

Разработка пакета программного обеспечения для параметрического моделирования детектора Супер С-Тау фабрики

Белозерова Мария Сергеевна

Новосибирский государственный университет
Физический факультет
Кафедра физико-технической информатики

Научный руководитель: Разуваев Г.П.

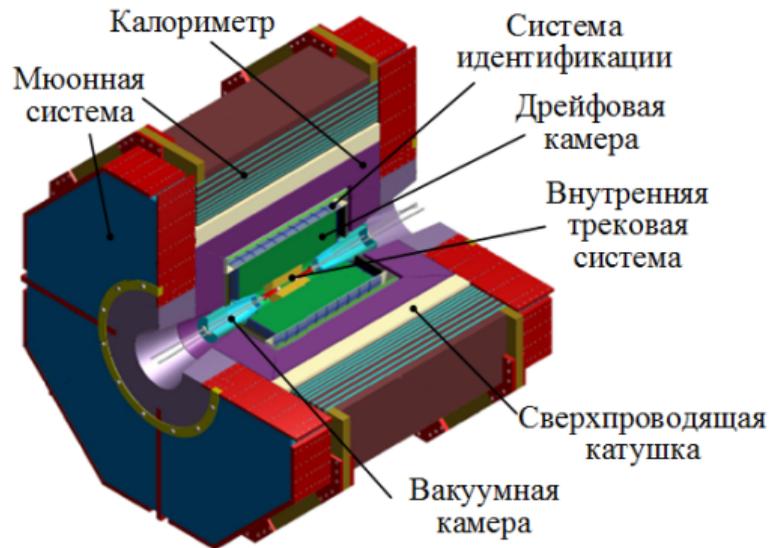
Новосибирск, 2020

- Супер c - τ фабрика
- Постановка цели и задач
- Программное обеспечение
- Результаты
- Заключение

- Электрон-позитронный коллайдер
- Энергия пучков $\sqrt{s} = 2-5$ ГэВ
- Светимость $10^{35} \text{ c}^{-1} \text{ см}^{-2}$
- Продольная поляризация пучков

Физическая программа

- Физика D -мезонов
- Запрещенные/подавленные распады c
- Параметры Мишеля в τ -распадах
- Адронные распады τ
- Возбужденные кварконии
- ...



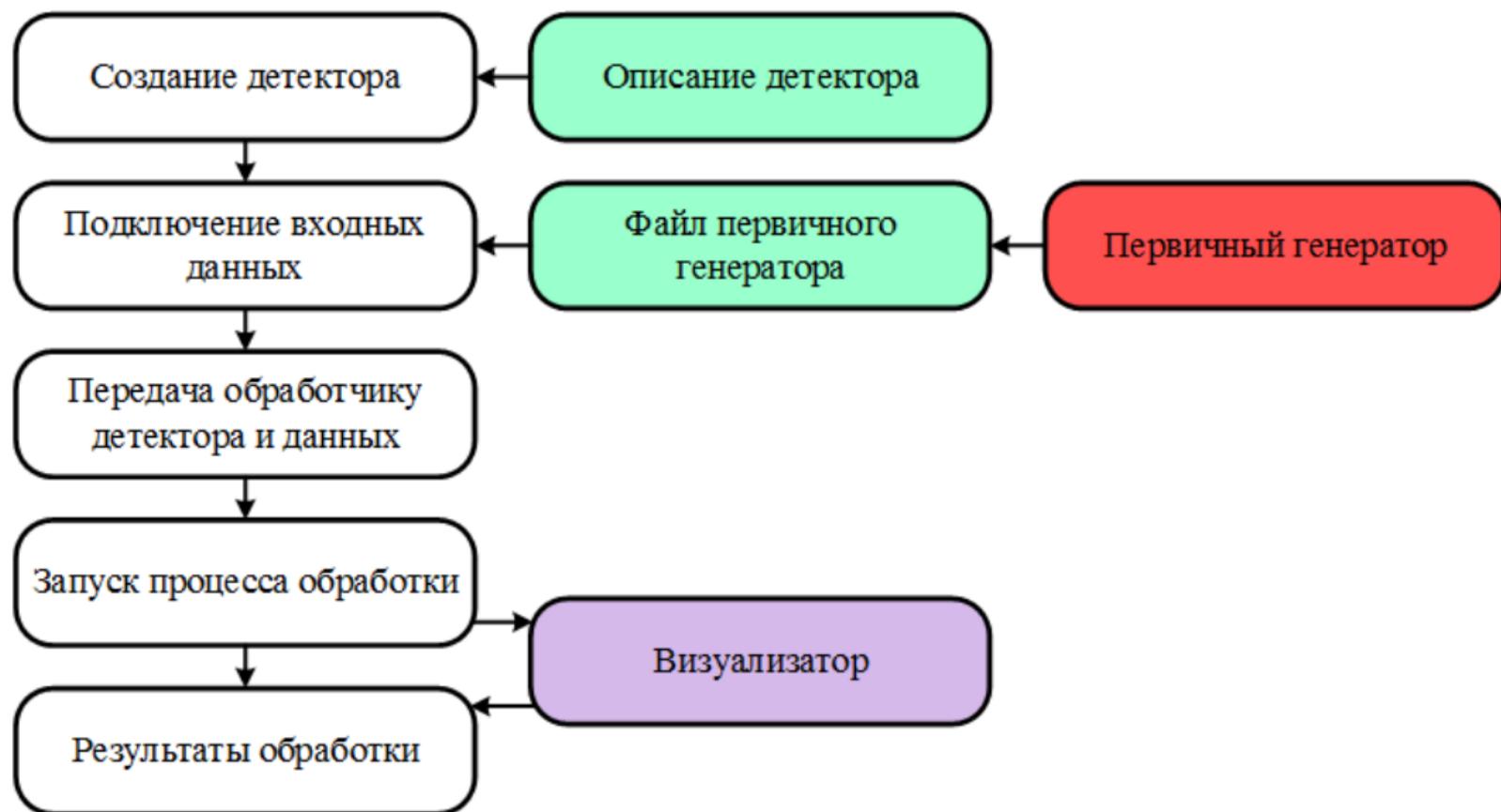
Цель

Разработать пакет программного обеспечения для параметрического моделирования Супер s - t фабрики

Задачи

- Обновить параметризацию дрейфовой камеры
- Внедрить систему идентификации частиц FDIRC
- Внедрить мюонную систему
- Разработать пользовательский интерфейс для изменения параметров моделируемого детектора
- Проверить правильность проведения частиц через объем детектора
- Разработать реконструкцию частиц
- Проверить отклики разрабатываемых подсистем на их соответствие заложенному описанию

Рабочая цепочка

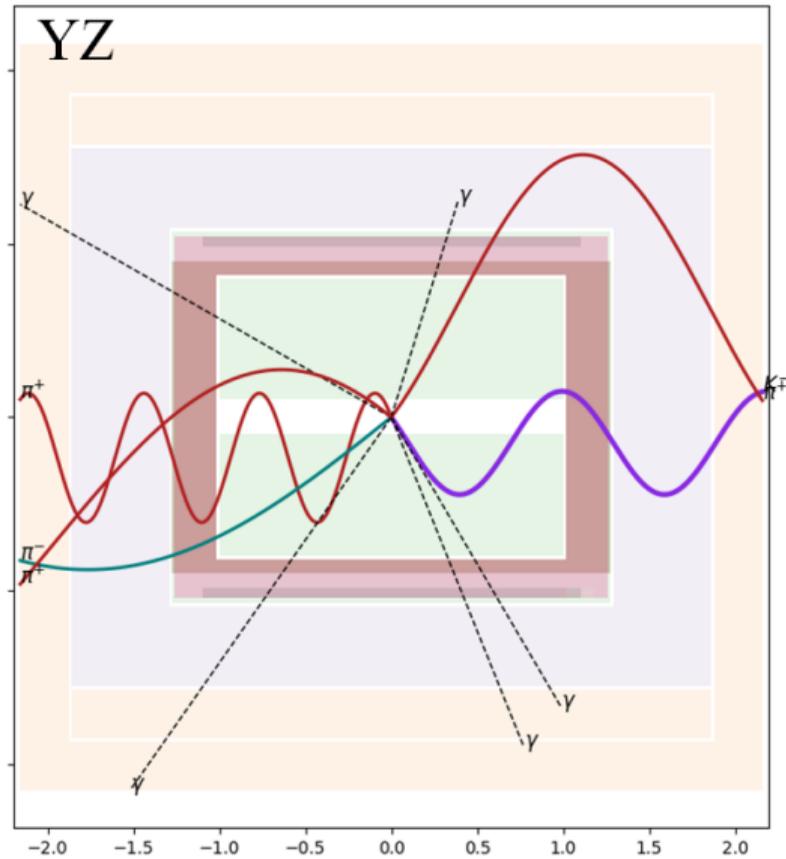
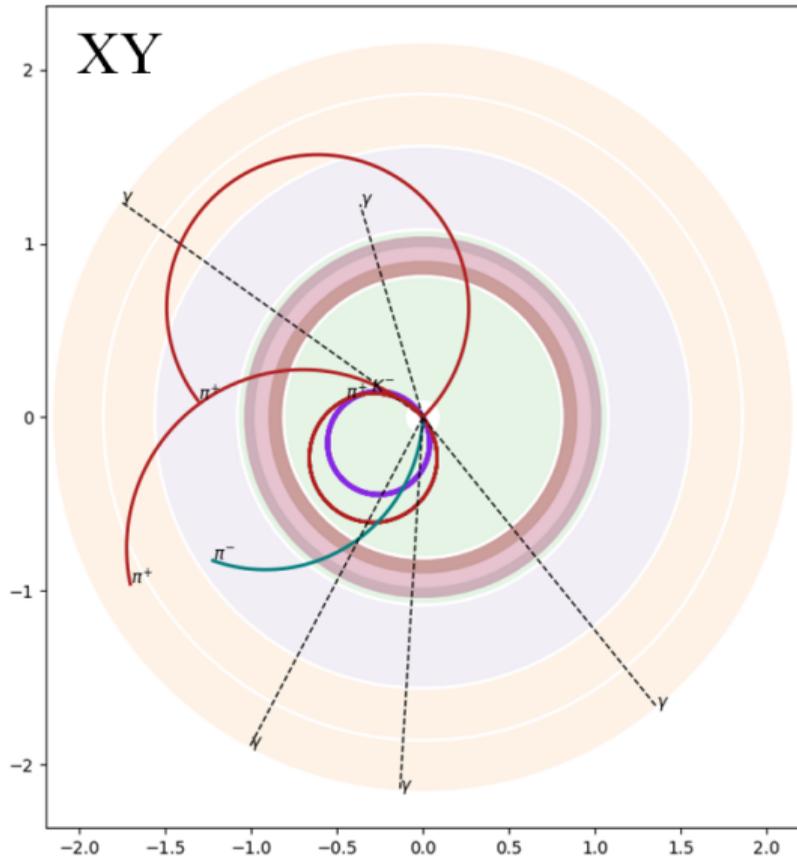


Основные упрощения

- Детектор — вложенные друг в друга цилиндры
- Магнитное поле постоянное и однородное
- Не учитывается взаимодействие с веществом
- Распада частиц — нет

Источники

- Концептуальный проект (https://ctd.inp.nsk.su/wiki/images/8/8b/CDR2_ScTau_ru_vol1.pdf)
- Меморандум по проекту дрейфовой камеры
- Результаты моделирования подсистем с помощью пакета Geant4



Подсистемы детектора

	Подсистема	Реализация в ПО
Трековая система	Многослойный кремниевый детектор	—
	ВПК	—
	Каскадный ГЭУ ($\mu RWELL$)	—
	Дрейфовая камера	+
	ДК TraPid	±
Калориметр	PureCsl	+
	LXe	—
Системы идентификации	ВПС	+
	ToP	+
	АШИФ	+
	ФАРИЧ	+
	FDIRC	+
	Мюонная система	+

```
class MuonHit(object):
    def __init__(self, layer)

class SctMuon(CylindricalLayer):
    def __init__(self, rMin, zMin, rMax,
                 , zMax)
    # Открытие файлов моделирования
    # мюонной подсистемы

    def process(self, gp, field)
    # Проверка типа частицы
    # Розыгрыш слоя по распределению,
    # взятому из соответствующего файла

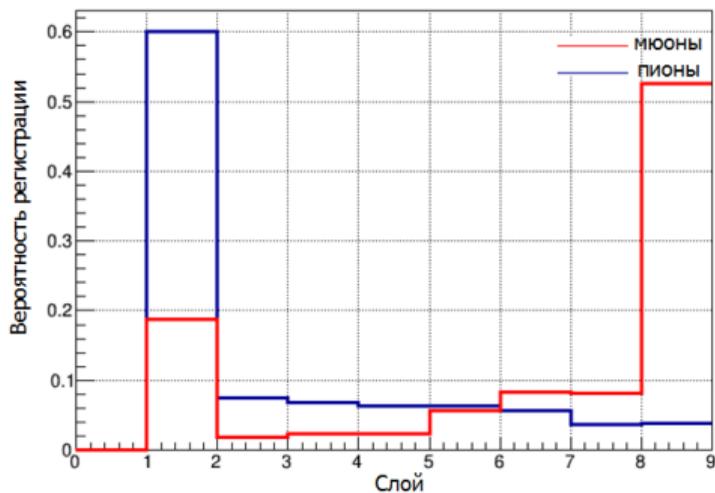
    def moduleP(self, gp)
    def thetaP(self, gp)
```

```
class FDIRCHit(object):
    def __init__(self, angle)

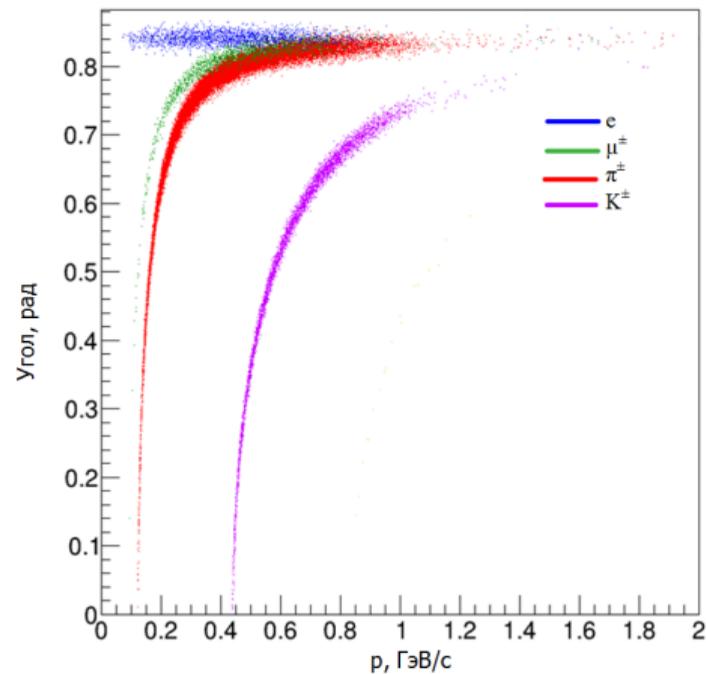
class SctFDIRC(CylindricalLayer):
    def __init__(self, rMin, rMax, zMax,
                 , hole_r, pars)

    def process(self, gp, field)
    # Проверка типа частицы
    # Проверка попадания частицы в
    # подсистему
    # Проверка условий появления
    # черенковского света
    # Розыгрыш угла по нормальному
    # распределению с заданным разрешением

    def modP(self, p4)
    def beta(self, p, mass)
```

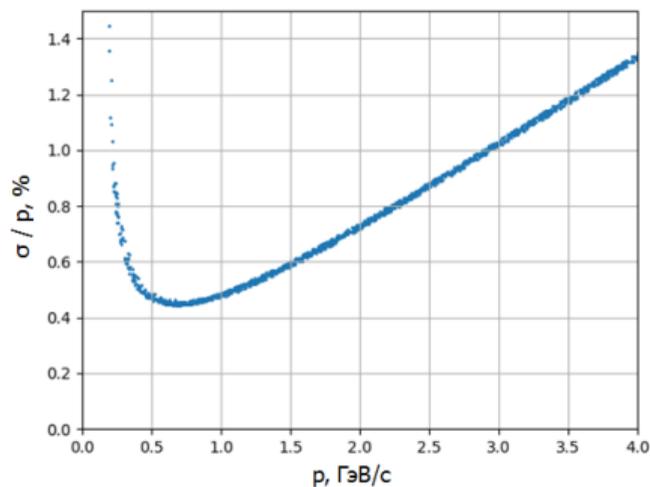


Распределение вероятности для мюонов и пионов долететь до определённого слоя в мюонной системе.



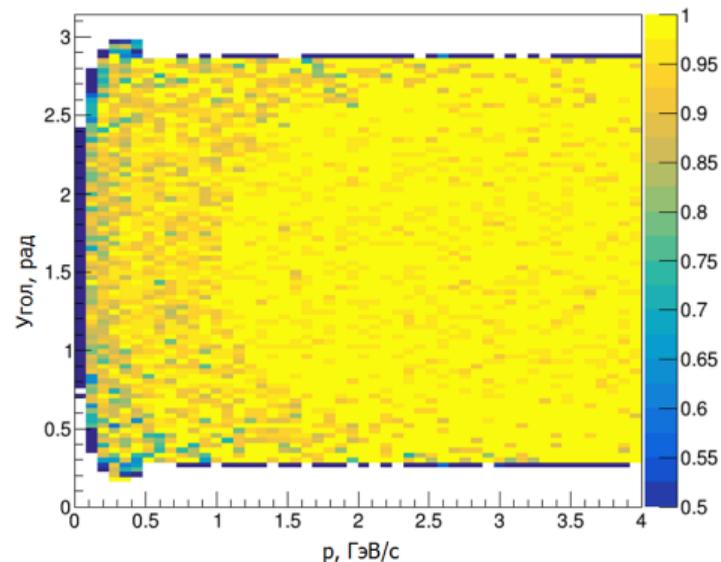
Зависимость угла, регистрируемого в FDIRC от импульса

Параметризация дрейфовой камеры



Зависимость импульсного разрешения для π^0 , летящего под углом к оси z близком к 60° , от импульса.

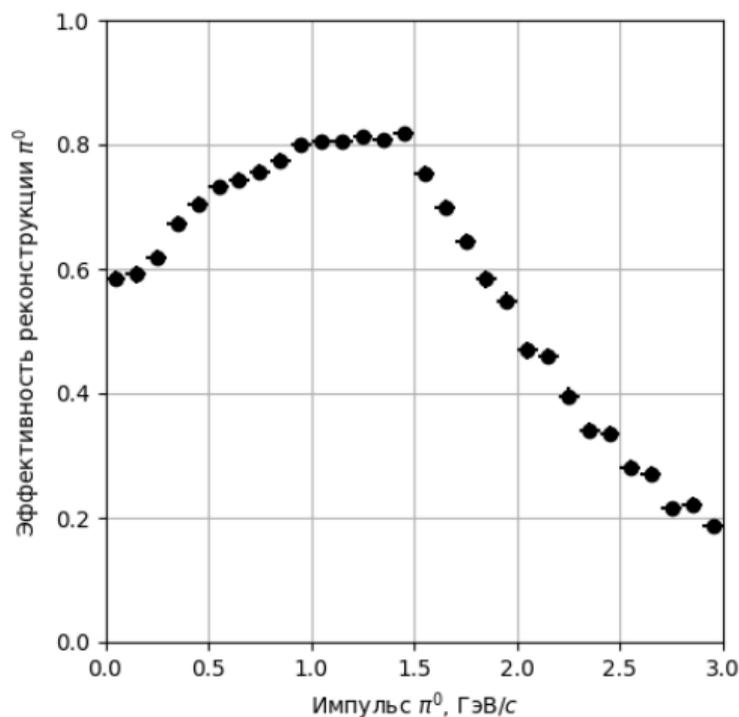
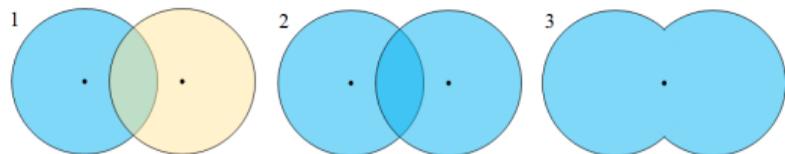
<http://kedr.inp.nsk.su/~todyshev/dcdescription.pdf>



Распределение эффективности регистрации π^0 в дрейфовой камере в зависимости от импульса и угла, между частицей и осью z .

Алгоритм сшивки данных, полученных трековой системой и калориметром:

- Объединение геометрически пересекающихся кластеров
- Нахождение соответствия между кластерами
- Пересчёт характеристик кластеров (время, энергия, размер кластера, точка конверсии)



Изменение параметров детектора:

```
coil.rMax 1.86  
field.value 1.4  
tracker.radLen 185  
tracker.corrMtx.xij -0.08  
tracker.momres [0.001,0.04]
```

Описание структуры детектора

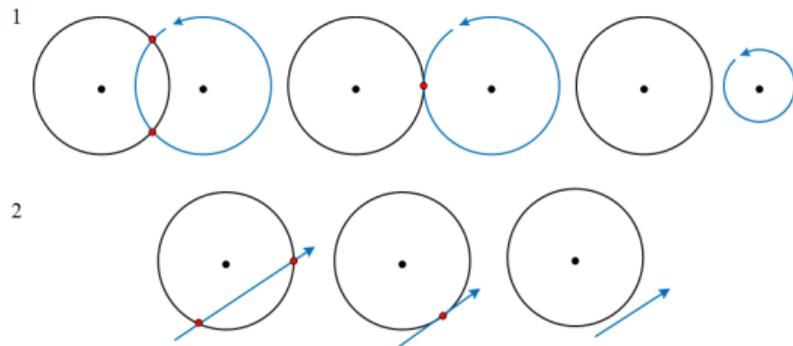
```
{  
  "calo": {  
    "rMax": 1.55,  
    "rMin": 1.09,  
    "zMax": 1.86,  
    "zMin": 1.293  
  },  
  "field": {  
    "value": 1.5  
  },  
  "fdirc": {  
    "rMax": 1.04,  
    "rMin": 0.98,  
    "zMax": 1.1,  
    "hole_r": 0.3,  
    "n": 1.5  
  },  
  ...  
}
```

После настройки среды:

```
$ cmake test
```

При сборке директории в build:

```
$ cmake ../aurora/Project/WorkDir  
$ make  
$ source x86_64-slc7-gcc8-opt/setup.sh  
$ make test
```



Схематическое изображение случаев в плоскости XY, проверенных тестами для цилиндра и 1 — винтовой линии, 2 — прямой.

Результаты:

- Обновлена параметризация дрейфовой камеры согласно меморандуму
- Внедрена система идентификации частиц FDIRC
- Внедрена мюонная система
- Разработан пользовательский интерфейс для изменения параметров моделируемого детектора
- Проверена правильность проведения частиц через объем детектора
- Разработана реконструкция частиц с учётом геометрического пересечения калометрических кластеров
- Проверены отклики разрабатываемых подсистем на соответствие заложенному описанию

Планы:

- Перенос разработанного программного обеспечения в рамки проекта Aurora с использованием функционала Gaudi на языке C/C++

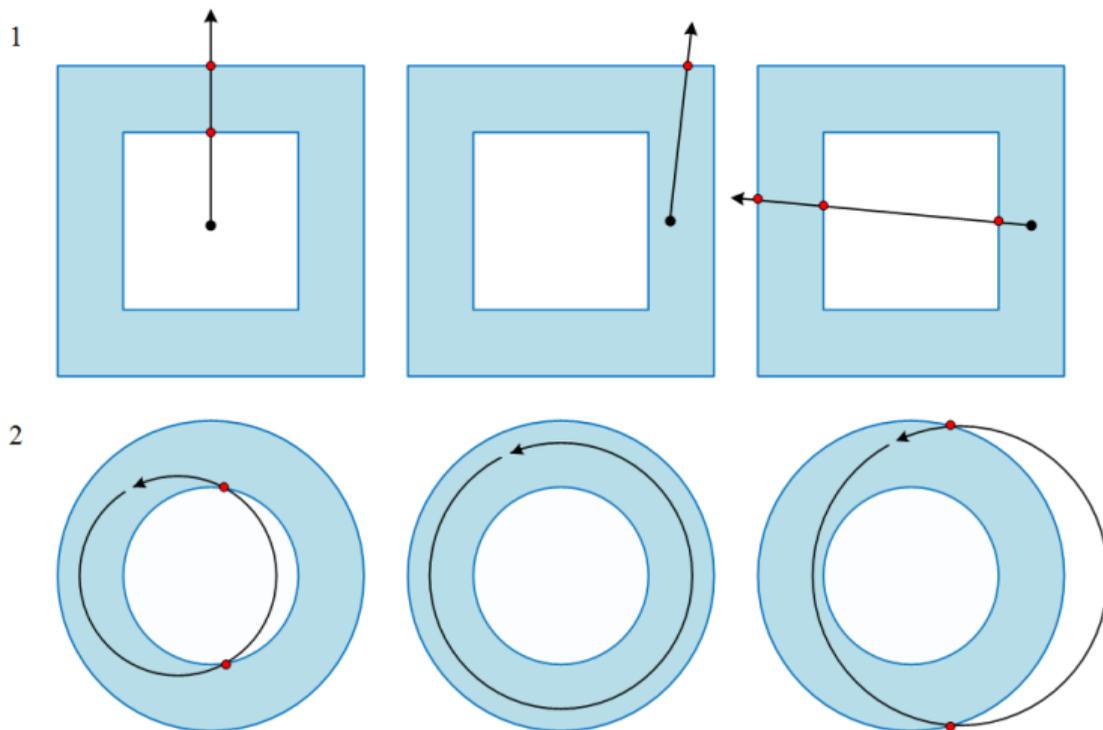
Результаты:

- Обновлена параметризация дрейфовой камеры согласно меморандуму
- Внедрена система идентификации частиц FDIRC
- Внедрена мюонная система
- Разработан пользовательский интерфейс для изменения параметров моделируемого детектора
- Проверена правильность проведения частиц через объем детектора
- Разработана реконструкция частиц с учётом геометрического пересечения калометрических кластеров
- Проверены отклики разрабатываемых подсистем на соответствие заложенному описанию

Планы:

- Перенос разработанного программного обеспечения в рамки проекта Aurora с использованием функционала Gaudi на языке C/C++

Спасибо за внимание!



Схематическое изображение случаев, проверенных тестами, для цилиндрического слоя для случаев: 1 — в плоскости XZ для прямой линии, 2 — в плоскости XY для винтовой линии.

Настройка среды:

```
$ setupSCTAU  
$ asetup SCTauSim, master, latest
```

Создание рабочей директории:

```
$ mkdir workarea  
$ cd workarea  
$ mkdir run
```

Копирование входных файлов:

```
$ cd ../run  
$ cp /home/razuvaev/public/misc/pi_ms_f1_mppc2_px3_d200_mla4_graph2d.root .  
$ cp /home/razuvaev/public/misc/gun1.cfg .  
$ cp /home/whitem/public/misc/g4beamline_pi_plus_100k_parse.root .  
$ cp /home/whitem/public/misc/g4beamline_mu_plus_100k_parse.root .
```

Запуск:

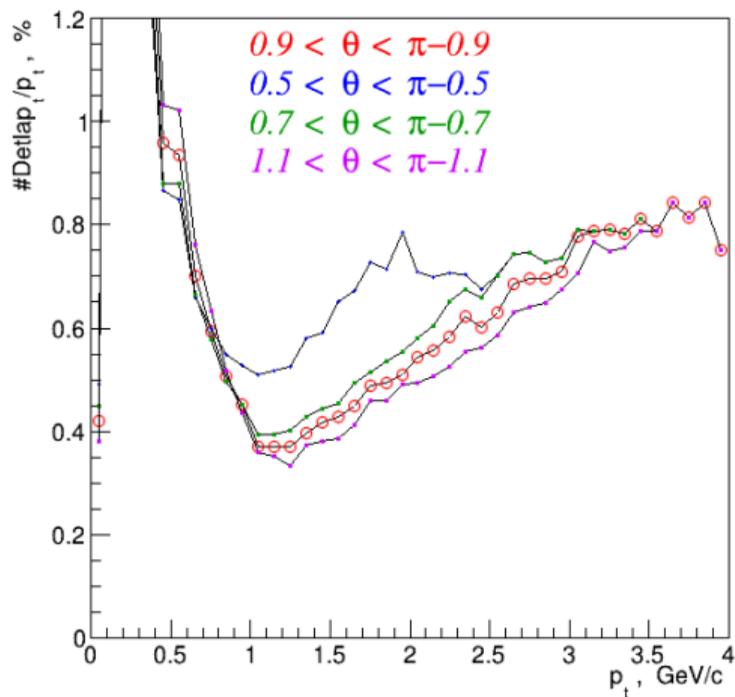
```
$ runparsim.py
```

Параметр	Значение	Пример
<code>-b, --batch</code>	выключает визуализацию	<code>runparsim.py -b</code>
<code>-i, --input</code>	задаёт входной файл	<code>runparsim.py -i ./jpsi_gen.root</code>
<code>-o, --output</code>	задаёт файл для выходных данных	<code>runparsim.py -o ./jpsi_out.root</code>
<code>-n, --neve</code>	задаёт количество событий (по умолчанию 3142)	<code>runparsim.py -n 12345</code>
<code>-c, --change</code>	изменение параметров детектора (необходим конфигурационный файл)	<code>runparsim.py -c parameters.dat</code>
<code>-g, --gun</code>	генератор частиц	<code>runparsim.py -g</code>
<code>-ig, --input-gun</code>	чтение конфигураций генератора	<code>runparsim.py -ig filename.json</code>
<code>--profile</code>	вывод перечня использованных функций и времени их выполнения	<code>runparsim.py --profile</code>

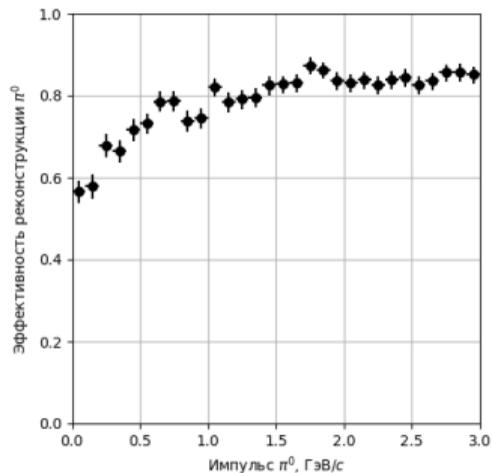
Подсистемы детектора

	Подсистема	Параметризация	PAPAS	ПО
Трековая система	Многослойный кремниевый детектор	—	—	—
	ВПК	—	—	—
	Каскадный ГЭУ ($\mu RWELL$)	—	—	—
	Дрейфовая камера	+	+ BaBar	+
	ДК TraPid	±	—	±
Калориметр	PureCsl	+	+	+
	LXe	—	—	—
Системы идентификации	ВПС	+	+	+
	ToP	+	+	+
	АШИФ	+	+	+
	ФАРИЧ	+	+	+
	FDIRC	+	—	+
	Мюонная система	+	—	+

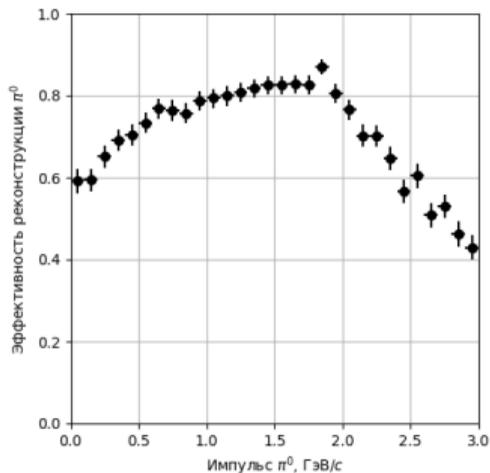
Импульсное разрешение дрейфовой камеры



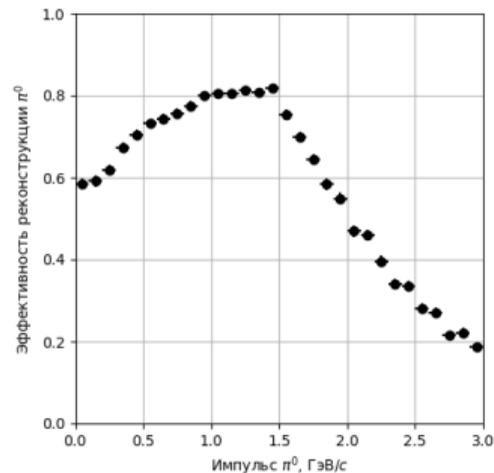
Реконструкция частиц



Размер кластера 5 см



Размер кластера 7,5 см



Размера кластера 10 см

Зависимость размера кластера от энергии фотона

CsI crystal size $5 \times 5 \times 35 \text{ cm}^3$

