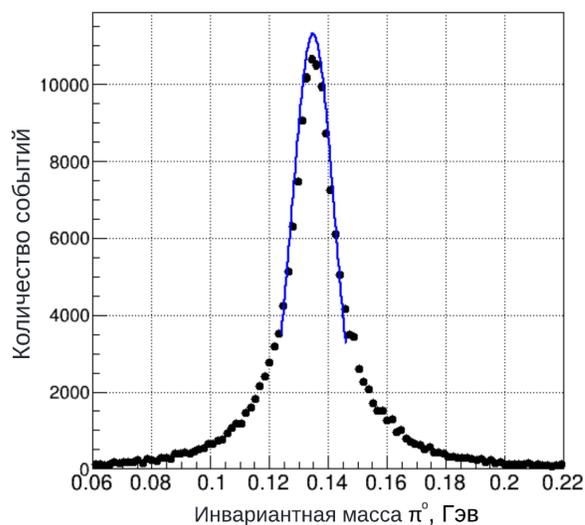
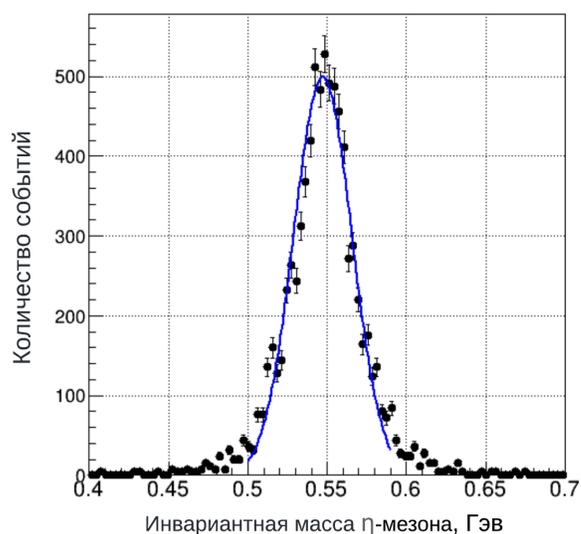


Рис. 18: Распределение количества событий, когда а) мюон б) пион долетает до определенного слоя в мюонной системе а, б) в зависимости от импульса в) для частиц с импульсом до 4 ГэВ

В качестве примера использования были построены инвариантные массы π^0 и η -мезона. Для этого был сгенерирован инклюзивный распад J/ψ . С помощью инструмента для отбора событий были отобраны фотоны, которые являлись кандидатами в распад π^0 . Кинематическая реконструкция не проводилась. Параметры разрешения калориметра соответствуют значениям по умолчанию (Приложение 4). Для определения σ данные были подогнаны нормальным распределением.



Инвариантная масса π^0 , ГэВ
 $\sigma = 7,1$ МэВ



Инвариантная масса мезона η -мезона, ГэВ
 $\sigma = 17$ МэВ

Рис. 19: Пример использования параметрического моделирования и инструментов для анализа. Точки — данные, линия — результат аппроксимации нормальным распределением

На странице внутреннего ресурса проекта Супер c - τ фабрики подготовлено и опубликовано краткое руководство пользователя и описание параметрического моделирования.

Страница с описанием параметрического моделирования содержит:

- список публикаций по данной теме;
- полный список параметров конфигурации с значениями по умолчанию (Приложение 4);

- краткое описание параметризации каждой реализованной подсистемы;
- описание выходной коллекции.

Руководство пользователя содержит примеры скриптов и инструкции по запуску параметрического моделирования совместно с встроенными во фреймворк Аугога инструментами: генераторами частиц, инструментами для чтения и записи файлов, инструментами для анализа.

7 Заключение

В ходе работы над выпускной квалификационной работой были выполнены следующие задачи:

- реализован алгоритм параметрического моделирования средствами C++ в программном окружении проекта Супер s - τ фабрика с использованием фреймворка Aurora;
- разработаны вспомогательные классы для работы с геометрией треков частиц и подсистем детектора;
- внедрены параметризации калориметра, трековой и мюонной системы;
- внедрена реконструкция частиц с учётом слияния электромагнитных калориметрических кластеров;
- подготовлено и опубликовано краткое руководство пользователя на wiki-странице внутреннего ресурса проекта Супер s - τ фабрики;
- опубликованы несколько примеров использования параметрического моделирования (с использованием генератора частиц (EvtGen, генератор одиночных частиц), инструментов для анализа);
- предложен механизм подключения первичных генераторов событий в программном окружении проекта Супер s - τ фабрика;
- реализован интерфейс первичного генератора ККМС.

Конечным результатом является программное обеспечение для параметрического моделирования детектора и интерфейс подключения генератора ККМС.

По теме выпускной квалификационной работы опубликованы четыре статьи ([35], [36], [37], [38]), также представлены выступления на молодёжной конференции “Физика элементарных частиц и космология” и международной научной студенческой конференции (МНСК-21, МНСК-22).

Примеры использования параметрического моделирования представлены в следующих статьях: [34], [39], [40], [41].

В дальнейшем планируется перенести разработанный интерфейс генератора в основной репозиторий проекта. Параметрическое моделирование будет дополнено реконструкцией данных, полученных из систем идентификации.

8 Список литературы

1. Супер Чарм - Тау Фабрика: физическая программа / М. Н. Ачасов [и др.]. — Новосибирск, Москва, 2022 — с. 147.
2. Super Charm Tau Factory [Электронный ресурс] / BINP. — 2022. — URL: <https://sct.inp.nsk.su/>
3. The Precision Monte Carlo Event Generator KK For Two-Fermion Final States In e^+e^- Collisions / S. Jadach, B.F.L. Ward, Z. Was // Computer Physics Communications. 2000. март. No 130. с. 260–325. DOI: 10.1016/S0010-4655(00)00048-5. arXiv: 9912214 [hep-ph]
4. M. Chrzaszcz, S. Jadach, A. Siodmok KKMC Status // Physics Performance meetings. 19.08.2020
5. Simulation of different options of the Inner Tracker for Novosibirsk Super Charm-Tau Factory Detector / T. V. Maltsev, L. I. Shekhtman, A. V. Sokolov, V. K. Vadakeppattu // EPJ Web of Conferences. 2019. No 212. с. 1–5. DOI: 10.1051/epjconf/201921201011
6. The micro-Resistive WELL detector: a compact spark-protected single amplification-stage MPGD / G. Bencivenni, R. De Oliveira, G. Morello, M. Poli Lener // Journal of Instrumentation,. 2015. T. 10. No 2. с. 1–8. DOI: 10.1088/1748-0221/10/02/p02008
7. Simulation of the CsI crystal calorimeter of the detector of charm-tau factory in Novosibirsk / V. L. Ivanov [и др.] // Journal of Instrumentation. 2015. T. 15. No 7. с. 1–8. DOI: 10.1088/1748-0221/15/07/c07026

8. Epifanov D. Pure CsI calorimeter for Super C-Tau factory // Super C-Tau factory workshop. BINP. 05.2018.
9. Proposal of the Muon System for the Super Tau-Charm Factory / T Ugllov // EPJ Web of Conferences. 2019. No 212. с. 1—4. DOI: 10.1051/epjconf/201921201010
10. Gaudi: User guide. – 2001. – с. 213
11. The Athena control framework in production, new developments and lessons learned / P. Calafiura [и др.] // Computing in high energy physics and nuclear physics. 2004. No. 2. с. 456 – 458. DOI: 10.5170/CERN-2005-002
12. The Belle II Core Software / T. Kuhr [и др.] // Computing and Software for Big Science. 2019. No. 3. с. 1 – 12. DOI: 10.1007/s41781-018-0017-9
13. EvtGen [Электронный ресурс]: рабочая программа. URL: <https://evtgen.hepforge.org/>
14. Pythia 6.4. Physics and Manual / T. Sjostrand, S Mrenna, P. Skands // Journal of High Energy Physics. 2006. март. No. 5. DOI: 10.1088/1126-6708/2006/05/026. arXiv: 0603175 [hep-ph]
15. HERWIG Information [Электронный ресурс]: рабочая программа. URL: <https://www.hep.phy.cam.ac.uk/theory/webber/Herwig/>
16. ISAJET 7.88 – A Monte Carlo Event Generator for p p, pbar p, and e+ e- Interactions / F.E. Paige, S.D. Protopopescu, H. Baer, X. Tata [Электронный ресурс]: рабочая программа. URL: <http://www.nhn.ou.edu/isajet/>
17. Sherpa [Электронный ресурс]: рабочая программа. URL: <https://sherpa-team.gitlab.io/>

18. TAUOLA of τ -lepton decays – framework for hadronic currents, matrix elements and anomalous decays / M. Chrzaszcz [и др.] // Computer Physics Communications. 2018. No. 232. с. 220 – 236. DOI: 10.1016/j.cpc.2018.05.017. arXiv: 1609.04617 [hep-ph]
19. Ariadne version 4 – A program for simulation of QCD cascades implementing the colour dipole model [Электронный ресурс]: рабочая программа. URL: <http://home.thep.lu.se/~leif/triadne/>
20. ALPGEN V2.14 [Электронный ресурс]: рабочая программа. URL: <http://mlm.home.cern.ch/mlm/alpgen/>
21. Particle data group / M. Tanabashi [и др.] [Электронный ресурс] – 2019 – URL: <http://pdglive.lbl.gov/Viewer.action>
22. PHOTOS Monte Carlo: a precision tool for QED corrections in Z and W decays / P. Golonka, Z. Was // European Physical Journal C. 2006. No. 45. с. 97 – 107. DOI: 10.1140/epjc/s2005-02396-4. arXiv: 0506026 [hep-ph]
23. The HepMC C++ Monte Carlo event record for High Energy Physics / M. Dobbs, J. B. Hansen // Computer Physics Communications. февраль. 2001. т. 134. No 1. с. 41 – 46. DOI: 10.1016/S0010-4655(00)00189-2
24. HEP-FCC. PAPAS (PARametrized PArticle Simulation) [Электронный ресурс]: рабочая программа. URL: <https://github.com/HEP-FCC/papas>
25. DELPHES 3, A modular framework for fast simulation of a generic collider experiment / J. de Favereau [и др.] // JHEP. 2014. т. 02. с. 057. DOI: 10.1007/JHEP02(2014)057. arXiv: 1307.6346 [hep-ex]
26. Geant4 – a simulation toolkit / S. Agostinelli [и др.] // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers,

- Detectors and Associated Equipment. 2003. июль. т. 506, No 3. с. 250—303.
DOI: 10.1016/s0168-9002(03)01368-8
27. Ziegler J. F. SRIM (Stopping and Range of Ions in Matter) [Электронный ресурс]: рабочая программа. 2013. URL: <http://srim.org/>
28. M. Selvaggi Fast simulation for future collider experiments // RDFFA meeting. – Bologna. 04.07.2017
29. Тодышев К. Ю. Меморандум по модели ДК для c - τ фабрики. Новосибирск, 2019. с. 6
30. BaBar DCH dE/dx calibration and a new technique of energy loss calculation / K. Todyshev // BABAR Analysis Document. 2007. т. 1698
31. G4beamline Particle Tracking in Matter Dominated Beam Lines / T. J. Roberts [и др.] // EPAC 2008. 2008. июнь. WEPP120. с. 2776—2779.
32. Пакет программ для отбора событий в эксперименте на Супер С-Тау фабрике / В. А. Майборода. 2021. с. 1–54
33. The drift chamber project for the Super Charm-Tau Factory detector I. Yu. Basoka, T. V. Bedarevaa et al. // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. сентябрь. 2021. Т. 1009. DOI: 10.1016/j.nima.2021.165490
34. Barnyakov A. Yu. PID system for Super С- τ Factory at Novosibirsk // The 15th Vienna Conference of Instrumentation. Vienna University of Technology. 18.02.2019

35. Software framework for the Super Charm-Tau factory detector project / M. S. Belozyorova [и др.] // EPJ Web of Conferences. 2021. т. 251. No 03017. с. 1–8. DOI: 10.1051/epjconf/202125103017
36. Computing Environment for the Super-Charm-Tau Factory Detector Project / M. S. Belozyorova [и др.] // CEUR Workshop Proceedings. 2021. т. 3041. с. 375–380. DOI: 10.54546/MLIT.2021
37. М. С. Белозерова Разработка параметрического моделирования эксперимента на Супер чарм-тау фабрике с использованием фреймворка Aurora // Международная студенческая конференция. 2021. т. 59. с. 252
38. М. С. Белозерова Разработка пакета программного обеспечения для параметрического моделирования детектора Супер с- τ фабрики // Международная студенческая конференция. 2020, т. 58, с. 19
39. Particle identification for the Super Charm-Tau Factory at Novosibirsk / A. Y. Barnyakov [и др.] // Nuclear Instruments and Methods in Physics Research Section A. 2020. апр. т. 958. с. 4. DOI: 10.1016/j.nima.2019.162352.43
40. The Super с- τ Factory identification system options / A. Y. Barnyakov [и др.] // EPJ Web of Conferences. 2019. июнь. т. 212. с. 11. DOI: 10.1051/epjconf/201921201012.
41. Overview of PID options for experiments at the Supr Charm-Tau Factory / A. Y. Barnyakov [и др.] // Journal of Instrumentation. 2020. апр. т. 15. с. 11. DOI: 10.1088/1748-0221/15/04/C04032

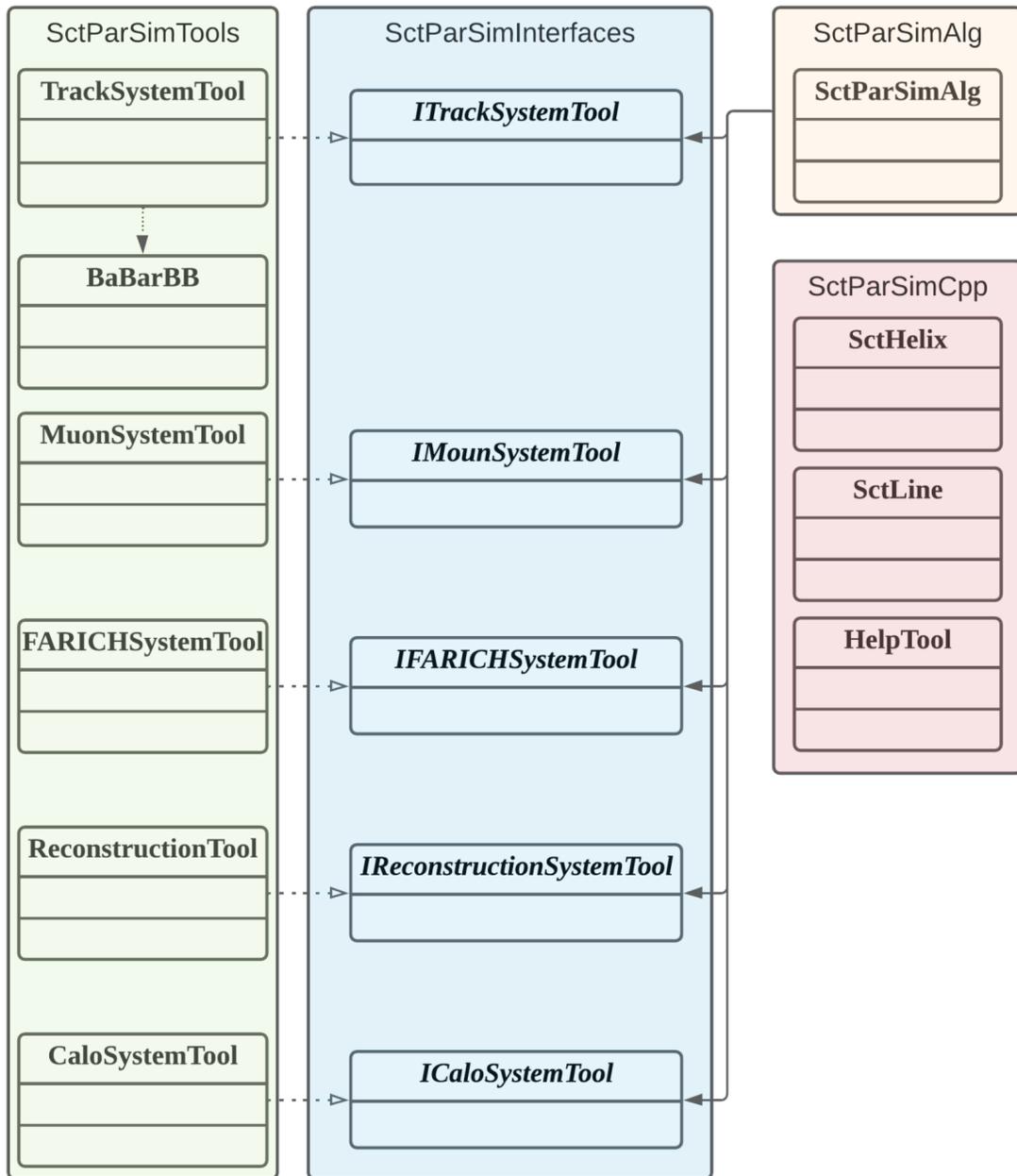
9 Приложение

Приложение 1. Список параметров, доступных пользователю для конфигурации генератора ККМС

Имя параметра	Описание	Значение по-умолчанию
leftBeamP4	4-импульс пучка электронов, ГэВ	{0, 0, 4.1/2, 4.1/2 + 5e-4}
rightBeamP4	4-х импульс пучка позитронов, ГэВ	{0, 0, -4.1/2, 4.1/2 + 5e-4}
beamEspread	Разброс энергии центра масс, ГэВ	0
onlyFinalParticles	Флаг записи в файл только конечных частиц	true
KKdefaultFileName	Конфигурационный файл (список распадов)	"./KK2f_defaults"
KKdefaultConfigFileName	Конфигурационный файл описания генератора	"./demo.input.QUICK"
outputFileName	Выходной файл с описанием конфигурации	"/dev/null"
pythiaDirective	Конфигурация Pythia	"MSTU(21)=1;"
varyingWeightEvents	Константный/вариантивный вес событий	true
Rres	Экспериментальный R, используемый для моделирования скорости распада	1
ISR	Включено/выключено ISR	true
FSR	Включено/выключено FSR	true
typeISR	Тип ISR	0
IFI	Включено/выключено IFI	true
photonEmission	Включено/выключено излучение фотонов конечных кварков	true

Имя параметра	Описание	Значение по-умолчанию
Hadronization	Включение/выключение адронизации	true
QCDfactor	Включение/выключение КХД фактора конечного состояния	true
polarizationRight Beam	Поляризация пучка позитронов	{0, 0, 0}
polarizationLeft Beam	Поляризация пучка электронов	{0, 0, 0}
otherParams	Другие параметры	{{}}
configFile	Конфигурационный файл	"0"
Part_d	Включена/выключена генерация d-кварка	true
Part_u	Включена/выключена генерация u-кварка	true
Part_s	Включена/выключена генерация s-кварка	true
Part_c	Включена/выключена генерация c-кварка	true
Part_b	Включена/выключена генерация b-кварка	true
Part_ne	Включена/выключена генерация электронного нейтрино	true
Part_mu	Включена/выключена генерация мюона	true
Part_nm	Включена/выключена генерация мюонного нейтрино	true
Part_tau	Включена/выключена генерация τ	true
Part_nt	Включена/выключена генерация τ -нейтрино	true

Приложение 2. Диаграмма классов пакета SctParSim



Приложение 3. Пример пользовательского скрипта запуска моделирования детектора

```

1 # Подключение библиотек
2 from Configurables import ApplicationMgr
3 from Gaudi.Configuration import *
4 from Configurables import GenAlg, EvtGenInterface
5 from Configurables import SctTauDataSvc
6 from Configurables import PodioOutput, HepMCtoEDMConverter
7
8 from Configurables import SctParSimAlg
9 from Configurables import EventLoader
10 from Configurables import ParticleCombinerAlg, NtupleAlg, ←
    NTupleSvc
11
12 from Configurables import Gaudi__ParticlePropertySvc
13 from PathResolver import PathResolver
14
15 #####
16 # Event generation - генерация событий
17 #####
18 podioevent = SctTauDataSvc("EventDataSvc")
19
20 # Particle service
21 particlePropertySvc = Gaudi__ParticlePropertySvc(
22     "ParticlePropertySvc", ParticlePropertiesFile=
23     PathResolver.FindDataFile(
24     'GenParticleData/ParticleTable.txt'))
25
26 # KKMC - генератор событий
27 kkmc = KKMC('SignalProvider')
28 #kkmctool.onlyFinalParticles = False
29 #kkmctool.KKMCOutputFileName = '/dev/null'
30
31 gen = GenAlg('EvtGenAlg', SignalProvider=kkmc)
32 gen.hepmc.Path = 'hepmc'
33
34 # Преобразование формата данных HepMC3 в PODIO
35 edm = HepMCtoEDMConverter("Converter")
36 edm.hepmc.Path=gen.hepmc.Path
37 edm.genparticles.Path="allGenParticles"
38 edm.genvertices.Path="allGenVertices"
39

```

```

40 #####
41 # SctParSim - параметрическое моделирование
42 #####
43 sct_alg = SctParSimAlg('SctAlg')
44 # пример изменения параметров подсистемы
45 #sct_alg.CaloSystemTool.caloClSizeEGamma = 0.2
46
47 #####
48 # Analysis - отбор событий
49 #####
50 evlo = EventLoader('EvtLoader')
51 evlo.pcl.Path = 'Particles' # Имя ветки для чтения во входном ROOT ←
    файле-
52 evlo.pListMap.Path = 'Lists1'
53 evlo.plist = [['gamma']] # Список частиц для дальнейшего анализа
54
55 # Select particle combinations
56 cmbr = ParticleCombinerAlg('Cmbr',
57     decStr = 'pi0 -> gamma gamma', # Распад
58     cutStr = 'E > 0.5', # Критерий отбора
59     selfConj = True # Если родительская частица нейтральная - True, ←
        иначе - False
60 )
61
62 cmbr.pListMapI = evlo.pListMap.Path
63 cmbr.pListMapO.Path = 'Lists2'
64
65 # Кинематические характеристики для сохранения в n-tuple
66 tupl = NtupleAlg('piTuple')
67 tupl.listName = 'pi0'
68 tupl.fileName = 'scttuple/tup'
69
70 # Список параметров, которые будут записаны в выходной файл
71 tupl.vars = [['px_mc', 'py_mc', 'pz_mc', 'E', 'pi0 -> ^gamma ^←
    gamma'],
72     ['M', ' '],
73 ]
74
75 tupl.pListMapI.Path = cmbr.pListMapO.Path
76 NtupleSvc(Output = ["scttuple DATAFILE='tup.root' OPT='NEW' TYP←
    ='ROOT'"])
77
78 options= {
79     'TopAlg' : [gen, edm, sct_alg, evlo, cmbr, tupl],
80     'EvtSel' : 'NONE',
81     'ExtSvc' : [particlePropertySvc, podioevent],
82     'EvtMax' : eventNumber, # количество событий
83     'StatusCodeCheck' : True,

```

```
84     'AuditAlgorithms' : True ,
85     'AuditTools'      : True ,
86     'AuditServices'   : True ,
87     'OutputLevel'     : INFO ,
88     'HistogramPersistency' : 'ROOT',
89 }
90
91 ApplicationMgr(**options)
```

Приложение 4. Список параметров, доступных пользователю для конфигурации параметрического моделирования

Таблица 1: Пакет SctParSimAlg

Имя параметра	Описание	Значение по-умолчанию
B	Магнитное поле детектора, Тл	1.5
mostProbMass	Масса наиболее вероятной частицы, ГэВ	0.13957

Таблица 2: Пакет TrackSystemTool

Имя параметра	Описание	Значение по-умолчанию
trackRhoMin	Внутренний радиус подсистемы, м	0.1
trackRhoMax	Внешний радиус подсистемы, м	0.8
trackZMin	Минимальная z-координата подсистемы, м	0
trackZMax	Максимальная z-координата подсистемы, м	1
trackMinPt	Порог регистрации импульса, ГэВ	0.05
trackPtProb	Вероятность регистрации для разных импульсов, (ГэВ, вероятность)	{{0.1, 0.8}, {0.3, 0.9}, {1, 0.95}, {10, 0.99}}
trackRadLen	Радиационная длина, м	187
trackResParPT	Параметр параметризации для ху-проекции	0.00212
trackResParPZ	Параметр параметризации для z-проекции	0.001281, 0.00308

Имя параметра	Описание	Значение по-умолчанию
trackLayerAx	Толщина и расположение слоев анодных проволочек, мм	{{6.306, 217.306}, {6.644, 227.1}, {7.165, 246.906}, {6.564, 341.938}, {6.794, 352.06}, {7.14, 371.992}, {7.388, 382.95}, {6.651, 467.57}, {6.823, 477.718}, {6.968, 488.097}, {7.12, 498.701}, {7.274, 509.535}, {6.768, 636.322}, {6.898, 646.501}, {7.007, 656.957}, {7.121, 667.581}, {6.791, 750.730}, {6.902, 761.027}, {6.995, 771.472}, {7.091, 782.061}}
trackLayerSt	Толщина и расположение стерео слоев, мм	{{6.473, 280.136}, {6.747, 290.136}, {7.182, 310.863}, {7.486, 321.938}, {6.603, 405.941}, {6.799, 416.04}, {7.104, 436.741}, {7.314, 447.606}, {6.741, 533.35}, {6.859, 543.615}, {7.026, 554.088}, {7.161, 564.762}, {6.778, 584.801}, {6.919, 595.108},

Имя параметра	Описание	Значение по-умолчанию
		{7.039, 605.606}, {7.163, 6169.289}, {6.746, 689.948}, {6.865, 700.185}, {7.041, 720.09}, {7.165, 730.775}}

Таблица 3: Пакет FARICHSystemTool

Имя параметра	Описание	Значение по-умолчанию
farichRhoMin	Внутренний радиус подсистемы, м	0.82
farichRhoMax	Внешний радиус подсистемы, м	0.9
farichZMin	Минимальная z-координата подсистемы, м	1.02
farichZMax	Максимальная z-координата подсистемы, м	1.273
farichHoleR	Радиус отверстия подсистемы	0.3
parSimFarichFileName	Путь до файла с результатами полного моделирования ФАРИЧ	./pi_ms_f1_mpp c2_px3_d200_ mla4_graph2d.root

Таблица 4: Пакет CaloSystemTool

Имя параметра	Описание	Значение по-умолчанию
caloRhoMin	Внутренний радиус подсистемы, м	1.09
caloRhoMax	Внешний радиус подсистемы, м	1.55
caloZMin	Минимальная z-координата подсистемы, м	1.293
caloZMax	Максимальная z-координата подсистемы, м	1.86
caloCosthmax	Максимальный косинус для попадания в подсистему	0.9
caloClSize	Радиус калориметрического кластера, м	0.045
caloClSizeE Gamma	Радиус калориметрического кластера для электронов и фотонов, м	0.15
caloEMinBarrel	Порог регистрации по энергии в барельной части подсистемы, ГэВ	0.015
caloEMinEndcup	Порог регистрации по энергии в боковой части подсистемы, ГэВ	0.015
caloResPar	Параметры параметризации	{1.34e-2, 0.066e-2, 0.0, 0.82e-2}

Таблица 5: Пакет MuonSystemTool

Имя параметра	Описание	Значение по-умолчанию
muonRhoMin	Внутренний радиус подсистемы, м	1.87
muonRhoMax	Внешний радиус подсистемы, м	2.15
muonZMin	Минимальная z-координата подсистемы, м	1.88
muonZMax	Максимальная z-координата подсистемы, м	2.16
parSimMuonFile NameMu	Путь до файла с результатами полного моделирования для мюонов	./g4beamline_mu_plus_100k_parse.root
parSimMuonFile- NamePi	Путь до файла с результатами полного моделирования для пионов	./g4beamline_pi_plus_100k_parse.root