

Разработка и оптимизация методов реконструкции данных с фотонного спектрометра PHOS эксперимента ALICE

Научный руководитель: Блау Дмитрий Сергеевич, к.ф-
м.н.

Студент: Бахтин Павел Андреевич

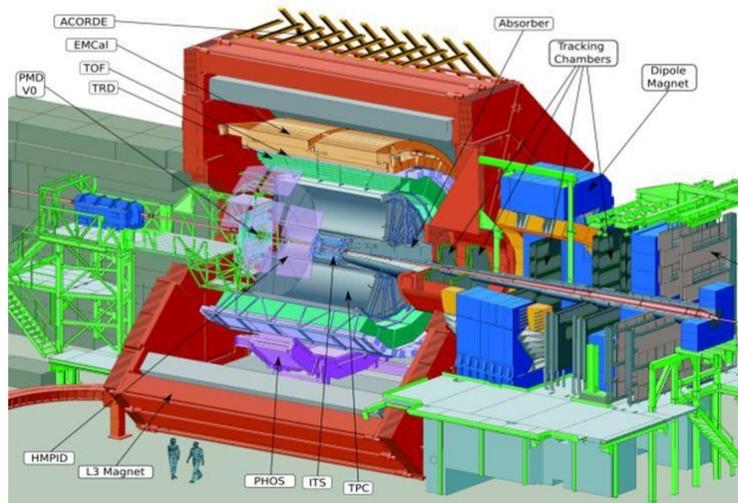
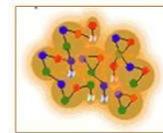
Введение

Цель эксперимента:
обнаружение и определение
свойств кварк-глюонной
материи.

PHOS - поперечно-
сегментированный гомгенный
электромагнитный калориметр
из $PbWO_4$.

Расстояние до него (физически)
478 см, ~ 1 радиационная длина

Эксперимент ALICE на Большом адронном коллайдере

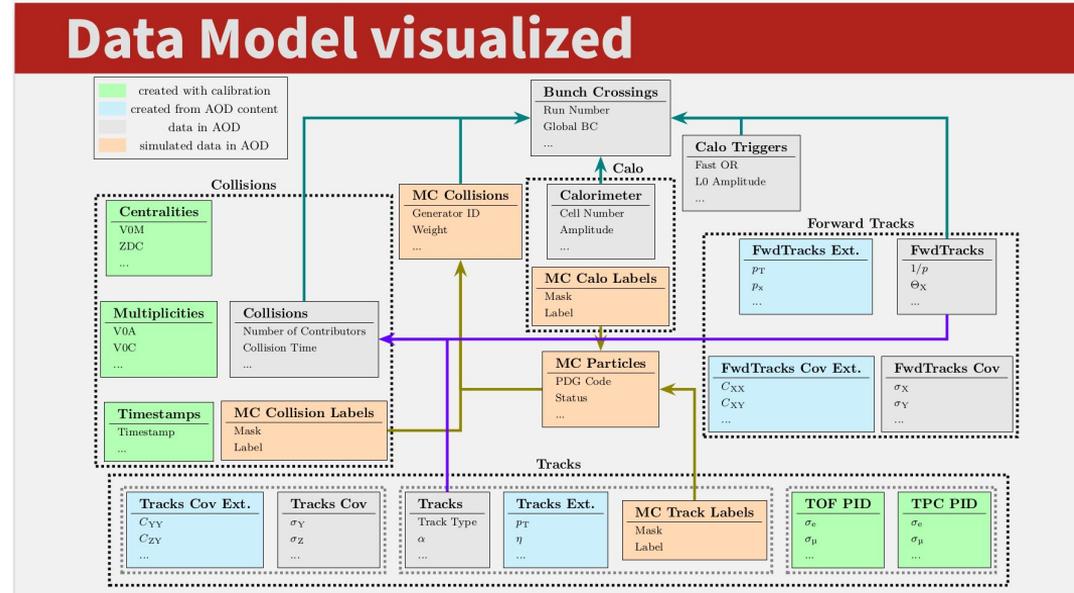


- Трековая система
 - Магнит L3
 - Внутренний трекер ITS
 - Время-проекционная камера TPC
- Детекторы PID
 - TOF
 - TRD
 - HMPID
- Калориметры
 - PHOS
 - EMCAL
 - ZDC
- Мюонная система
 - Магнит
 - Трековые камеры
 - Триггерные камеры
- Триггерные детекторы
 - T0
 - V0

Модель данных в программном пакете O2

Тип данных: связанные
между собой таблицы
(Индексами)

При использовании
монте-карло
моделирования, все
последующие данные по
цепочке создаются из MC.



Создание и обработка AO2D таблиц

Процесс создания при моделировании делится на несколько этапов, каждый из которых описывается последовательностью действий, а она делится на отдельные задачи.

Задача - это C++ структура с функциями `init` и несколькими `process`, которые не могут изменять содержание получаемых таблиц.

Для работы с информацией электромагнитного калориметра нужны следующие таблицы:

- 1) Clusters (создаются из Cells и CaloTriggers)
- 2) MC_Particles_001 (Содержат параметры энергии, импульсов, точки рождения, etc)

Header file: [Common/DataModel/CaloClusters.h](#)

Is used in:

- `o2::aod::CaloAMBCluster = o2::aod::CaloAmbiguousClusters::iterator`

Name		Getter	Type	Comment
<code>o2::soa::Index</code>	GI	<code>globalIndex</code>	<code>int64_t</code>	
<code>o2::aod::calocluster::BCId</code>	I	<code>bcId</code>	<code>int32_t</code>	BC index
<code>o2::aod::calocluster::Px</code>		<code>px</code>	<code>float</code>	momenta components
<code>o2::aod::calocluster::Py</code>		<code>py</code>	<code>float</code>	
<code>o2::aod::calocluster::Pz</code>		<code>pz</code>	<code>float</code>	
<code>o2::aod::calocluster::E</code>		<code>e</code>	<code>float</code>	
<code>o2::aod::calocluster::Module</code>		<code>mod</code>	<code>uint8_t</code>	module/supermodule number
<code>o2::aod::calocluster::Ncell</code>		<code>ncell</code>	<code>uint8_t</code>	cluster multiplicity
<code>o2::aod::calocluster::X</code>		<code>x</code>	<code>float</code>	cluster local coordinates
<code>o2::aod::calocluster::Z</code>		<code>z</code>	<code>float</code>	
<code>o2::aod::calocluster::GlobalX</code>		<code>globalx</code>	<code>float</code>	cluster global coordinates
<code>o2::aod::calocluster::GlobalY</code>		<code>globaly</code>	<code>float</code>	cluster global coordinates
<code>o2::aod::calocluster::GlobalZ</code>		<code>globalz</code>	<code>float</code>	cluster global coordinates
<code>o2::aod::calocluster::Time</code>		<code>time</code>	<code>float</code>	cluster time (seconds)
<code>o2::aod::calocluster::NLM</code>		<code>nlm</code>	<code>uint8_t</code>	number of local maxima
<code>o2::aod::calocluster::M02</code>		<code>m02</code>	<code>float</code>	longer dispersion axis
<code>o2::aod::calocluster::M20</code>		<code>m20</code>	<code>float</code>	shorter dispersion axis
<code>o2::aod::calocluster::TrackDist</code>		<code>trackdist</code>	<code>float</code>	distance to closest track
<code>o2::aod::calocluster::TrackIndex</code>		<code>trackIndex</code>	<code>uint8_t</code>	index of closest track
<code>o2::aod::calocluster::FiredTrigger</code>		<code>firedTrigger</code>	<code>uint8_t</code>	Matched with trigger tile
<code>o2::aod::calocluster::DistBad</code>		<code>distBad</code>	<code>float</code>	distance to closest bad channel

Физическая задача

Электромагнитные калориметры позволяют регистрировать спектр электромагнитного излучения - которое внутренними трековыми системами регистрируется только в очень узком диапазоне и с малой вероятностью (для высоких энергий).

Поскольку технически регистрируется энергия, выделенная частицей в сцинтилляторе - то возникает необходимость решить обратную задачу. То есть, восстановить по полученному сигналу энергии частиц, породивших ту или иную структуру энерговыведения.

Выборка

В таком случае было предложено исключить вероятность не попасть в PHOS из рассмотрения (поскольку она вычисляется из расположения детектора), и в качестве тестовой статистики рассмотреть:

1000 фотонов

С энергиями от 0,5 до 10 ГэВ

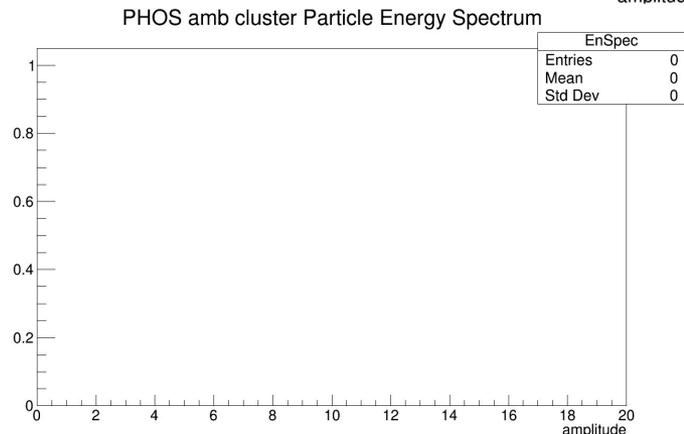
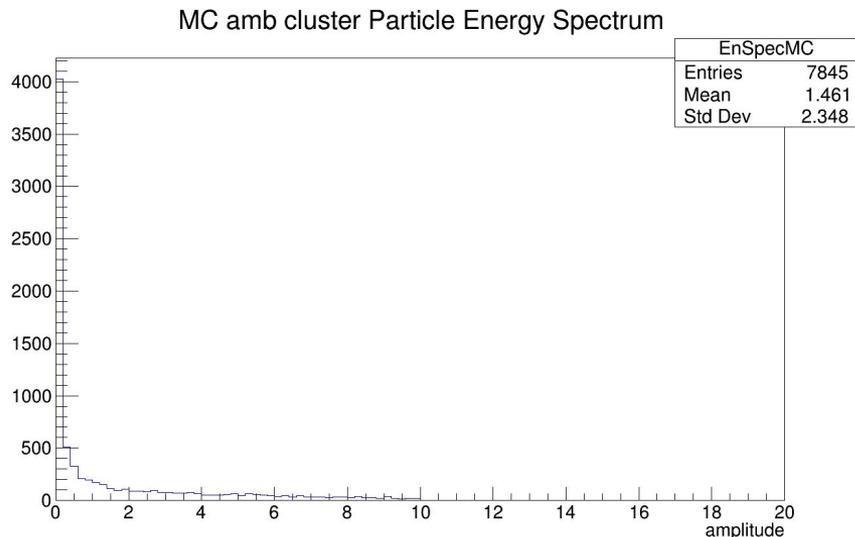
Исходный спектр - равномерный

Ширина распределения по псевдобыстроте 0,02 единицы (+-0,01)

Ширина распределения по азимутальному углу 1 градус (+-0,5)

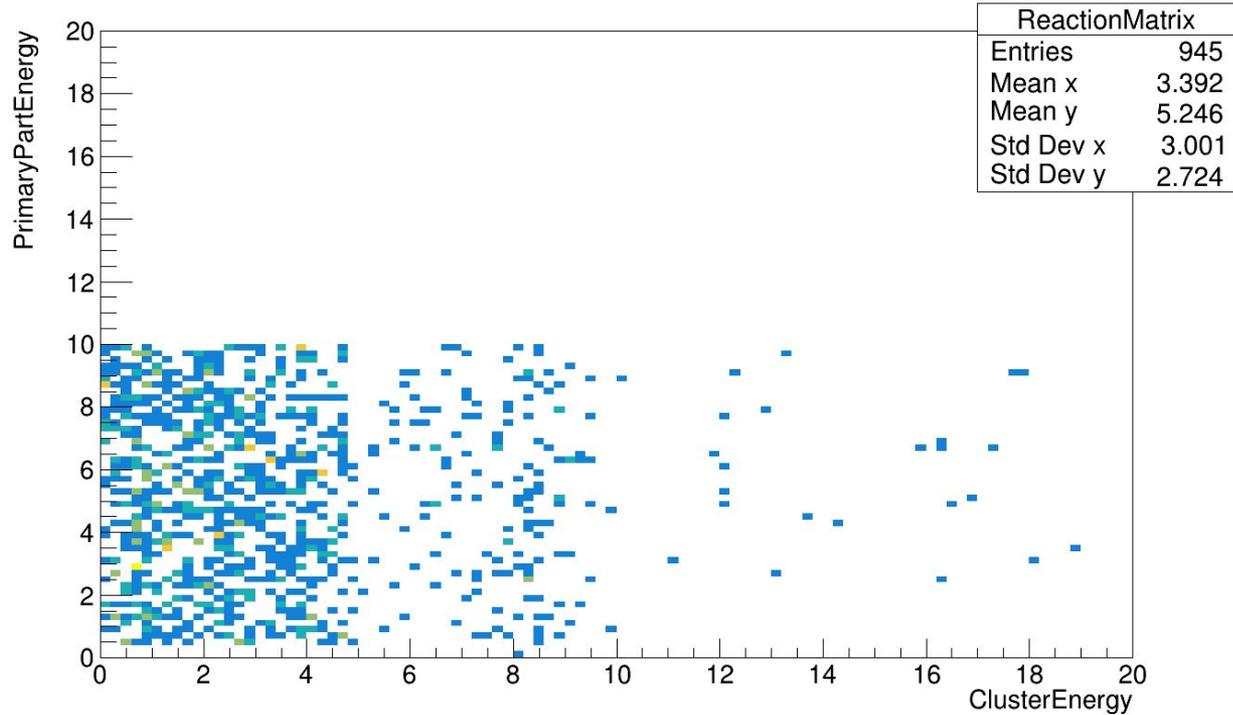
Анализ AO2D

На данном слайде представлены результаты получения полного спектра всех возникших частиц путем монте-карло моделирования и распределение по энергиям кластеров. Несложно заметить, что распределение по энергиям кластеров - не заполняется.



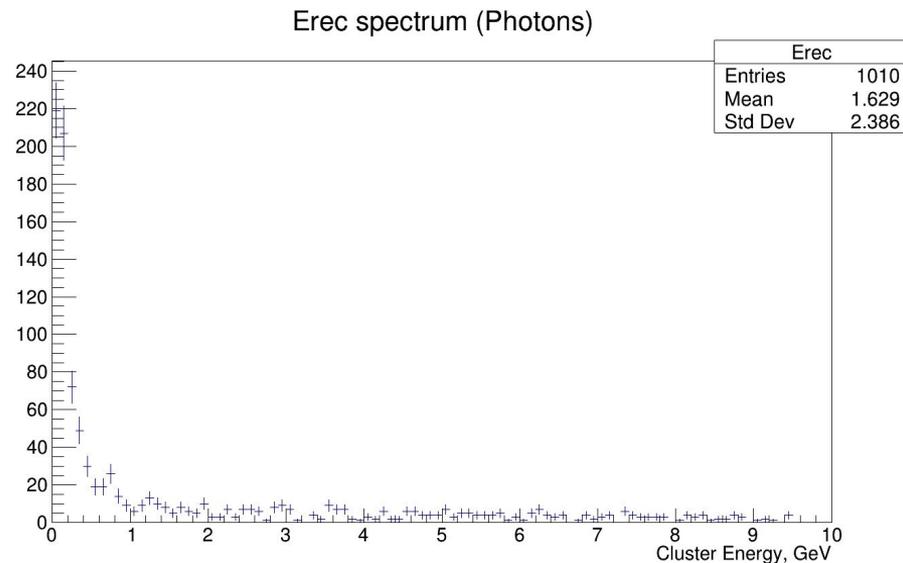
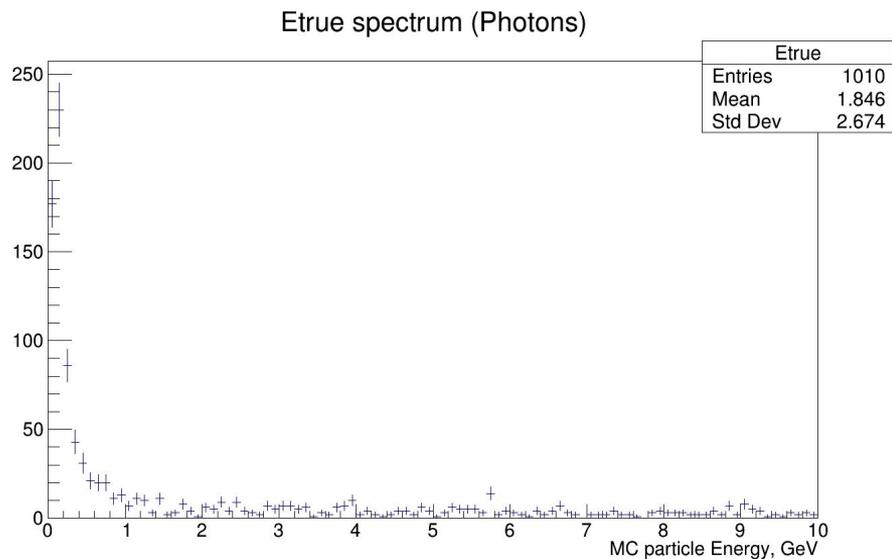
Анализ АО2D

Matrix, includes largest cluster energy of BC and primary particle energy



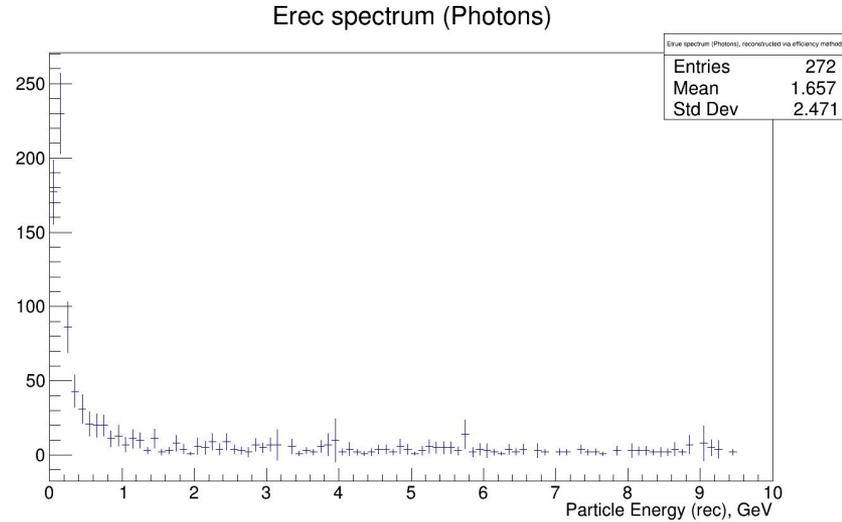
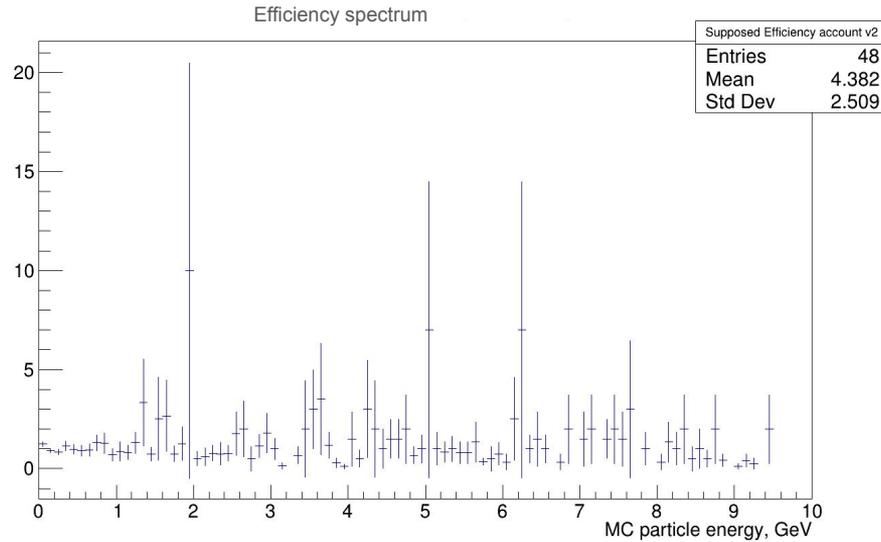
Анализ путем прямой кластеризации данных

В данном случае можно увидеть следующие распределения по энергиям: справа - по энергиям кластеров, слева - по энергиям частиц, выделивших максимум энергии при формировании кластера.



Эффективность-1

Тогда можно построить эффективность - спектр, полученный как отношение двух спектров друг к другу, и с его помощью связать энергии.



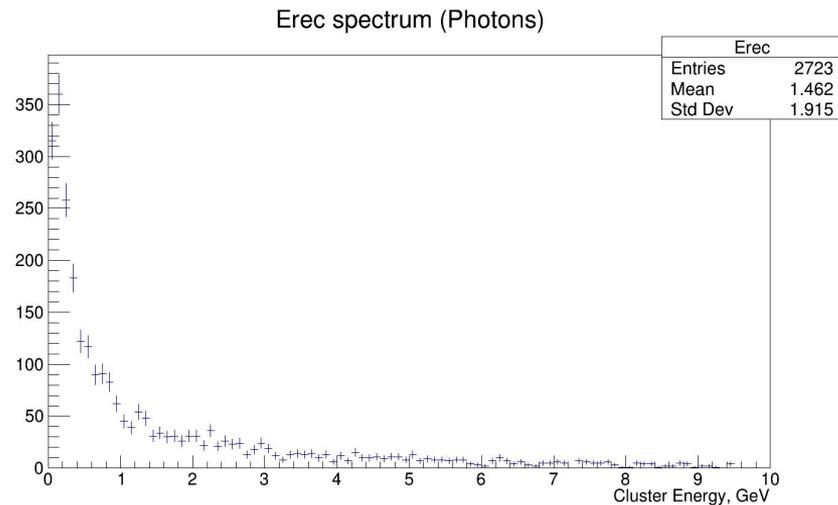
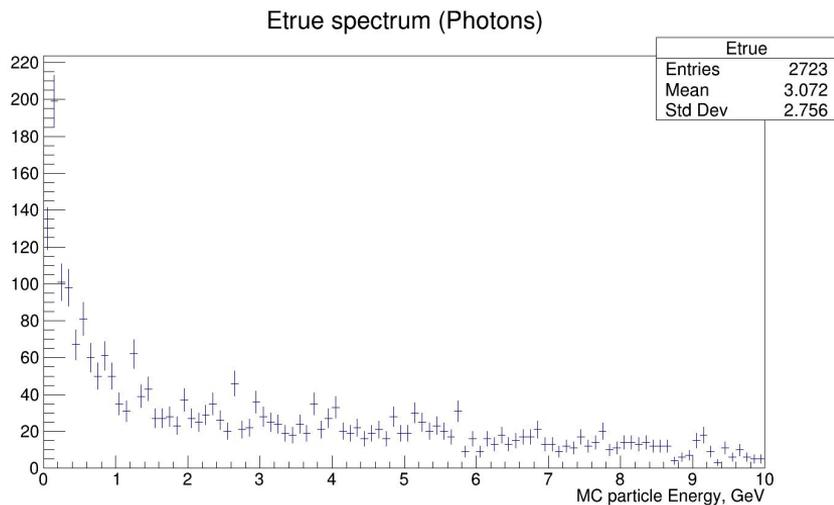
Бины с необычным значением эффективности.

Irr Part PDG: 22
Irr Part Edep: 2.3403
Irr Part Energy: 0.00189405
Irr Part fm PDG: 11
Irr Part sm PDG: -1072203375
Irr Part fd PDG: -1072203375
Irr Part ld PDG: -1072203375
Vertex x: -0.562543
Vertex y: -257.42
Vertex z: -1.95118
Irr part primaries list size is: 4
Number of Primary: 1
Deposited Energy: 2.3403
Particle TrackID: 30
Particle PDG: 22
Particle Enregy: 0.00189405
Number of Primary: 2
Deposited Energy: 0.536122
Particle TrackID: 16
Particle PDG: 22
Particle Enregy: 0.00516021
Number of Primary: 3
Deposited Energy: 0.0209125
Particle TrackID: 2
Particle PDG: 11
Particle Enregy: 6.80974
Number of Primary: 4
Deposited Energy: 0.0209125
Particle TrackID: 6
Particle PDG: 11
Particle Enregy: 0.275258

Текущее решение
проблемы смешивания
выделенных энергий и
перераспределений:
Усреднить полные энергии
по всем частицам
кластера.

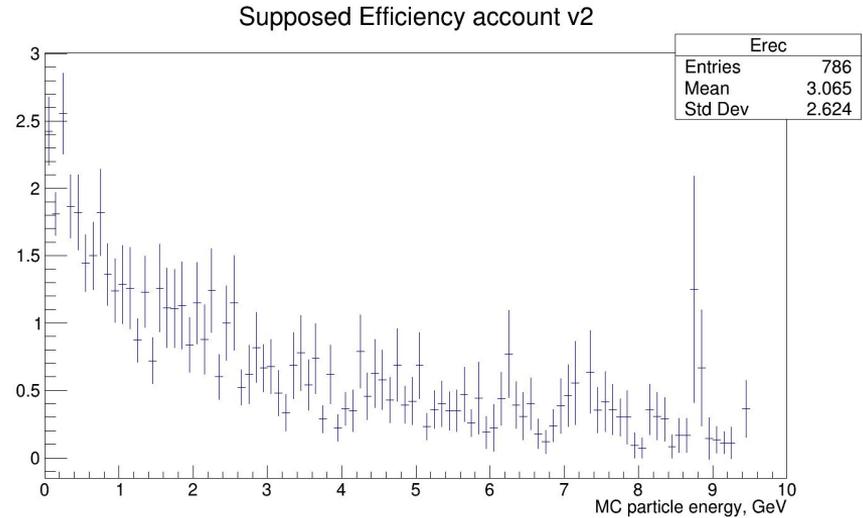
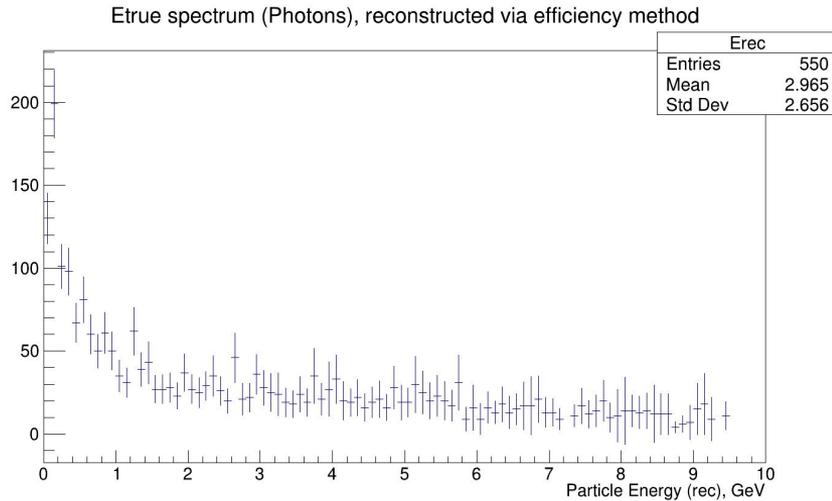
Кластеризация - 2

Несложно заметить, что такое соотношение дает спектр энергий, который менее быстро убывает, чем в случае спектра отдельных избранных частиц.



Эффективность - 2

Здесь можно отметить, что распределения куда более равномерно, чем в случае 1. Однако и в этом случае.



Событие

Irr Part PDG: 22
rr Part Edep: 0.378327
Irr Part Energy: 0.00197892
Irr Part fm PDG: 11
Irr Part sm PDG: 0
Irr Part fd PDG: 0
Irr Part ld PDG: 0
Vertex x: -70.9363
Vertex y: -456.741
Vertex z: 2.41643
Irr part primaries list size is: 1
Number of Primary: 1
Deposited Energy: 0.378327
Particle TrackID: 60
Particle PDG: 22
Particle Enregy: 0.00197892

Несложно заметить, что в этой ситуации - нет никакой дополнительной частицы, кроме материнской, от которой могла бы физически проявиться эта энергия - и, разумеется, остается возможность ошибки в алгоритме кластеризации.

Выводы

- Необходимо дождаться исправления собирающего AO2D-файлы алгоритма, чтобы продолжить работать над актуальными методами обработки. Там будет также необходимо встроить внутрь задачи аналогичный способ с описанием кластера (входящие частицы и их энерговыделения).
- В алгоритме кластеризации присутствует вероятность того, что энерговыделение одной частицы будет записано на другую. Необходимо изучить, как и почему.
- Реализовано построение распределений эффективности, которые можно применять для восстановления спектров.

Спасибо за внимание!

Доступная монте-карло информация

Is used in:

- o2::aod::McParticles = o2::aod::McParticles_001

Name		Getter	Type	Comment
o2::soa::Index	GI	globalIndex	int64_t	
o2::aod::mcparticle::McCollisionId	I	mcCollisionId	int32	MC collision of this particle
o2::aod::mcparticle::PdgCode		pdgCode	int	PDG code
o2::aod::mcparticle::StatusCode		statusCode	int	Generators status code or physics process. Do not use directly. Use dynamic columns getGenStatusCode() or getProcess()
o2::aod::mcparticle::Flags		flags	uint8_t	ALICE specific flags, see MCParticleFlags. Do not use directly. Use the dynamic columns, e.g. producedByGenerator()
o2::aod::mcparticle::MothersIds	SAI	mothersIds		Mother tracks (possible empty) array. Iterate over mcparticle.mothers_as<aod::McParticles>()
o2::aod::mcparticle::DaughtersIdSlice	SSLI	daughtersIds	int32_t	Daughter tracks (possibly empty) slice. Check for non-zero with mcparticle.has_daughters(). Iterate over mcparticle.daughters_as<aod::McParticles>()
o2::aod::mcparticle::Weight		weight	float	MC weight
o2::aod::mcparticle::Px		px	float	Momentum in x in GeV/c
o2::aod::mcparticle::Py		py	float	Momentum in y in GeV/c
o2::aod::mcparticle::Pz		pz	float	Momentum in z in GeV/c
o2::aod::mcparticle::E		e	float	Energy
o2::aod::mcparticle::Vx		vx	float	X production vertex in cm
o2::aod::mcparticle::Vy		vy	float	Y production vertex in cm
o2::aod::mcparticle::Vz		vz	float	Z production vertex in cm
o2::aod::mcparticle::Vt		vt	float	Production time
o2::aod::mcparticle::ProducedByGenerator	D	producedByGenerator	bool	True if particle produced by the generator (==TMCPProcess::kPrimary); False if by the transport code
o2::aod::mcparticle::FromBackgroundEvent	D	fromBackgroundEvent	bool	Particle from background event
o2::aod::mcparticle::GetGenStatusCode	D	getGenStatusCode	int	The native status code put by the generator, or -1 if a particle produced during transport
o2::aod::mcparticle::GetHepMCStatusCode	D	getHepMCStatusCode	int	The HepMC status code put by the generator, or -1 if a particle produced during transport
o2::aod::mcparticle::GetProcess	D	getProcess	int	The VMC physics code (as int) that generated this particle (see header TMCPProcess.h in ROOT)
o2::aod::mcparticle::IsPhysicalPrimary	D	isPhysicalPrimary	bool	True if particle is considered a physical primary according to the ALICE definition
o2::aod::mcparticle::Phi	E	phi	float	Phi in the range [0, 2pi)
o2::aod::mcparticle::Eta	E	eta	float	Pseudorapidity, conditionally defined to avoid FPEs
o2::aod::mcparticle::Pt	E	pt	float	Transverse momentum in GeV/c
o2::aod::mcparticle::P	E	p	float	Total momentum in GeV/c
o2::aod::mcparticle::Y	E	y	float	Particle rapidity, conditionally defined to avoid FPEs

Код генерации Монте-Карло модели

```
#Executive .sh code, prepare via chmod+x <filename>.sh and enforce alienv surrounding variables
# export IGNORE_VALIDITYCHECK_OF_CCDB_LOCALCACHE=1
# export ALICEO2_CCDB_LOCALCACHE=".ccdb"
NWORKERS=${NWORKERS:-8}
MODULES="--skipModules ZDC"
SIMENGINE=${SIMENGINE:-TGeant4}
# create workflow
# run number 302000 is ok for pp and -0.5 T
${O2DPG_ROOT}/MC/bin/o2dpg_sim_workflow.py -eCM 14000 -col pp -gen boxgen -proc cdiff -tf 1 \
    -ns 1000 -e ${SIMENGINE} \
    -j ${NWORKERS} -interactionRate 1000 \
    -run 302000 -seed 624 \
    -confKey 'BoxGun.pdg=22; BoxGun.number=1; BoxGun.prange[0]=0.5; BoxGun.prange[1]=10.;
BoxGun.phirange[0]=269.5; BoxGun.phirange[1]=270.5; BoxGun.eta[0]=-0.01; BoxGun.eta[1]=0.01;'
# run workflow
${O2DPG_ROOT}/MC/bin/o2_dpg_workflow_runner.py -f workflow.json -tt aod -j 1 --mem-limit 20000
```

Полезные ссылки

<https://aliceo2group.github.io/analysis-framework/docs/datamodel/ao2dTables.htm>

↓ - описание содержимого таблиц в O2

<http://alice-ccdb.cern.ch/browse> - ссылка с данными, задающими параметры детектора (ALICE) при желании собрать вручную локальную копию